

С. Т. Мухамбетжанова, А. С. Тен, Н. Г. Голикова

ИНФОРМАТИКА

Учебник для 9 класса общеобразовательной школы

9

Рекомендовано Министерством образования и науки
Республики Казахстан



Алматы «Атамұра» 2019

УДК 373.167.1
ББК 32.973я72
М92

*Учебник подготовлен в соответствии
с Типовой учебной программой по учебному предмету
«Информатика» для 5–9 классов уровня основного среднего
образования по обновленному содержанию*

Условные обозначения:



Самое главное по изучаемой теме



Вопросы и задания для самообучения



Вопросы и задания для самоконтроля



Практикум



Исследование



Домашнее задание



Для тех, кто хочет знать больше



Индивидуальная работа



Парная работа



Групповая работа



Работа класса



Задания к теме

Мухамбетжанова С. Т. и др.

М92 Информатика: Учебник для 9 кл. общеобразоват. шк. / С. Т. Мухамбетжанова, А. С. Тен, Н. Г. Голикова – Алматы: Атамұра, 2019. – 220 с.

ISBN 978-601-331-608-6

ISBN 978-601-331-608-6

© Мухамбетжанова С. Т., Тен А. С.,
Голикова Н. Г., 2019

© «Атамұра», 2019

СОДЕРЖАНИЕ

Предисловие	4
Раздел I. РАБОТА С ИНФОРМАЦИЕЙ	
1.1. Свойства информации.....	8
1.2. Совместная работа с документами.....	15
1.3. Сетевой этикет	23
Итоговые задания	32
Раздел II. ВЫБИРАЕМ КОМПЬЮТЕР	
2.1. Конфигурация компьютера	38
2.2. Выбор программного обеспечения.....	46
2.3. Расчет стоимости компьютера	55
Итоговые задания	64
Раздел III. БАЗЫ ДАННЫХ	
3.1. Базы данных	70
3.2. Правила создания базы данных в электронной таблице Excel	77
3.3. Методы поиска информации	87
3.4. Сортировка и фильтрация данных	97
3.5. Работа с базой данных.....	106
Итоговые задания	114
Раздел IV. ПРОГРАММИРОВАНИЕ АЛГОРИТМОВ НА ЯЗЫКЕ ПРОГРАММИРОВАНИЯ PYTHON (Пайтон)	
4.1. Одномерный массив.....	120
4.2. Поиск элемента с заданными свойствами	128
4.3. Перестановка элементов	137
4.4. Двумерный массив.....	144
4.5. Сортировка	152
4.6. Удаление и вставка элемента.....	160
Итоговые тесты и задания	168
Практические задания	170
Раздел V. СОЗДАНИЕ 2D-ИГРЫ НА ЯЗЫКЕ ПРОГРАММИРОВАНИЯ PYTHON	
5.1. Библиотека PyGame	173
5.2. Задний фон и персонаж игры.....	183
5.3. Анимирование персонажей	190
5.4. Программирование условий	201
Итоговые практические задания.....	208
Глоссарий	210
Рекомендуемая литература	213
Рекомендуемые интернет-ресурсы	214
Приложение	216

ПРЕДИСЛОВИЕ

Дорогие девятиклассники!

Мы живем в XXI веке на этапе смены технологической парадигмы. Информационные технологии, определившие образ и сущность XX века, уступили место Smart-технологиям, которые привели к качественным изменениям жизни.

Изменения в социальной среде тесно связаны со сменой технологических тенденций на разных этапах:

- 1-й этап – *доиндустриальное общество*, в котором существовала человеческая организация связи, основанная на *аналоговом мышлении*;
- 2-й этап – *информационное общество* с компьютерной организацией связи на основе *цифрового мышления*;
- 3-й этап – *креативное общество* с социальной организацией связи и *гибридным мышлением*.

Наряду с известными природными, промышленными, сельскохозяйственными, энергетическими, людскими и другими ресурсами появился новый ресурс человечества, в основе которого находится *информация*. С каждым днем данный ресурс только увеличивается. Для качественного и быстрого поиска, обработки, защиты, кодирования, восприятия, передачи и хранения информации применяют компьютеры и современные технические средства.

Чему вы научитесь, изучая предмет «Информатика» по данному учебнику? Основная задача учебника сводится не только к освоению информационно-коммуникационных технологий (ИКТ), но и к формированию познавательных, творческих способностей, навыков работы с информацией и применению их на практике, развитию алгоритмического и логического мышления, привитию информационной культуры.

Современный человек должен свободно ориентироваться в информационных потоках, уметь получать, обрабатывать и использовать информацию с помощью компьютеров, телекоммуникаций и других средств связи.

Мы надеемся, что задачи, вопросы, упражнения, которые необходимо будет выполнять в процессе обучения, привлекут ваше внимание и вызовут интерес как к теоретическим, так и к практическим аспектам информатики.

В 5 классе вы познакомились с двумя основными понятиями – *алгоритм* и *программа*. Алгоритмическое мышление необходимо человеку не только при работе с компьютером, оно полезно в уникальном, оригинальном, творческом подходе к делу, учит превращать абстрактные идеи в реальность на практике.

В 6, 7, 8 классах вы продолжили составлять программы. *Программа* – это и есть алгоритм, представленный на языке программирования.

Изучение алгоритмизации и программирования в школьном курсе информатики имеет два аспекта:

– *развивающий*, под которым понимают развитие алгоритмического мышления учащихся с усилением фундаментальной компоненты курса информатики;

– *программистский* – развитие навыков составления учебных программ. Этот аспект носит профориентационный характер.

В 9 классе вы продолжите формировать операционный стиль мышления, являющегося одним из важных признаков общей культуры человека в цифровом обществе.

Материал в учебнике изложен таким образом, чтобы подвести вас к систематизации, осмыслению, обобщению уже имеющихся знаний и анализу приобретенных.

Условные обозначения, представленные в данном учебнике, придуманы авторами специально для того, чтобы вы могли быстрее ориентироваться в учебнике, а значит, лучше усваивать изучаемый материал.

Учебник содержит много интересного, важного, дополняющего и расширяющего кругозор материала, а также приведены примеры решения типовых задач по изучаемой теме.

Практические навыки работы на компьютере вы будете осваивать, выполняя соответствующие задания, творческие работы, исследовательские проекты, с дифференциацией и интеграцией.

Желаем вам успешного и приятного изучения предмета «Информатика»!

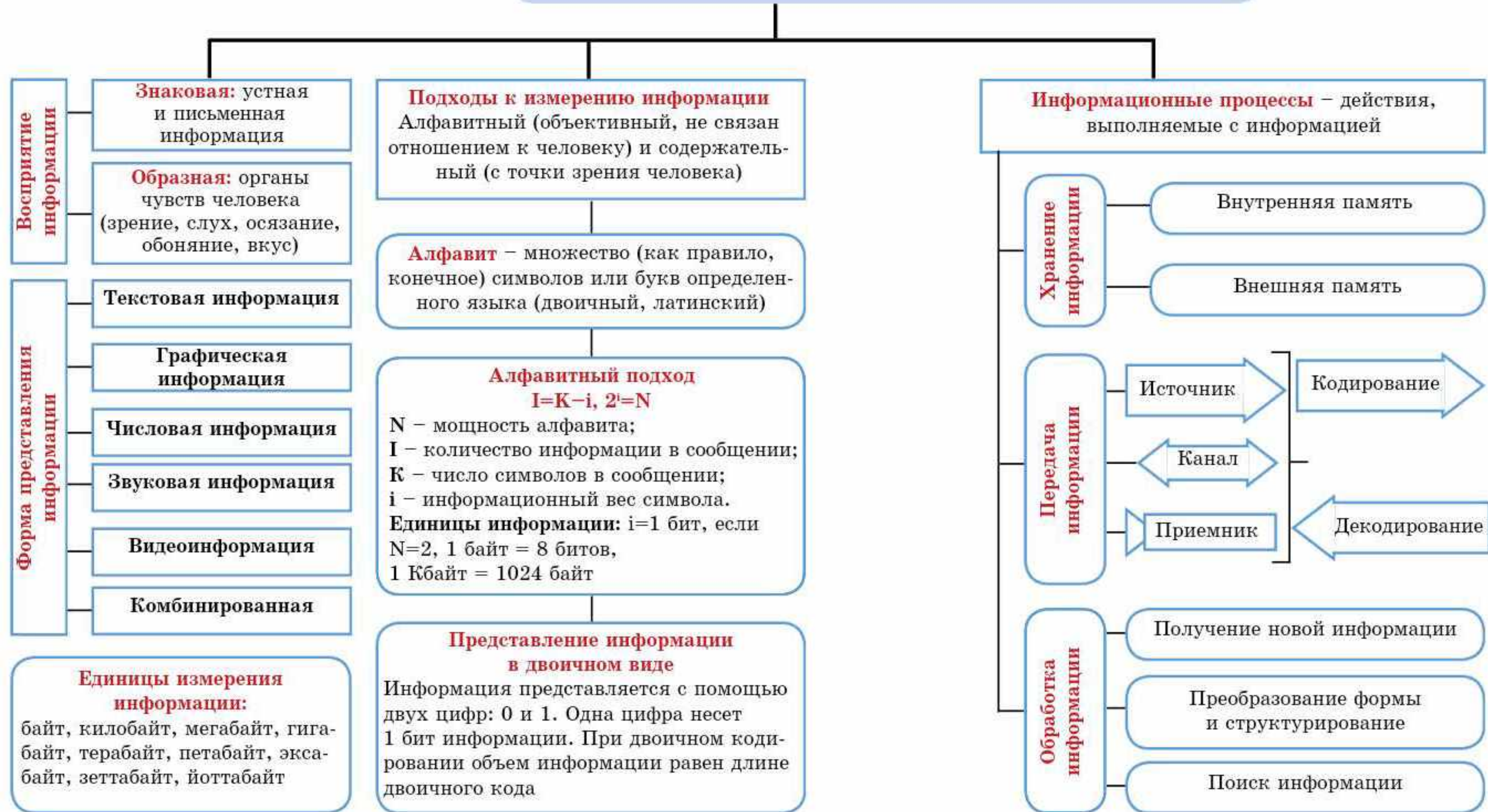
Авторы

СИСТЕМА ОСНОВНЫХ

ПОНЯТИЙ РАЗДЕЛА I

РАБОТА

С ИНФОРМАЦИЕЙ



Раздел I

РАБОТА С ИНФОРМАЦИЕЙ

1.1. Свойства информации

Вы научитесь:

- определять свойства информации (актуальность, точность, достоверность, ценность);
- критически оценивать риски, связанные с продолжительным использованием компьютеров.

Ключевые слова

<i>Свойство информации</i>	<i>Ақпараттың қасиеті</i>	<i>Property Information</i>
<i>Актуальность</i>	<i>Өзектілік</i>	<i>Relevance</i>
<i>Точность</i>	<i>Нақтылық</i>	<i>Accuracy</i>
<i>Достоверность</i>	<i>Анықтылық</i>	<i>Reliability</i>
<i>Ценность</i>	<i>Құндылық</i>	<i>Value</i>

При обработке, сборе, хранении, передаче, анализе и оценке информации необходимо знать их свойства. Критически мыслящий человек в любой ситуации анализирует свойства информации. Он оценивает содержащиеся в ней сведения на актуальность, точность, достоверность, ценность, доступность, полноту и другие свойства.

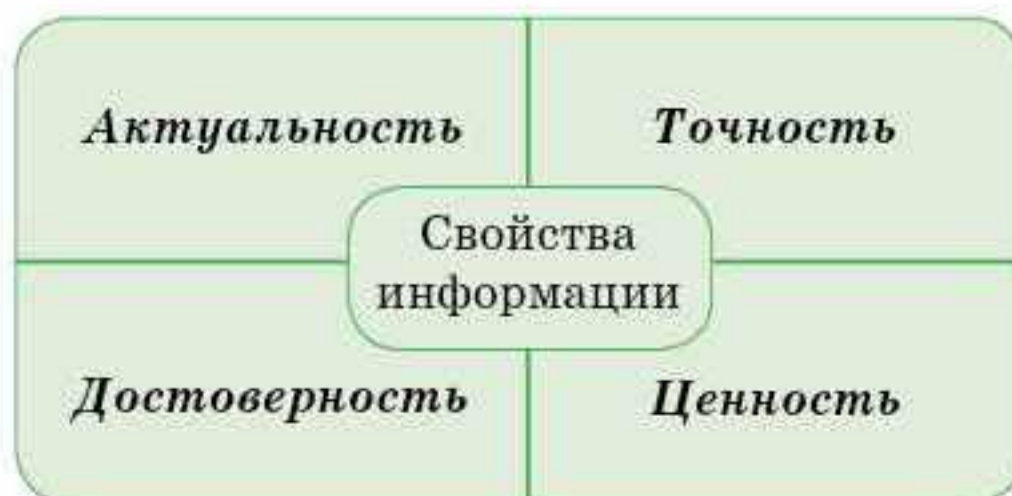


Рис. 1.1. Свойства информации

Мы рассмотрим наиболее важные свойства с точки зрения информатики, перечисленные на рис. 1.1.



Подумайте и ответьте.

Какие бы вы привели примеры качественных сторон информации:

- *актуальность – маңыздылық – relevance,*
- *точность – дәлдік – accuracy,*
- *достоверность – нақтылық – reliability,*
- *ценность – құндылық – value?*



Информация имеет свойство **актуальность**, если она *своевременна, важна и существенна* для настоящего времени.

В зависимости от конкретных ситуаций определяется значимость тех или иных свойств информации. Например, сайт bnews.kz должен содержать самые актуальные и оперативные новости дня в рубрике «Последние новости Казахстана».



Свойство **точность** информации определяется степенью ее ясности, близости к реальному состоянию объекта, процесса, явления.



Подумайте и определите правильный ответ. Какой из них является более точным?

1. Для поездки в город Нур-Султан нужно сесть в поезд.
2. Для поездки в город Нур-Султан нужно купить билет на выбранный день, рейс, время, купе и место. Показать билет проводнику, сесть в поезд на указанное место.





Достоверность информации определяется ее свойством отражать истинное положение дел.

Показатель качества достоверной информации, обозначает ее точность. Основные критерии **достоверной информации** – это отсутствие ложных, искаженных данных, преувеличение или преуменьшение реального факта (слухи, истории и т. д.).



Подумайте и ответьте.

Какая информация будет достоверной, верной?

1. Вы пропустили занятие. Ваш одноклассник сказал, что задали только прочитать параграф, а о подготовке к контрольной работе не сказал.
2. Температура воздуха в городе Атырау 30 градусов по Цельсию.
3. Свойства информации бывают актуальными, достоверными, точными.



Свойство **ценность** зависит от полезности конкретных объектов и задач, которые можно решать с помощью полученной информации.

Рассмотрим примеры.

1. В списке учеников, сдавших экзамены, указаны все классы. Для вас будет ценна информация, касающаяся вашей оценки.

2. На уроке информатики вы посмотрели видеофильм о работе в интегрированной среде разработки Code::Blocks. На уроке информатики он имеет ценность, но на уроке истории Казахстана – нет.



В повседневной жизни вы продолжительно используете компьютер, мобильные устройства. Подумайте и критически оцените, насколько высок риск для здоровья человека, связанный с работой за компьютером.



В настоящее время подростки все больше времени проводят за компьютером, у экрана телевизора или за другими видеотерминалами, ведут малоподвижный образ жизни. Как это связано со здоровьем? Во-первых, такое времяпровождение ведет к снижению зрения, слабости мышечной ткани, искривлению позвоночника из-за неправильного положения тела.

Во-вторых, так как работа за компьютером является интеллектуальным трудом, то основная часть нагрузки приходится на **центральную нервную систему (ЦНС)**. Поэтому длительная работа за компьютером

приводит к *головным болям*. Это связано с постоянным напряжением лицевых мышц.

В-третьих, часто у подростков возникают *расстройство внимания и концентрации* как следствие *хронического переутомления*. Кроме того, при длительной работе за компьютером могут появиться шум в ушах, головокружение, тошнота.



На рис. 1.2 указаны факторы риска и меры предосторожности. Вам необходимо правильно их систематизировать. Кроме того, вспомните эргономику и правила работы за компьютером из предыдущих классов. На что еще необходимо обратить внимание и какие меры принять, чтобы сберечь здоровье? Дополните информацией.



Рис. 1.2. Факторы риска, влияющие на пользователей компьютеров, и меры предосторожности

На просторах глобальной сети Интернет размещена не только полезная информация, но и негативная, вредная, что приводит к множеству проблем у подрастающего поколения. Для контроля и дозирования информации в Казахстане 2 июля 2018 г. был принят Закон «О защите детей от информации, причиняющей вред их здоровью и развитию».

В закон входит введение законодательных гарантий и организационно-правовых механизмов защиты детей от информации, причиняющей вред их физическому и психическому здоровью, нравственному, духовному, психическому и социальному развитию.



Знание и понимание

1. Перечислите важные свойства информации на русском, казахском и английском языках. Приведите примеры актуальной, точной, достоверной и ценной информации.
2. Какая информация является актуальной?
3. Вы получили сообщение, о том, что надо встретить родственника на вокзале. Однако в сообщении была допущена ошибка в дате приезда. Какую информацию вы получили?
4. Составьте предложения, отразив в них сведения о свойствах информации, представленных в левой колонке. Заполните правую колонку в таблице.

Свойства информации	Сформулируйте предложения
Недостоверной, но актуальной информации	
Неактуальной, но достоверной информации	
Достоверной, но бесполезной информации	
Актуальной, но непонятной информации	
Ценной, но недостоверной информации	

5. Насколько высок уровень риска для здоровья учащихся при длительной работе за компьютером и мобильными устройствами?
6. Составьте и запишите свою версию рассказа, употребив все предложенные ключевые термины по теме.



Применение

Анализ

7. Приведенные примеры сопоставьте с реальной ситуацией и определите свойства информации.
 - а) Учащимся 9 класса дали задание по подсчету количества информации за 5 часов непрерывного чтения при средней длине слова, равной 10 буквам, и скорости чтения, составляющей 160 слов в минуту. Но с формулой подсчета количества информации учитель познакомил учащихся только на следующий день.
 - б) Для написания самостоятельной работы вы скопировали из Интернета текст. При вставке текста в документ текстового процессора Word вы обнаружили информацию в виде двоичного кода.

- в) В г. Нур-Султан летом теплее, чем в Алматы.
 г) Конакасы – казахский обычай, связанный с угощением гостя чем-нибудь вкусным.
 8. Приведите примеры информации, которая в конкретной ситуации будет: актуальной (неактуальной), точной (неточной), достоверной (недостоверной), ценной (неценной), и заполните таблицу.

Свойства информации

Актуальная		Неактуальная	
Точная		Неточная	
Достоверная		Недостоверная	
Ценная		Неценная	



Синтез

Оценивание



9. Классифицируйте информацию в таблице. Какие из свойств в предложенных сведениях выполняются или не выполняются? Отметьте знаками «+» – выполняется свойство, «-» – не выполняется: 1 – актуальность, 2 – точность, 3 – достоверность, 4 – ценность.

Информация	1	2	3	4
Чтение прошлогодних газет				
Сейчас идет дождь и снег				
В прямоугольном треугольнике квадрат гипотенузы равен сумме квадратов катетов				
Номер телефона «Скорой помощи» – 101				
Акбота живет в доме под номером А				
Белый медведь живет на Северном полюсе				
Экзамен по информатике начнется 5 июня 2018 года ровно в 9.00				

10. Объедините информацию по технике безопасности и эргономике о рисках, связанных с работой за компьютером. Сделайте вывод о полезности и актуальности профилактических мер по устранению негативного воздействия компьютера на человека.

11. Создайте презентацию «Польза и вред компьютера для человека».

Польза

- Создание текстовых, видео-, графических документов
- Электронная почта
- Видеочаты
- Хранение в облачных сервисах
- Просмотр фильмов
- Музыка
- Справочники, энциклопедии
- Переводчики
- Дистанционное обучение
- ...

Вред

- Зрение
- Нарушение осанки
- Психологические и нервные расстройства
- Электромагнитное излучение
- Виртуальное общение
- Отсутствие цензуры
- Массовая зависимость
- Стресс
- Киберпреступность
- ...

12. Проект «Кроссворд – проверь свои знания».

В предлагаемом проекте кроссворд служит для проверки знаний по пройденным темам информатики не только того, кто его разгадывает, но также и для того, кто его создает.

Планируемый результат. Создание и оформление тематического кроссворда в электронной таблице Excel.



Кроме перечисленных ранее важных свойств информации, можно привести другие ее свойства. Например:

Объективность и субъективность информации – свойства, зависящие от человеческого фактора. Информация объективна, если она не зависит от чьего-либо мнения, суждения. Например, сообщение «На улице холодно» – субъективная информация, а сообщение «На улице минус 30°C» – объективная.

Объективную информацию можно получить с помощью исправных датчиков, измерительных приборов.

Полнота – это свойство информации, если ее достаточно для понимания и принятия правильных решений. Неполнота информации сдерживает принятие решений или может повлечь ошибки. Например, историческая информация никогда не бывает полной, и ее полнота уменьшается по мере удаленности от нас исторической эпохи.

Адекватность – степень соответствия реальному объективному состоянию дела. В свойстве адекватность всегда присутствует большая или меньшая степень неопределенности. Степень адекватности инфор-

мации реальному состоянию объекта влияет на правильность принятия решений человеком.

Приведем пример.

Вы хотите продолжить образование после 9 класса в колледже по направлению «Программирование». Из бесед с друзьями вы получили сведения, что подобную подготовку можно получить в разных колледжах Казахстана. Полученная информация не адекватна реальному состоянию дел. Для того чтобы получить более достоверные сведения, вы покупаете «Справочник для поступающих в колледжи Казахстана», из которого получаете исчерпывающую информацию.

В этом случае информация, полученная вами из справочника, адекватно отражает направление обучения в колледжах и помогает вам определиться с окончательным выбором.

1.2. Совместная работа с документами

Вы научитесь:

– осуществлять совместную работу с документами с использованием облачных технологий.

Ключевые слова

Облачные технологии
Совместная работа

Бұлттық технологиялар
Бірлесіп жұмыс істеу

Cloud computing
Collaboration



Дайте общую трактовку ключевых слов и предположите, как они будут фигурировать в последующем тексте по теме.



Облачные технологии (вычисления) (на английском языке *cloud computing*) – это вычислительный ресурс для электронного хранения, обработки и совместной работы с данными в сети Интернет.

Облачные технологии являются важнейшими технологиями будущего и широко применяются в образовании. В качестве примеров можно назвать следующие облачные технологии:

- электронный журнал;
- электронный дневник;
- электронное портфолио;
- личный кабинет ученика;
- личный кабинет учителя;
- интерактивная приемная;
- тематические форумы.

Для выполнения конкретных учебных операций можно использовать инструментарий, который обеспечивает выполнение обработки текстов, составление таблиц, тестов, анкет, презентаций и др. На рис. 1.3 приведены инструментарии для применения в образовании.

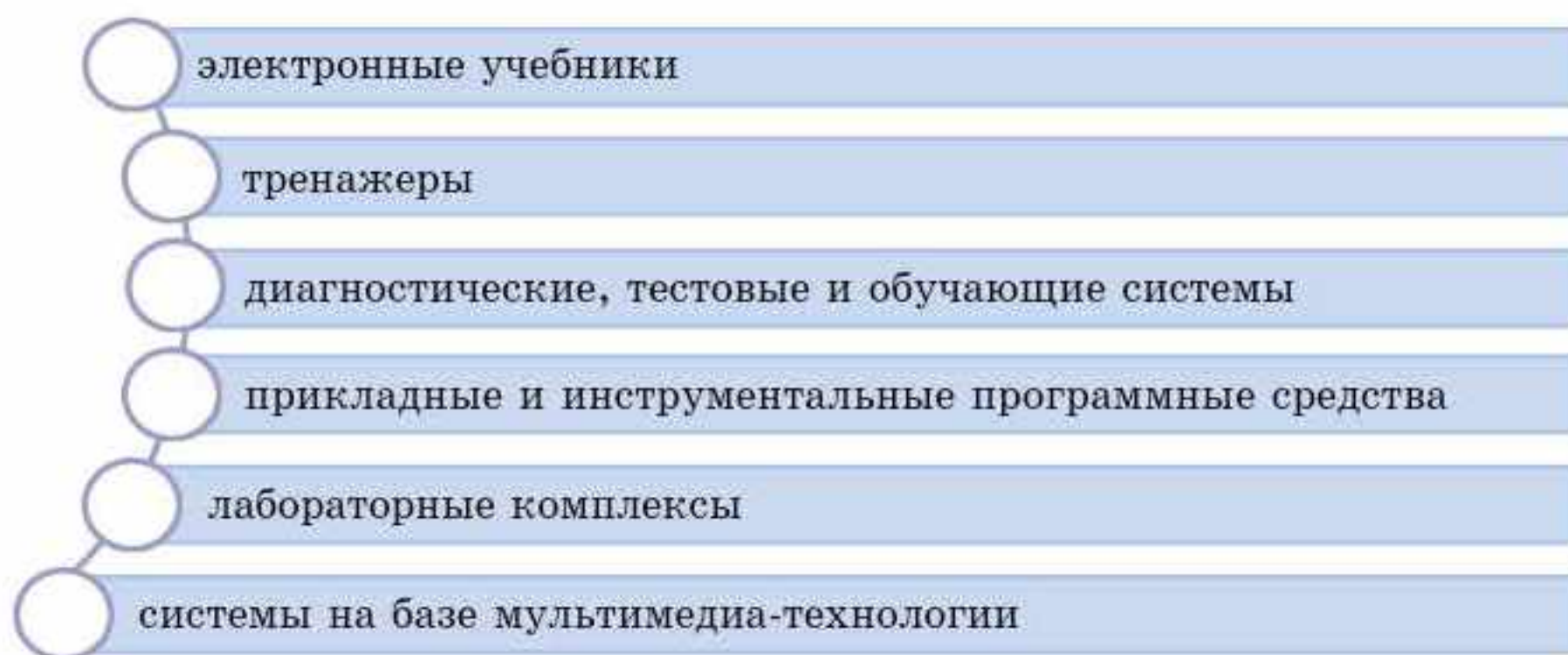


Рис. 1.3. Инструментарии в образовании

Для работы с сервисами Google (рис. 1.4) необходимо создать аккаунт. С помощью него можно войти во все сервисы.



Создайте аккаунт Google на странице регистрации **mail.google.com** или **gmail.com** для совместной работы над документом в облачных технологиях.





Рис. 1.4. Сервисы Google

Шаг за шагом

Нажмите ссылку *Создать аккаунт*. В открывшемся окне заполните поля (рис. 1.5).

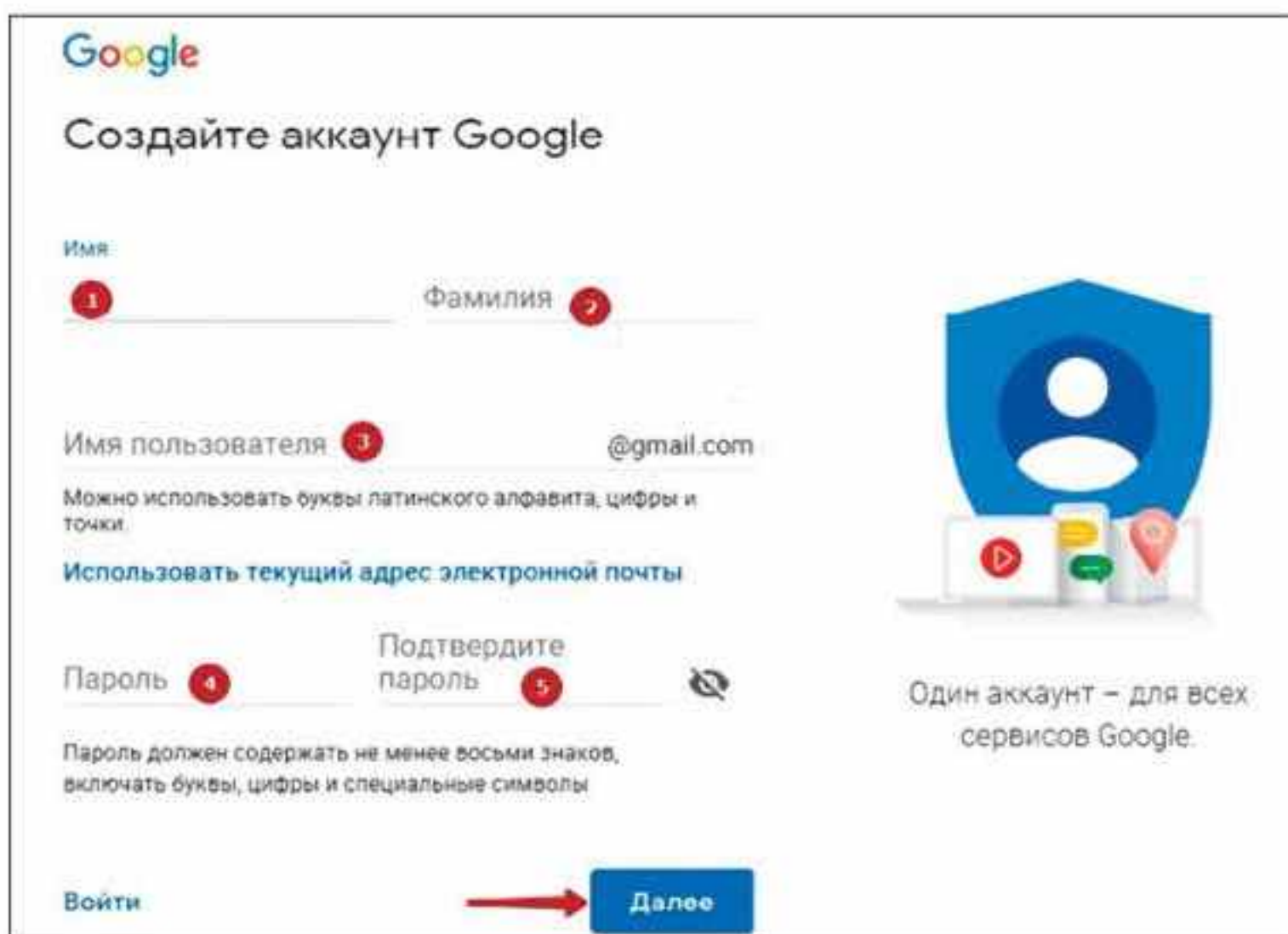


Рис. 1.5. Создание аккаунта Google

Созданные *имя пользователя* и *пароль* для входа вы сможете использовать в любом сервисе Google.

После создания аккаунта в Google можно использовать **Google Диск** (Google Drive) – облачное хранилище для файлов (рис. 1.6).

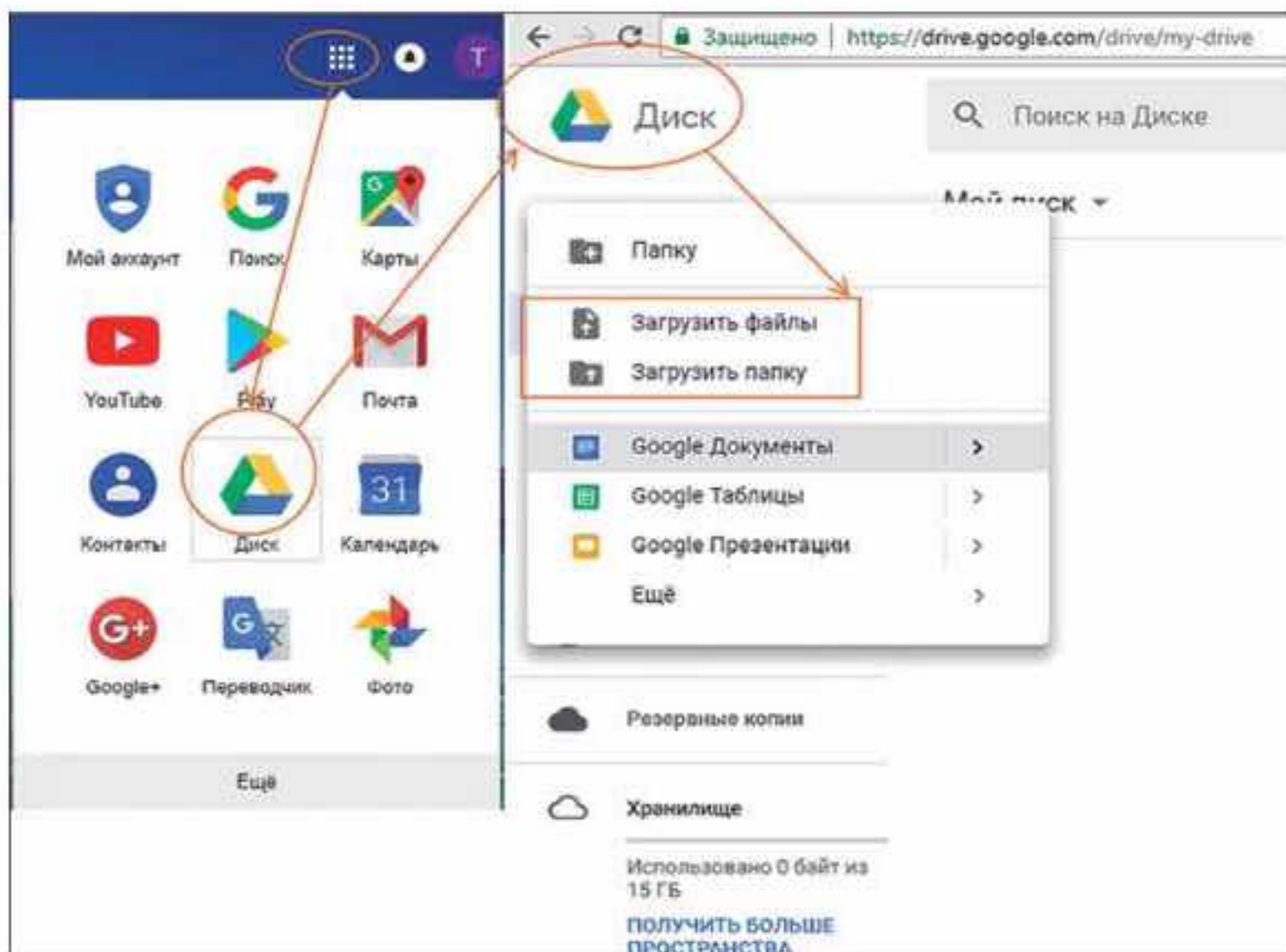


Рис. 1.6. Облачное хранилище файлов в Google Диск

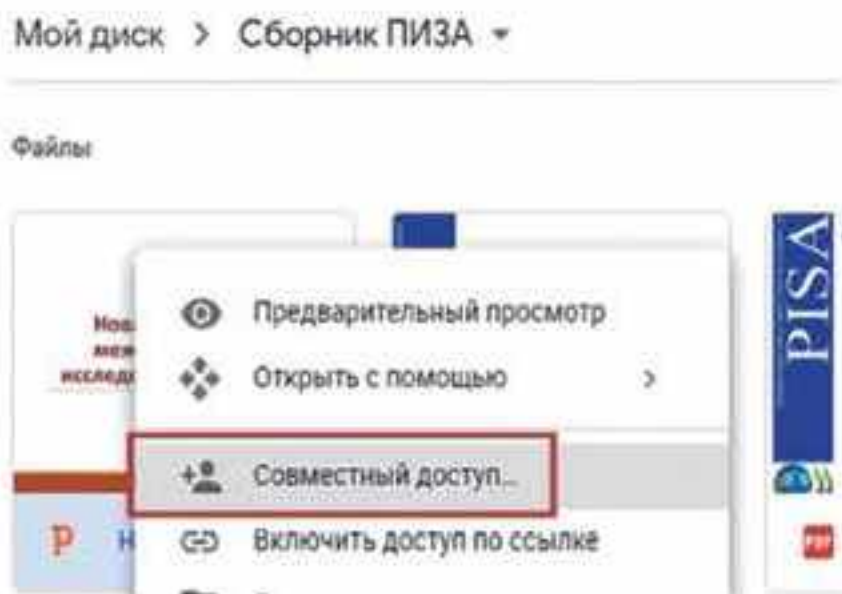



Рис. 1.7. Совместный доступ к документу

Чтобы открыть доступ к файлу или папкам, выполните следующие действия:

1. Откройте **Google Диск**, **Google Документы**, **Google Таблицы** или **Google Презентации** на компьютере.
2. Выберите нужный файл, щелкнув правой кнопкой мыши на файле.
3. В контекстном меню выберите команду **Совместный доступ** (рис. 1.7).

Можно предоставить доступ к файлам другим способом. Выберите **Настройки доступа** или нажмите на значок **Открыть доступ к объекту** .

4. В окне **Уровни доступа** измените доступ к документу.

5. Открыть доступ вы можете, выбрав тот или иной параметр (рис. 1.8):

– **Редактировать** могут все, у кого есть ссылка.

– **Комментировать** могут все, у кого есть ссылка.

– **Просматривать** могут все, у кого есть ссылка.

6. Сохраните выбранный параметр.

7. Все соавторы получают совместный доступ для редактирования (рис. 1.9).

Можно совместно работать над документами, презентациями или таблицами Google всем, кому открыт доступ. Все исправления при совместной работе с документами будут видны всем соавторам и на всех устройствах в режиме реального времени.

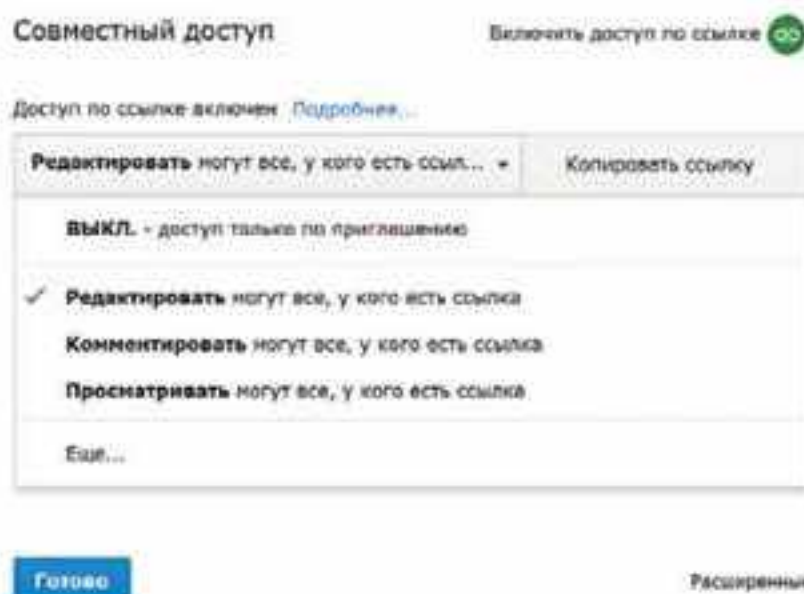


Рис. 1.8. Выбор параметра

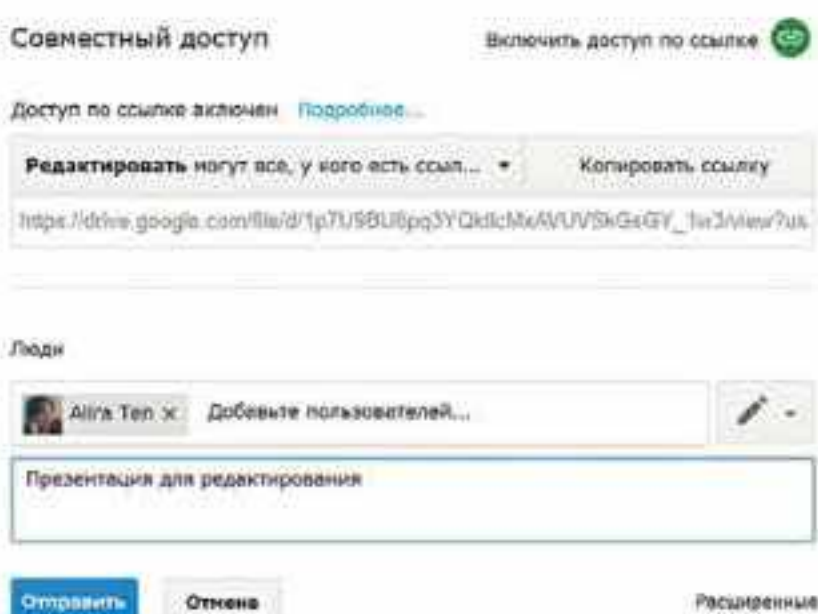


Рис. 1.9. Совместный доступ для редактирования



Знание и понимание



1. Как называется услуга по размещению оборудования клиента на территории провайдера?
2. Приведите пример использования облачных технологий в образовании.
3. Перечислите инструментарию, которые применяются для выполнения конкретных учебных операций.
4. Какие наиболее популярные облачные провайдеры вам известны?
5. Почему сервисы Google получили большую популярность?

6. Каким образом вы можете получить совместный доступ к документу?
7. Составьте и запишите свою версию рассказа, употребив все предложенные ключевые термины по теме.



Применение

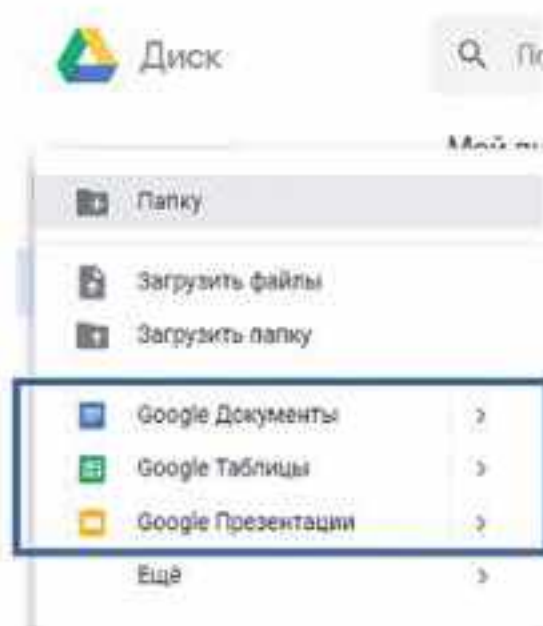
Анализ



8. Создайте документ (таблицу, презентацию) Google.

– Перейдите в **Мой Диск** (или <https://docs.google.com/document/> <https://docs.google.com/presentation/> <https://docs.google.com/spreadsheets/>).

– Нажмите кнопку **Google Документы/Google Таблицы/ Google Презентации** (см. рис. ниже). Недавно созданные документы по умолчанию называются **Новый документ/Новая таблица/Презентация без названия**.



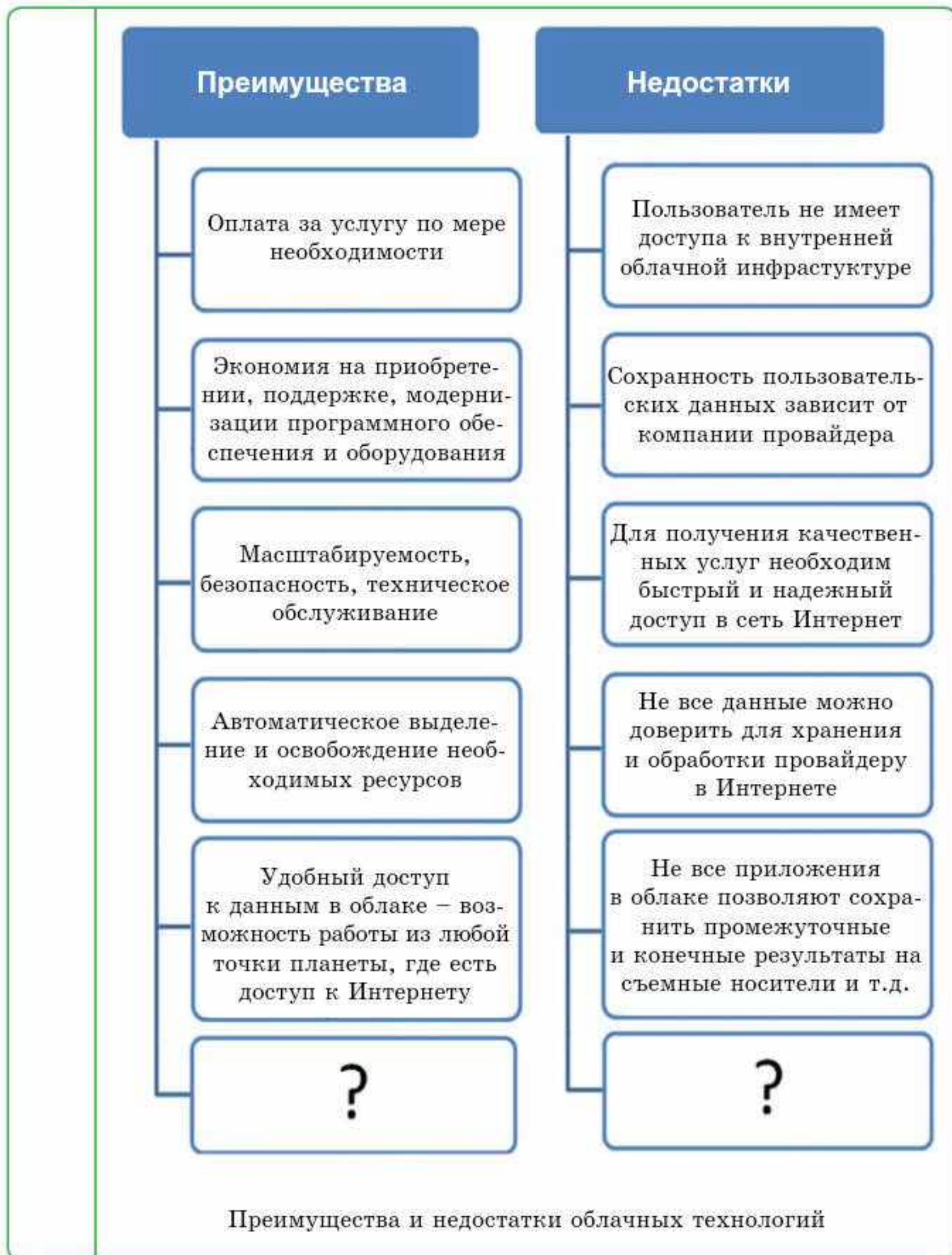
– Чтобы изменить название, в меню **Файл** выберите команду **Переименовать...** Затем введите и подтвердите новое название. Эту же операцию можно выполнить, нажав на название документа, которое отображается в верхней части страницы.

Длина названия не может превышать 255 символов.

Интерфейс окна аналогичен текстовому процессору Word (электронной таблице Excel, Microsoft Power Point).

– Откройте доступ к файлу для совместного комментирования ваших одноклассников.

9. Возможно, вы уже использовали облачные технологии для хранения файлов, фотографий, создания сайтов, сообществ и т. п. Какие, по вашему мнению, существуют преимущества и недостатки облачных технологий? Дополните рисунок «Преимущества и недостатки облачных технологий».





Синтез

Оценивание

10. Рассмотрите рисунок «Доступ по ссылке». Каким образом вы откроете доступ к файлу по ссылке?



Совместный доступ

Включить доступ по ссылке 

Доступ по ссылке включен. [Подробнее...](#)

Просматривать могут все, у кого есть ссы... ▼ Копировать ссылку

https://drive.google.com/file/d/1p7U9BU8ppq3YQkicMzAVUVSkGzGY_1w3Mew?usp=sharing

Люди

Введите имена или адреса эл. почты...

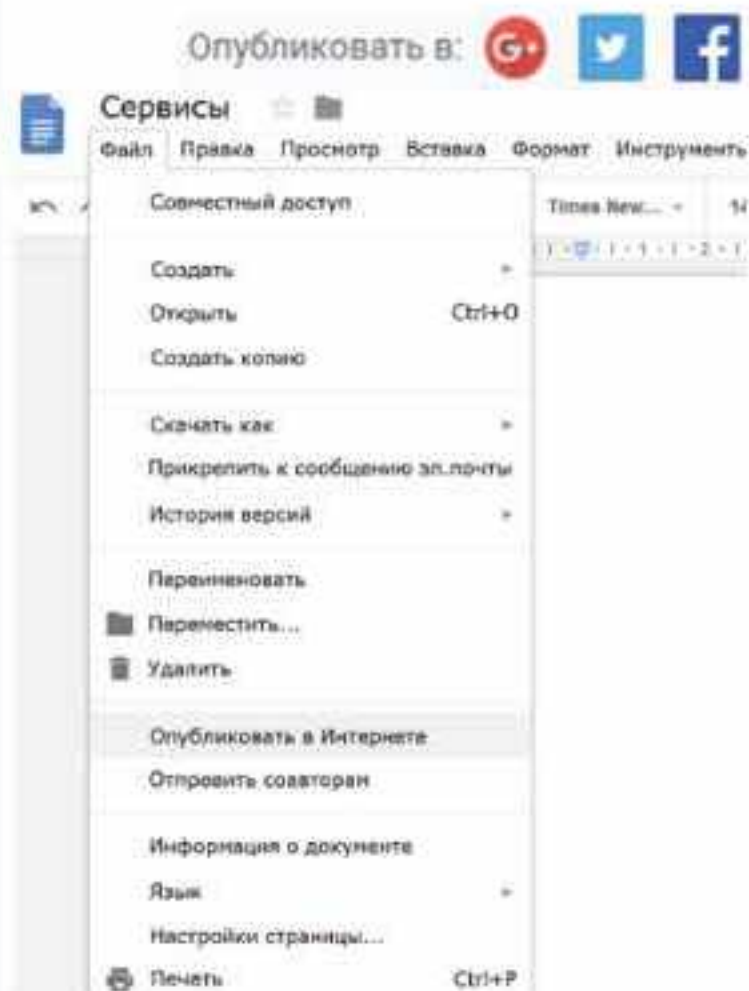
Доступ получили: Alra Tep

Пользователи, у которых есть право просмотра этого файла, увидят комментарии и советы. [Подробнее...](#)

Готово

Расширение

Доступ по ссылке



Вкладка **Файл**

2-я группа. Создайте презентацию класса в Google Презентации, одновременно работая с группой пользователей с помощью функции совместного редактирования.

11. Вы хотите опубликовать в Интернете файл из **Google Документов, Таблиц** или **Презентаций**, чтобы его копия стала доступна вашим друзьям.


Каким образом можно опубликовать в Интернете документ для ваших друзей (см. рис. «Вкладка **Файл**»)? Как вы поделитесь ссылкой для совместного доступа (только соавторам) через социальные сети: Google+, Twitter Facebook?

12. Творческая работа совместного редактирования таблицы, презентации, формы.

1-я группа. Создайте список класса в Google Excel, одновременно работая с группой пользователей с помощью функции совместного редактирования.

3-я группа. Создайте форму «Социальный опрос об использовании мобильных телефонов в обучении» в Google Формы, одновременно работая с группой пользователей с помощью функции совместного редактирования.

Для того чтобы пользователям запретить скачивать, распечатывать и копировать файл, необходимо выполнить следующие действия:

1. Выберите **Настройки доступа** или нажмите на значок  **Открыть доступ к объекту**.
2. В открывшемся окне в правом нижнем углу нажмите **Расширенные**.
3. Установите флажок **Запретить скачивание, печать и копирование для пользователей с правами на комментирование и просмотр** (см. рис. «Настройка доступа»).

Настройка доступа

4. Нажмите команду **Сохранить изменения**.
5. Нажмите команду **Готово**.

1.3. Сетевой этикет

Вы научитесь:

– рассуждать о последствиях нарушения этических и правовых норм в сети.

Ключевые слова

Этикет
Сетевой этикет
Правовые нормы

Этикет
Желілік этикет
Құқықтық ережелер

Etiquette
Network etiquette
Legal provisions



Дайте общую трактовку ключевых слов и предположите, как они будут фигурировать в последующем тексте по теме.



Современный Интернет проникает во все грани нашей жизни – социальные, культурные, психологические – и становится универсальной глобальной информационной средой. Широчайшие технические возможности Интернета делают жизнь, работу и общение в социальных сетях (рис. 1.10) реальными.

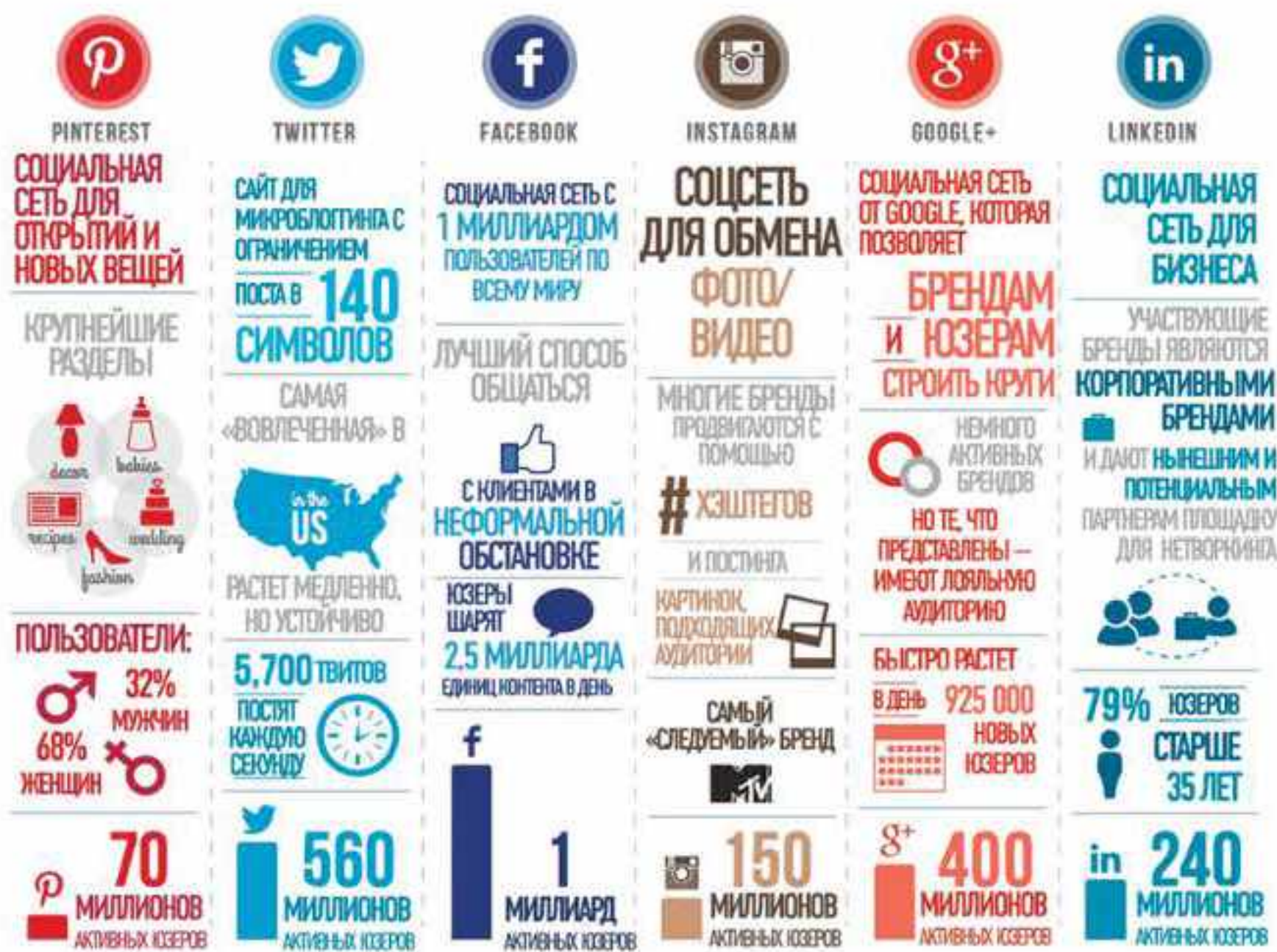


Рис. 1.10. Социальные сети

Однако с ростом популярности Интернета необходимо помнить о правовых нормах и последствиях их нарушения в сети. Для этого очень важно корректно взаимодействовать в сети по определенным правилам поведения и общения, называемых этикетом.



Сетевой этикет – это правила поведения, общения в сети, традиции и культуры интернет-сообщества, которых придерживается большинство пользователей.



Рис. 1.11. Правила сетевого этикета

Сетевой этикет включает базовые правила культурного поведения в сети. Эти правила несложные и не жестко установленные. Большинство из них не носят официального характера, а представляют собой правила хорошего тона, которые приняты в обществе. Некоторые правила записаны в виде формального устава или списка. А некоторые нигде не записаны, но известны многим участникам сообщества и строго соблюдаются.

Рассмотрим ряд правил, позволяющих не нарушать этические и правовые нормы в сети Интернет при работе с электронной почтой, участвовать в телеконференции, использовать различную информацию в сообществах (рис. 1.11).

В соответствии с Законом Республики Казахстан «О доступе к информации» каждый человек может свободно получать и распространять информацию любым не запрещенным законом способом.

В Законе Республики Казахстан «Об информатизации» говорится, что международное сотрудничество в сфере информатизации осуществляется на основе следующих принципов [1]:

1) запрет на использование информационно-коммуникационных технологий во вред человеку, обществу и государству на основе взаимности;

2) запрет на действия, посягающие на информационно-коммуникационную инфраструктуру государства и подрывающие политическую, экономическую, социальную и иные сферы государственной деятельности;

3) предотвращение использования электронных информационных ресурсов, информационных систем или информационно-коммуникационных технологий для преступных целей.

Чтобы обеспечить информационную безопасность, выработайте снижающую риски линию поведения в Интернете:

- Установите четкие правила по использованию Интернета.
- Держите личные данные в секрете, аккуратно обращайтесь с ними.
- Используйте технологию защиты от нежелательной почты (спама), встроенную в операционную систему.
- Установите и регулярно обновляйте антивирусное программное обеспечение.



Поделитесь с одноклассниками правилами сетевого этикета, которых вы придерживаетесь при работе в сети Интернет.

Исследуйте. Какие последствия могут возникнуть при нарушении этических и правовых норм в сети?



Знание и понимание



1. Что такое *сетевой этикет*?
2. Каковы основные правила сетевого этикета при работе с электронной почтой и участии в телеконференции?
3. Какие правила нужно помнить при написании сообщений в сообществе?
4. Какие существуют нарушения этических и правовых норм в сети?
5. Какие основные принципы международного сотрудничества в сфере информатизации отражены в Законе Республики Казахстан «Об информатизации»?
6. Обмениваясь текстовыми сообщениями, пользователи сети для обозначения своих эмоций используют простые картинки, составлен-

ные из нескольких знаков препинания. Их называют «смайлики». Самые распространенные из них:

:-) – улыбка, радость;

:(– печаль, грусть;

:-0 – удивление;

;-) – подмигивание.

Какие смайлики вы еще знаете?

7. Составьте и запишите свою версию рассказа, употребив все предложенные ключевые термины по теме.



Применение

Анализ



8. Напишите электронное письмо на тему «Для чего нужен сетевой этикет?». Отправьте письмо: 1) однокласснику, 2) учителю информатики.

9. Заполните таблицу ответами на вопросы по теме «Сетевой этикет».

Вопросы	Ответы
Как соблюдать сетевой этикет при участии в форуме, телеконференции?	
Можно ли сообщать ваш логин и пароль незнакомым участникам общения в сети?	
Можно посылать большие файлы в письме?	
Почему все сообщение нежелательно писать прописными (большими) буквами?	
Можно ли допускать грамматические ошибки в сообщениях?	
Можно ли использовать чат для рекламы?	
Допустим, вы столкнулись с проблемой этического характера в киберпространстве. Как бы вы поступили в реальной жизни?	

10. Проанализируйте ответы. Какие нарушения будут относиться к этическим, а какие – к правовым нормам в сети?



Синтез

Оценивание



11. Эрик получил зашифрованное письмо. Он должен был дать срочный ответ. Но текст письма содержал сокращения из Интернета на английском языке. Помогите Эрику прочитать следующее письмо:

«Привет! IMHO, я достиг больших результатов в исследовании нашей родословной. AFAIK наш прадедушка был королевским врачом!

BTW, ты не знаешь, кем был наш прадедушка по материнской линии? ASAP мне хотелось бы это срочно узнать, чтобы продолжить свои исследования.

AKA наш общий друг Тамерлан оказался нам дальним родственником по дедушкиной линии. LOL и радуюсь этой приятной новости. Я думаю, что Тамерлан тоже обрадуется, когда получит мое письмо.

WBR, твой брат».

Внизу письма прилагался список сокращений:

IMHO, *имхо* – «*In My Humble Opinion*» – «по моему скромному мнению».

LOL – «*Laugh Out Loud*» – «я громко смеюсь».

BTW – «*By The Way*» – «кстати».

AFAIK – «*As Far As I Know*» – «Насколько я знаю».

NFC – «*No Further Comments*» – «Я все сказал».

WBR – «*With Best Regards*» – «С наилучшими пожеланиями».

ASAP – «*As Soon As Possible*» – «Как можно скорее».

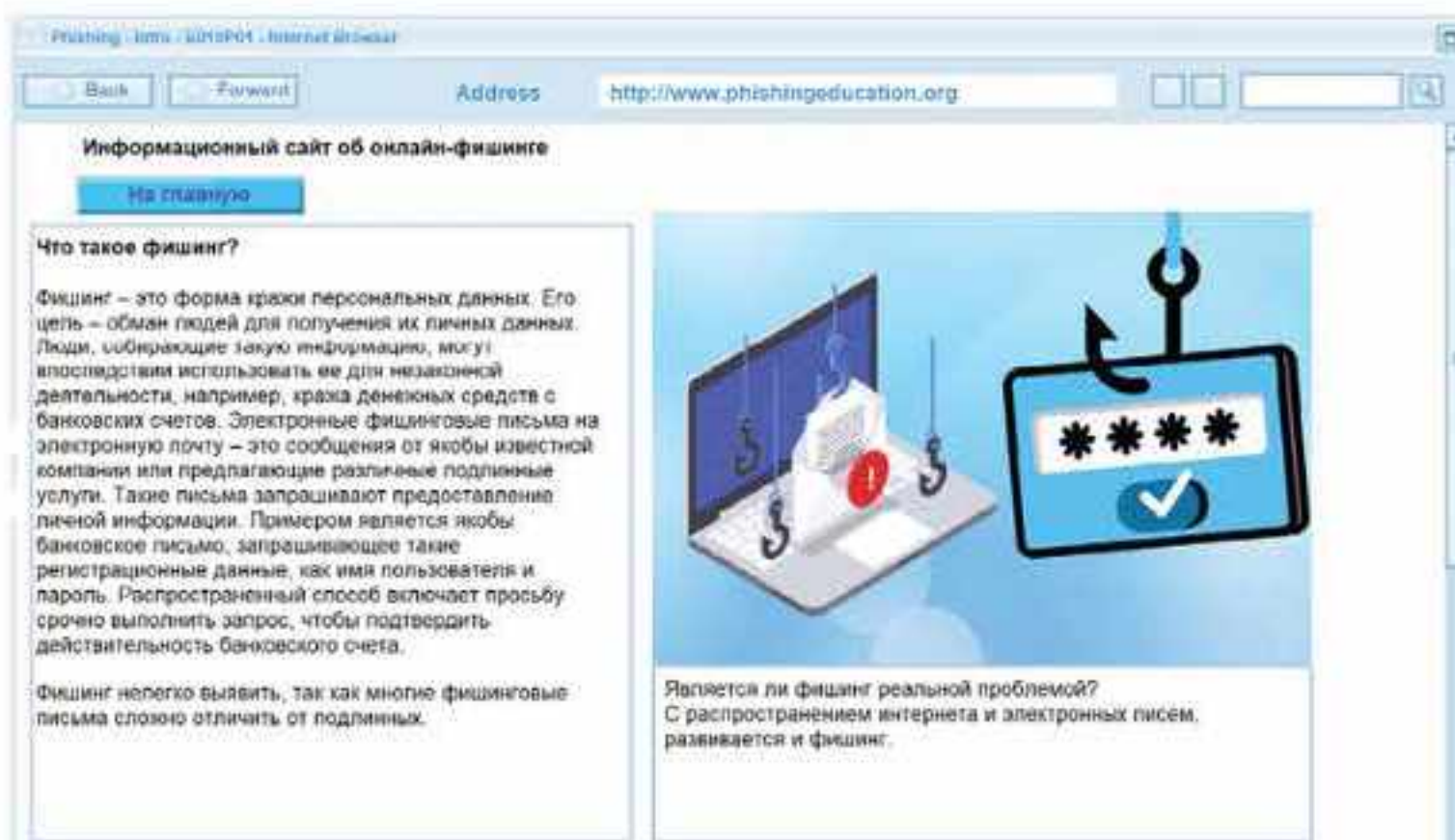
AKA – «*Also Known As*» – «Известный так же, как».

Изучив его, Эрик написал брату ответное письмо, используя сокращения: **NFC, AFAIK, WBR, AKA**.

Как вы думаете, какое письмо составил Эрик брату? Попробуйте написать письмо с помощью сокращенных слов своему другу.

Назовите аргументы «за» и «против» использования сокращений в письмах, сообщениях.

12. Задание «Фишинг» по развитию функциональной грамотности.



Phishing - info - 0019001 - Internet browser

Back Forward Address <http://www.phishingeducation.org>

Информационный сайт об онлайн-фишинге

[На главную](#)

Что такое фишинг?

Фишинг – это форма кражи персональных данных. Его цель – обман людей для получения их личных данных. Люди, собирающие такую информацию, могут впоследствии использовать ее для незаконной деятельности, например, кражи денежных средств с банковских счетов. Электронные фишинговые письма на электронную почту – это сообщения от якобы известной компании или предлагающие различные подлинные услуги. Такие письма запрашивают предоставление личной информации. Примером является якобы банковское письмо, запрашивающее такие регистрационные данные, как имя пользователя и пароль. Распространенный способ включает просьбу срочно выполнить запрос, чтобы подтвердить действительность банковского счета.

Фишинг нелегко выявить, так как многие фишинговые письма сложно отличить от подлинных.

Является ли фишинг реальной проблемой?
С распространением интернета и электронных писем, развивается и фишинг.

Материал, представленный в данном блоке, касается общей проблемы, возникающей как результат в мире, богатом технологиями: фишинг, попытка обмануть людей с целью получения их личной информации, чтобы использовать ее в криминальных целях. Информационный сайт об онлайн-фишинге основан на подлинных материалах веб-сайтов, объясняющих эту проблему, и дает советы читателям о том, как выявить фишинг и не стать его жертвой.

Вопрос 1. Вы на главной странице Информационного сайта об онлайн-фишинге. Какое из нижеприведенных утверждений, согласно информации, представленной на данной странице, свойственно фишинговым электронным письмам?

- А. В нем запрашивается личная информация.
- Б. Оно содержит нежелательную рекламу.
- В. Оно предлагает подлинные услуги.
- Г. Оно приходит от известной компании.

Вопрос 2. Сколько фишинговых писем рассылается в среднем по всему миру в течение месяца?

- А. 1200.
- Б. Более 6 миллиардов.
- В. Около 25 000.
- Г. 55 000.

Вопрос 3. Какой из следующих обманных приемов объясняется на странице «Выявление фишинга»?

Что такое фишинг?
 Фишинг – это форма кражи персональных данных. Его цель – обман людей для получения их личных данных. Люди, собирающие такую информацию, могут впоследствии использовать ее для незаконной деятельности, например, кража денежных средств с банковских счетов. Электронные фишинговые письма на электронную почту – это сообщения от якобы известной компании или предлагающие различные подлинные услуги. Такие письма запрашивают предоставление личной информации. Примером является якобы банковское письмо, запрашивающее такие регистрационные данные, как имя пользователя и пароль. Распространенный способ включает просьбу срочно выполнить запрос, чтобы подтвердить действительность банковского счета.
 Фишинг нелегко выявить, так как многие фишинговые письма сложно отличить от подлинных.

Типичные особенности фишингового письма:
 № 1. Электронный адрес отправителя (см. далее)
 № 2. Приветствие в письме (см. далее)
 № 3. Запрос личной информации (см. далее)
 № 4. Ссылка в электронном письме (см. далее)
 Ссылки в письме могут быть фальшивыми. Ссылка может выглядеть так «www.BankNorthside.com», но привести вас на другой сайт. Она может привести вас на сайт, который является фальшивой версии сайта, на который они ссылаются.

Уважаемый клиент банка Нортсайд!
 Это письмо было выслано сервером банка Нортсайд, чтобы подтвердить ваш электронный адрес.
 Вы должны выполнить верификацию, пройдя по ссылке ниже. Введите номер вашей дебетовой/кредитной карты и пароль в указанных полях.
 Этот процесс необходимо выполнить для вашей защиты, так как некоторые из наших клиентов больше не имеют доступа к своим электронным адресам, и мы должны их подтвердить.
 Для подтверждения вашего электронного адреса и получения доступа к своему банковскому счету перейдите по ссылке ниже:
https://web.BankNorthside.com/au/announcements-mail_verify.asp

А. В письме получателя просят пожертвовать деньги на благотворительность.

Б. Фишинговые электронные письма устанавливают шпионскую программу на компьютер пользователя.

В. Автор письма прилагает фальшивую ссылку на фальшивый веб-сайт.

Г. Письмо сообщает о фальшивом выигрыше.

13. Примените прием критического мышления Таск-анализ по теме «Сетевой этикет». ТАСК – это сокращение для слов: Тезис – Анализ – Синтез – Ключ. На десять последовательно заданных вопросов рационально ответьте и занесите ответы в специально сконструированную ниже таблицу. Здесь мы снова встречается с культурой вычленения тезисов и антитезисов текста.

Вопрос	Ответ
1. Какая тема обсуждается?	
2. Каково основное утверждение по теме?	
3. Сформулируйте контрутверждение. Что скорее всего выскажет оппонент в защиту либо для опровержения данного утверждения?	
4. Что поддерживает основное утверждение и контрутверждение? Перечислите эти доводы в отдельных колонках.	
5. Содержит ли этот текст непонятные, сложные или «перегруженные» слова и выражения? Если да, то найдите и поясните их.	
6. Проведите оценку защиты утверждения и контрутверждения. Определите спорные выводы, отвлеченные моменты, ошибочные заключения и другие слабые места спора.	
7. Видите ли вы какие-либо допущения, ценности или идеологическое влияние в основном утверждении или доводах в его защиту? Найдите их и укажите, насколько они влияют на справедливость утверждения.	
8. Изложите свое утверждение полностью в следующей форме: <i>Несмотря на то, что...</i> (укажите контрутверждение либо один из самых сильных доводов в его защиту), <i>...</i> (основное утверждение), <i>поскольку...</i> (главные причины, побуждающие верить в истинность основного утверждения).	

9. Является ли полный тезис спорным, но в то же время приемлемым для защиты, либо неприемлемым, либо слишком сложным для принятия?	
------------------------------------------------------------------------------------------------------------------------------------	--

10. В случае необходимости пересмотрите ваше основное утверждение и повторите все стадии TASK.	
------------------------------------------------------------------------------------------------	--



Оффтопик – это форумы или чаты, которые посвящены обсуждению только определенной темы, например программирование.

Оверквотинг (*overquoting*) – избыточное цитирование письма, которое является вредным. По мере возможности его следует избегать.

Флейм – (спор ради спора) когда разговор друг с другом переходит в спор, а затем в ожесточенное ругательство.

Человек «разжигает» флейм, когда:

- в процессе разговора переходит на личности;
- допускает оскорбления личного, национального, религиозного, полового или профессионального характера;
- ведет дискуссию с эмоциями, грубо, неуравновешенно, провоцируя скандал.

Никогда не поддерживайте флейм, игнорируйте скандалистов.

Флуд – это поток сообщений, которые не несут в себе никакого смысла. Применяется для привлечения к себе внимания. При флуде замедляется загрузка страниц, увеличивается количество ненужной информации и увеличиваются затраты.

Итоговые задания по разделу «Работа с информацией»

1. Для того чтобы человек мог правильно ориентироваться в окружающем мире, информация должна быть:

- a) достоверной;
- b) полезной;
- c) точной;
- d) актуальной;
- e) все выше перечисленное.

2. Информацию, которая своевременна, важна для настоящего времени называют:

- a) ценной;
- b) актуальной;
- c) достоверной;
- d) недостоверной;
- e) понятной.

3. Информацию, содержащую сведения, отражающие истинное положение дел, называют:

- a) неактуальной;
- b) точной;
- c) актуальной;
- d) достоверной;
- e) ценной.

4. Информацию, полезную для решения конкретных задач, называют:

- a) ценной;
- b) неточной;
- c) актуальной;
- d) точной;
- e) достоверной.

5. Информацию, очень близкую к реальному состоянию объекта, процесса, явления, называют:

- a) достоверной;
- b) точной;
- c) недостоверной;
- d) актуальной;
- e) ценной.

6. Установите соответствие между свойством информации и его описанием.

- | | |
|-------------------|--------------------------------------|
| 1) достоверность; | a) язык понятен получателю; |
| 2) полнота; | b) правильность, непротиворечивость; |
| 3) понятность; | c) вовремя, в нужный срок; |
| 4) ценность; | d) имеются все необходимые данные; |
| 5) актуальность. | e) полезность, важность, значимость. |

7. Выберите из списка только актуальную информацию, если вы выпускники 9 класса и впереди у вас экзамены:

- a) информация о датах проведения экзаменов;
- b) информация о зачислении в 1 класс;
- c) информация о вступительных экзаменах в институт;
- d) информация о невыполненной домашней работе в 5 классе;
- e) информация о ближайших распродажах в магазине.

8. Вечером, когда вы уже вернулись из школы, неожиданно позвонил одноклассник и предупредил, что расписание уроков на завтра изменилось, и продиктовал новое. Какая для вас это информация:

- a) достоверная;
- b) точная;
- c) актуальная;
- d) недостоверная;
- e) ценная?

9. Вы собирались на прогулку и по телевизору увидели заставку: ☼ +22, а на улице идет дождь. Какая для вас это информация:

- a) достоверная;
- b) неценная;
- c) полезная;
- d) актуальная;
- e) недостоверная?

10. Прочитав книжку по истории, вы узнали, что у египтян не было алфавита, и вместо букв они использовали рисунки-иероглифы. В школе вы как раз изучали Древний Египет. Какая для вас это информация:

- a) актуальная;
- b) ценная;
- c) нужная;
- d) достоверная;
- e) точная?

11. Определите, о чем идет речь: «...вычислительный ресурс для электронного хранения, обработки и совместной работы с данными в сети Интернет»:

- a) вычислительная программа;
- b) облачные технологии;
- c) вычислительные технологии;
- d) нет верного ответа;
- e) калькулятор.

12. Перечислите преимущества облачных технологий:

- a) возможность работы из любой точки планеты;
- b) нет доступа к внутренней облачной инфраструктуре;
- c) сохранность данных зависит от компании провайдера;
- d) экономия на приобретении, поддержке, модернизации программного обеспечения и оборудования;
- e) оплата за услугу по мере необходимости.

13. Перечислите недостатки облачных технологий:

- a) оплата за услугу по мере необходимости;
- b) сохранность данных зависит от компании провайдера;
- c) возможность работы из любой точки планеты;
- d) нет доступа к внутренней облачной инфраструктуре;
- e) экономия на приобретении, поддержке, модернизации программного обеспечения и оборудования.

14. Установите соответствие.

- | | |
|---------------------|----------------------------------------------------------------------|
| 1) редактирование; | a) можно оставлять комментарии и предлагать изменения; |
| 2) комментирование; | b) нельзя редактировать файл или изменять настройки доступа; |
| 3) просмотр. | c) можно только открывать файл и просматривать; |
| | d) можно редактировать, принимать и отклонять предложенные поправки; |
| | e) можно менять настройки доступа. |

15. Выберите, те высказывания, которые относятся к правилам сетевого этикета:

- a) привлекайте внимание к своей персоне;
- b) уважайте время и возможности других;
- c) придерживайтесь таких же стандартов поведения, что и в реальной жизни;
- d) отправляйте сообщения, не имеющие смысловой нагрузки;
- e) указывайте автора при использовании информации других лиц.

16. Что такое флуд:

- a) массовая рассылка рекламы;
- b) словесная война;
- c) реклама на сайте;
- d) сообщение на сайте/форуме, не несущее смысловой нагрузки;
- e) неинформативный ответ?

17. Что такое флейм:

- a) сообщение на сайте/форуме, не несущее смысловой нагрузки;
- b) неинформативный ответ;
- c) словесная война;
- d) неактивный аккаунт;
- e) массовая рассылка рекламы?

18. Что такое оверквотинг:

- a) избыточное цитирование вредных сообщений;
- b) сообщение, написанное с орфографическими ошибками;
- c) сообщение, написанное с пунктуационными ошибками;
- d) сообщение на сайте/форуме, не несущее смысловой нагрузки;
- e) неинформативный ответ?

19. Что является хорошим тоном общения в Интернете (из нижеперечисленного):

- a) изучение традиций форума/сайта;
- b) спор, словесная война;
- c) создавать темы, не имеющие отношение к тематике форума;
- d) не задавать личные вопросы;
- e) обязательно указывать тему письма?

20. Что такое оффтоп:

- a) создание тем, не имеющих отношение к тематике форума;
- b) создание тем, имеющих отношение к тематике форума;
- c) создание оскорбительных сообщений;
- d) создание сообщений, написанных с орфографическими ошибками;
- e) сообщение на сайте/форуме не несущее смысловой нагрузки?

Проекты к разделу I

1. Проект «Влияние автомобильных выхлопов на экологию Казахстана» (совместная работа с документами в Google).
2. Исследование «Информация в новостных лентах» (свойства информации).
3. Исследование «Уничтожение лесов Казахстана» (совместная работа с документами в Google).

СИСТЕМА ОСНОВНЫХ

ПОНЯТИЙ РАЗДЕЛА II

ВЫБИРАЕМ

КОМПЬЮТЕР



Раздел II

ВЫБИРАЕМ КОМПЬЮТЕР

2.1. Конфигурация компьютера

Вы научитесь:

– выбирать конфигурацию компьютера в зависимости от его назначения.

Ключевые слова

Конфигурация

Базовая конфигурация

Сбалансированная

конфигурация

Оптимальность

Конфигурация

Базалық конфигурация

Теңгерілген

конфигурация

Оңтайлылық

Configuration

Basic configuration

Balanced

configuration

Optimality



Дайте общую трактовку ключевых слов и предположите, как они будут фигурировать в последующем тексте по теме.



При выборе персонального компьютера надо определить, для решения каких задач он будет предназначен. В зависимости от функционального назначения компьютеры можно условно разделить на несколько групп: офисные, учебные, игровые, домашние, дизайнерские. Кроме того, компьютеры дополнительно классифицируют по уровню производительности, выделяя начальный, средний, высший уровень.



Подумайте и ответьте.

Какие задачи выполняют компьютеры каждой группы?



Для выхода в Интернет, работы с простыми текстовыми документами и прослушивания музыкальных композиций достаточно компьютера

с офисной конфигурацией. Сравнительно небольшой мощности, которой он обладает, вполне достаточно для очерченного круга задач, а среди его достоинств такие ценные качества, как бесшумность работы и минимальный уровень энергопотребления.

Домашний компьютер мультимедийной направленности позволит работать в графическом редакторе, просматривать видео и слушать музыку в хорошем качестве, обеспечивая хорошую производительность.

Игровой и дизайнерский персональный компьютер оценят, соответственно, геймеры и дизайнеры, которым требуется максимально высокая производительность компьютера для работы со сложными графическими приложениями и программами.

Решение данного вопроса напрямую зависит от рационального конфигурирования компьютера.



Конфигурация персонального компьютера – это набор устройств и программного обеспечения, характеристики которых определяют основное направление использования персонального компьютера.

Конфигурация компьютера должна соответствовать классу решаемых на нем задач. В этом смысле нет плохих конфигураций компьютера, есть лишь конфигурации, не соответствующие типу выполняемых на компьютере задач.

Понятие «конфигурация» охватывает компоненты двух видов: *аппаратные* и *программные средства* с необходимым набором их характеристик, параметров и назначений.



Базовая конфигурация компьютера – это минимальный комплект аппаратных и программных средств, достаточный для начала работы на компьютере.



Подумайте и ответьте.

- 1) Какой комплект аппаратных средств составляет базовую конфигурацию компьютера?
- 2) Какие внутренние устройства входят в состав системного блока?
- 3) Назовите аппаратные средства компьютера, представленные на рис. 2.1.





Рис. 2.1. Аппаратные средства компьютера

Кроме того, можно говорить о **сбалансированности** или **несбалансированности** конфигурации компьютеров, ориентированных на решение одного и того же класса задач.



Подумайте и ответьте.

Как вы понимаете сбалансированность и несбалансированность конфигурации компьютера?



Рассмотрим на примере игрового компьютера, что такое *сбалансированная* и *несбалансированная конфигурация*. В компьютерах данной категории основное внимание требуется уделить видеокарте и процессору. Именно их правильное сочетание и делает конфигурацию компьютера сбалансированной.

Если у вас хорошая мощная видеокарта, но слабый процессор, то невозможно будет увеличить скорость обработки кадров. При любых разрешениях экрана она останется низкой.

Это и есть несбалансированность конфигурации компьютера, когда видеокарта справляется со своими задачами и даже простаивает часть времени, а процессор не способен загрузить ее потоком данных с нужной скоростью.

Несбалансированность будет также наблюдаться, если видеокарта малой мощности, а процессор высокопроизводительный. Только в данном случае, наоборот, возможности процессора превышают возможности видеокарты. В результате скорость обработки кадров будет зависеть от разрешения экрана, уменьшаясь по мере увеличения разрешения.



Конфигурацию компьютера можно назвать **сбалансированной** тогда, когда все возможности его процессора, памяти, видеокарты, шин для обмена информацией дают возможность обрабатывать данные максимально быстро, минуя малейшие простои, для каждого типа решаемых задач.



В игровом компьютере мощный процессор, мощная видеокарта и большой объем оперативной памяти. На производительность каких устройств в конфигурации компьютера нам следует обратить внимание, если мы приобретаем его для работы с фото, видео, хранения и прослушивания аудиофайлов?



Кроме сбалансированности, следует обязательно обратить внимание на **оптимальность** конфигурации.

Для каждой категории компьютеров существует оптимальный уровень производительности, и его превышение не принесет никакой пользы.

Выбор оптимального варианта компьютера всегда сопряжен с решением вопроса об экономном расходовании денежных средств. Необходимо найти приемлемое сочетание между расходами и комплектом приобретаемых функциональных устройств компьютера.

Целесообразно рассматривать минимальные требования к аппаратным средствам, исходя из планируемых для решения задач.

Зачем переплачивать за избыточную мощность компьютера, если оптимальной для вас будет конфигурация, при которой все задачи решаются с хорошей скоростью при минимальной стоимости?

При выборе определенной конфигурации компьютера следует обратить внимание на следующие характеристики устройств компьютера:

- 1) тип (модель) процессора (Intel Celeron, Intel Core 2 Duo, AMD Athlon, Athlon 64 X2) и его характеристики (тактовая частота, разрядность);
- 2) объем оперативной памяти (RAM);
- 3) тип и емкость жесткого диска (HDD);
- 4) видеокарта;
- 5) наличие мультимедиа-компонентов (CD-ROM, DVD-R, RW, колонки);

- 6) сетевая плата (LAN), звуковая карта (SD);
- 7) клавиатура (kbd), манипулятор «мышь» (M&P);
- 8) тип монитора и его характеристики (разрешающая способность, размер, яркость, контрастность).



Создайте пары, соединив название комплектующих компьютера на английском и русском языках:
CPU, RAM, HDD, LAN, SD, kbd, звуковая карта, оперативная память, клавиатура, процессор, видеокарта, жесткий диск.



Знание и понимание



1. На какие группы по назначению условно можно разделить все компьютеры? Дайте характеристику компьютеров каждой группы.
2. Что понимают под конфигурацией компьютера?
3. Какими компонентами конфигурации будут отличаться друг от друга два компьютера, если один из них предполагается использовать для работы с текстами, а второй – для работы с графикой?
4. Какие компоненты конфигурации компьютера определяют его быстродействие? Ответ обоснуйте.
5. Какие компоненты конфигурации компьютера выполняют функции кратковременной и долговременной памяти? Приведите их основные количественные характеристики.
6. Определите состав и основные характеристики оборудования компьютера, установленного на рабочем месте в компьютерном классе. Информацию о составе и характеристиках оборудования компьютера представьте, используя в качестве образца приведенную ниже таблицу.

Устройство	Тип устройства	Характеристики	Значение	Где расположено устройство
Процессор	Intel Pentium	1. Тактовая частота	2,4 ГГц	Системный блок, материнская плата
		2. Частота переключения шины МП	800 МГц	

7. Составьте и запишите свою версию рассказа, употребив все предложенные ключевые термины по теме.



Применение

Анализ



8. Определите, к какому виду устройств компьютера относятся приведенные ниже записи, и представьте их в развернутом виде.

- 1) CPU Intel Celeron D 352 3.2 ГГц/ 512К/ 533МГц 775-LGA;
- 2) Genius G-Pen 560 (4.5» x 6», 2000 lpi, 1024 деңгейі, USB);
- 3) Canon CanoScan 5000F(A4 Color, plain, 2400 x 4800dpi, USB 2.0, слайд-адаптер);
- 4) BenQ Digital Camera E53 <Silver> (5.0Мрх, 32-96mm, 3x, F2.8–4.8, JPG, (8-32) Mb SD, 2.5», USB, Li-Ion);
- 5) 17» MONITOR 0.27 LG Matron EZ T710PU;
- 6) 17» MONITOR LG L1770HQ-BF Flatron <Silver> (LCD, 1280x1024, +DVI);
- 7) Epson STYLUS COLOR 680 (A4, 2880dpi, USB).

9. В прайс-листе компьютерной фирмы указана конфигурация компьютера:

IntelCore2Duo – 3,0 GHz/1 Gb/400 Gb/128 Mb GeForcePCX6600/DVD+RW/-RW/ CD-RW(16ЧR, 16ЧW, 8ЧRW/48ЧR, 48ЧW, 32ЧRW) /FDD/ LAN1Gb/ SB/kbd/ M&P/ 19,0''SamsungSyncMaster 970 black (DVI, 1280?1024-6ms, 250cd/m2, 1000:1,1780/1780). Заполните таблицу, используя данную информацию.

Тип комплектующих	Модель в прайс-листе
Процессор	IntelCore2Duo – 3,0GHz



Синтез

Оценивание



10. Скачайте из Интернета прайс-лист любой компьютерной фирмы. Представьте, что вы работаете продавцом-консультантом в магазине компьютерной техники. Подберите по прайс-листу конфигурацию компьютера для:

- ученика 9 класса;
- дизайнера интерьеров помещений;
- домохозяйки;
- монтажера рекламных роликов.

Решение оформите в таблице.

Покупатель	Конфигурация компьютера

11. Соберите компьютер из предложенных устройств. Характеристики модулей компьютера должны быть оптимальными.

- Микропроцессоры: Pentium IV с сокетом 775 LGA, Pentium D с сокетом 775 LGA, Athlon64 с сокетом 939, Celeron с сокетом 478.
- Материнские платы: с сокетом 775, с сокетом 477, с сокетом 939, с сокетом S-AM2.
- Жесткие диски объемом: 120 Гбайт; 200 Гбайт; 300 Гбайт.
- Видеокарты объемом; 128 Мбайт, 256 Мбайт, 512 Мбайт.
- Монитор CRT, 17 дюймов, монитор LCD, 17 дюймов, монитор LCD, 19 дюймов.
- Мышь оптическая (3 шт.), клавиатура (3 шт.)
- Корпус компьютера с блоком питания мощностью: 350 В, 400 В (2 шт.)



Совместимость отдельных устройств с материнской платой

Прежде всего, это относится к совместимости по интерфейсу подключения. Существует несколько различных процессорных интерфейсов, для каждого из которых выпускаются свои модели материнских плат. Для процессоров фирмы Intel, например, использовались интерфейсы Socket 478, Socket 775 LGA, а для процессоров фирмы AMD – Socket A, Socket 754, Socket 969, Socket S-AM2. Поэтому при выборе материнской платы, в первую очередь, следует обращать внимание на ее процессорный интерфейс.

Стандартным интерфейсом для подключения видеокарт на данный момент является шина PCI-Express (PCIe или PCI-E), PCI-Express 16x

и PCI-Express 2.0 – наиболее используемый для подключения дискретных видеокарт. Основное различие между этими версиями состоит в том, что в версии 2.0 была увеличена максимальная пропускная способность до 8 Гбит/с в каждом направлении, а также увеличились возможности энергоподачи до 300 Вт. Для этого на видеокарты устанавливается 2×4-штырьковый разъем питания. PCI-Express реализован в различных версиях, отличающихся пропускной способностью: 1x, 2x, 4x, 8x, 16x и 32x. Видеоинтерфейс PCI-E 16x обеспечивает пропускную способность, равную 4 Гб/с в каждом направлении. Также были реализованы PCI-Exp 8x (в бюджетных SLI- или CrossFire-решениях) и PCI-E 4x (или PCI-Express Lite).

Современная оперативная память обычно имеет тип DDR, DDRII или DDRIII и соответствующие интерфейсы подключения к материнской плате. Иногда на одной материнской плате могут одновременно присутствовать оба этих типа разъемов.

Жесткие диски подключаются по интерфейсам Serial ATA, Serial ATA II и Serial ATA III (обозначаются SATA, SATA II и SATA III). Существуют также переносные жесткие диски, подключаемые по интерфейсу USB.

Кроме того, следует учитывать, что устройства, имеющие одинаковый интерфейс, могут отличаться по пропускной способности, которая измеряется в мегабайтах в секунду или мегабитах в секунду. Надо обращать внимание на то, какую пропускную способность имеет данное устройство и какую пропускную способность обеспечивает выбранная материнская плата. Если они не совпадают, то либо само устройство, либо материнская плата будет работать не в оптимальном режиме, что, в свою очередь, будет влиять на быстродействие всей компьютерной системы в целом.

При комплектации компьютера необходимо также учитывать, что некоторые компоненты могут быть встроены непосредственно в материнскую плату (видеокарты, звуковые карты, сетевые карты). И приобретение дополнительных аналогичных устройств может быть оправдано только в том случае, если они имеют лучшие характеристики, чем интегрированное устройство. Наличие встроенной звуковой карты можно определить по названию кодека, обычно Realtek, а встроенной сетевой карты – по обозначению LAN, после которого обычно указывается пропускная способность в мегабитах в секунду.

2.2. Выбор программного обеспечения

Вы научитесь:

– выбирать программное обеспечение в зависимости от потребности пользователя.

Ключевые слова

<i>Программное обеспечение</i>	<i>Программалық жасақтама</i>	<i>Software</i>
<i>Базовое программное обеспечение</i>	<i>Базалық программалық жасақтама</i>	<i>Basic Software</i>
<i>Системное программное обеспечение</i>	<i>Жүйелік программалық жасақтама</i>	<i>System software</i>
<i>Служебное программное обеспечение</i>	<i>Қызметтік программалық жасақтама</i>	<i>Utility software</i>
<i>Прикладное программное обеспечение</i>	<i>Қолданбалы программалық жасақтама</i>	<i>Application software</i>



Дайте общую трактовку ключевых слов и предположите, как они будут фигурировать в последующем тексте по теме.



Компьютеры невозможно себе представить без комплекса **программного обеспечения (ПО)**. Рассмотрим, по каким критериям выбрать программное обеспечение того или иного типа.



Уровни программной конфигурации:

- **базовое программное обеспечение** (встроено в компьютер);
- **системное и служебное программное обеспечение** (устанавливается в общем пакете);
- **прикладное программное обеспечение** (устанавливается пользователем).



Выбор программного обеспечения (рис. 2.2) делают исходя из тех задач, которые необходимо решить. Это может быть работа с документами, разработка приложений, интернет-развлечения, работа с базами данных, графика, математика, звук, видео и др. Как видим, список можно про-

должать до бесконечности. В любом случае прикладное программное обеспечение компьютера должно базироваться именно на выполнении конкретной функции.

С точки зрения современных компьютерных технологий можно выделить две большие общие группы программного обеспечения: **системное программное обеспечение** (обслуживающее операционную систему) и **прикладное программное обеспечение** (предназначенное для решения конкретных задач). Обе категории находятся в тесном взаимодействии. Прикладное программное обеспечение в большей степени зависит от системного.



Рис. 2.2. Программное обеспечение



Подумайте и ответьте.

Охарактеризуйте системное и прикладное программное обеспечение. Какие программы можно отнести к системному программному обеспечению, а какие – к прикладному?



На рис. 2.3 перечислены качества программного обеспечения. Охарактеризуйте каждое качество, как вы его понимаете.





Рис. 2.3. Качество программного обеспечения

Базовое программное обеспечение – это программное обеспечение, которое встроено в компьютер.



BIOS (Basic Input/Output System) – это набор микропрограмм, которые отвечают за управление всеми компонентами, установленными на материнской плате и подключенными к компьютеру устройствами.



Рис. 2.4. BIOS

BIOS является неотъемлемой составляющей системной платы (рис. 2.4). Поэтому она отнесена к особой категории компьютерных компонентов, занимающих промежуточное положение между аппаратным и программным обеспечением.



Подумайте и ответьте.

Какие еще основные функции, помимо управления компонентами и устройствами, выполняет BIOS?



POST (power-on self-test) – тестирование всего установленного на материнской плате оборудования (за исключением дополнительных плат расширения), проводимое после каждого включения компьютера.

В процессе тестирования оборудования BIOS сравнивает данные системной конфигурации с информацией, хранящейся в CMOS – специальной энергозависимой памяти, расположенной на системной плате. Хранение данных в CMOS поддерживается специальной батареей, а информация обновляется всякий раз при изменении каких-либо настроек BIOS. Именно эта память хранит последние сведения о системных компонентах, текущую дату и время, а также пароль для входа в BIOS.



BIOS обеспечивает начальную загрузку компьютера с последующим запуском операционной системы.

При выходе из строя, повреждении или удалении батарейки все данные в CMOS-памяти обнуляются.

BIOS обеспечивает начальную загрузку операционной системы. В дальнейшем загрузчик ищет и загружает в память код операционной системы и передает ему управление.

Системное и служебное программное обеспечение (устанавливается в общем пакете) – совокупность программ и программных комплексов для обеспечения работы компьютера и вычислительных сетей.

Виды системного программного обеспечения: операционные системы, программы-оболочки, операционные оболочки, драйверы, утилиты.



Подумайте и ответьте.

Приведите примеры программ системного и служебного программного обеспечения.



Основная составляющая всего системного программного обеспечения – **операционная система** (рис. 2.5).



Рис. 2.5. Виды операционных систем



Подумайте и ответьте.

Какие операционные системы вы знаете? Какие из них используются чаще всего? От чего зависит выбор операционной системы?



Первое, на что нужно обратить внимание, – **назначение компьютера**. Если вы планируете использовать его в качестве сервера, нужно устанавливать серверную операционную систему. Если это просто пользовательский компьютер, можно использовать обычную сетевую систему, каковыми сегодня являются практически все однопользовательские операционные системы.

Второй аспект – **конфигурация компьютера**. Если компьютер слабый, следует обратить внимание на операционную систему. Например, Linux без графического интерфейса.

Если компьютер мощный, можно установить любую операционную систему. Начинающему пользователю лучше выбирать систему, в которой легко разобраться.

Прикладное программное обеспечение (устанавливается в общем пакете) – это специализированные приложения, ориентированные на выполнение конкретной задачи. Прикладные программы могут использоваться автономно или в составе программных комплексов, или пакетов.



Подумайте и ответьте.

Какие программы входят в прикладное программное обеспечение? В какой ситуации они необходимы?



Как правило, все пользователи предпочитают иметь набор прикладных программ, который нужен практически каждому. Их называют **программами общего назначения**.

Кроме того, имеется большое количество **прикладных программ специального назначения** для профессиональной деятельности. Их часто называют пакетами прикладных программ.



Подумайте и ответьте.

Какая программа потребуется, если нужно: создать рисунок на компьютере; оформить доклад; произвести начисление заработной платы; посетить сайт; подготовить презентацию для защиты проекта; создать базу данных сотрудников фирмы?



Системы программирования предназначены для создания системного и прикладного программного обеспечения.

Базовые инструменты любой среды программирования одинаковые. Они отличаются только формой представления и базовым набором инструментов в разных средах программирования.



Подумайте и ответьте.

Какие языки программирования можно использовать для создания прикладных программ?



Каковы же главные критерии выбора программного обеспечения для вашего компьютера?

В первую очередь выбираем операционную систему. Например операционная система Windows (рис. 2.6) содержит системные и прикладные программы, необходимые для функционирования самой системы, а также для установки других программных пакетов.



Рис. 2.6. Схема взаимодействия программного обеспечения компьютера



Знание и понимание



1. Назовите виды программного обеспечения. Как они взаимодействуют между собой?
2. Какая разница между программным обеспечением и собственно программой?
3. Определите, к какому виду относятся программные средства, встроенные в посудомоечную машину.
4. Опишите, какое программное обеспечение может быть использовано в работе малого предприятия, и для каких целей.
5. Дано различное программное обеспечение: операционные системы, файловые менеджеры, программы диагностики, антивирусные программы, программы обслуживания дисков, архиваторы, драйверы устройств. Заполните таблицу по образцу.

Наименование	Назначение	Примеры
Операционные системы	Комплекс программ, распределяющих ресурсы компьютерной системы и организующих работу других программ	Windows

6. Определите, о каком программном обеспечении идет речь.
 - 1) Это программное обеспечение включает:
 - программы диагностики состояния компьютера, антивирусные средства,
 - программы для работы с накопителями (переносными или стационарными жесткими дисками),
 - программы для архивации данных и обслуживания сетей.
 - 2) Это программное обеспечение включает:
 - офисные приложения,
 - графические редакторы,
 - браузеры, почтовые клиенты,
 - обработка видео и звука,
 - мультимедийные проигрыватели,
 - программы по разным школьным предметам,
 - игры.
 - 3) Это программное обеспечение включает:
 - программы для создания системного и прикладного программного обеспечения.
7. Составьте и запишите свою версию рассказа, употребив все предложенные ключевые термины по теме.



Применение

Анализ



8. В списке представлен набор самых необходимых программ для операционной системы Windows. Дополните его программами, которые вы используете для безопасной и эффективной работы на компьютере.

- 1) **Браузеры** – для просмотра страниц сайта в сети Интернет:
 А) Google Chrome; В)
- 2) **Антивирусные программы** – защищают от нежелательного программного обеспечения, спама, хакерских атак:
 А) 360 Total Security; В)
- 3) **Программы архивации** – для архивации и разархивации файлов:
 А) WinRAR; В)
- 4) **Офисный пакет Microsoft Office** – предназначен для работы с различными приложениями:
 А) текстовый процессор Word; В)
- 5) **Графические редакторы** – для работы с графической информацией:
 А) графический редактор Paint; В)
- 6) **Системы программирования:**
 А) Scratch; В)
9. Для каждой профессии подберите прикладные программы, которые им потребуются для выполнения их работы: бухгалтер (начисляет зарплату), фотограф (создает коллаж), учитель (готовит тест для учащихся), делопроизводитель (создает базу сотрудников предприятия), ученик (пишет программу).

Профессия	Что делает?	Программа
Бухгалтер	Начисление зарплаты	1С Бухгалтерия



Синтез

Оценивание

10. Определите, какую программу и на какой компьютер мы можем установить, учитывая конфигурацию компьютера и системные требования программ.

В таблице приведены три различные конфигурации компьютера.

Название комплектующих	Компьютер А	Компьютер В	Компьютер С
Процессор	Pentium IV	AMD Athlon 64	Intel Pentium 4
Тактовая частота	2,4 ГГц	3,6 ГГц	3,3 ГГц
Объем ОЗУ (RAM)	512 Мб	2 Гб	4 Гб
Объем жесткого диска	250 Гб	1 Тб	250 Гб
Объем видеопамяти	1 Гб	2 Гб	2 Гб

Есть список программ, которые необходимо установить (в скобках указаны их системные требования):

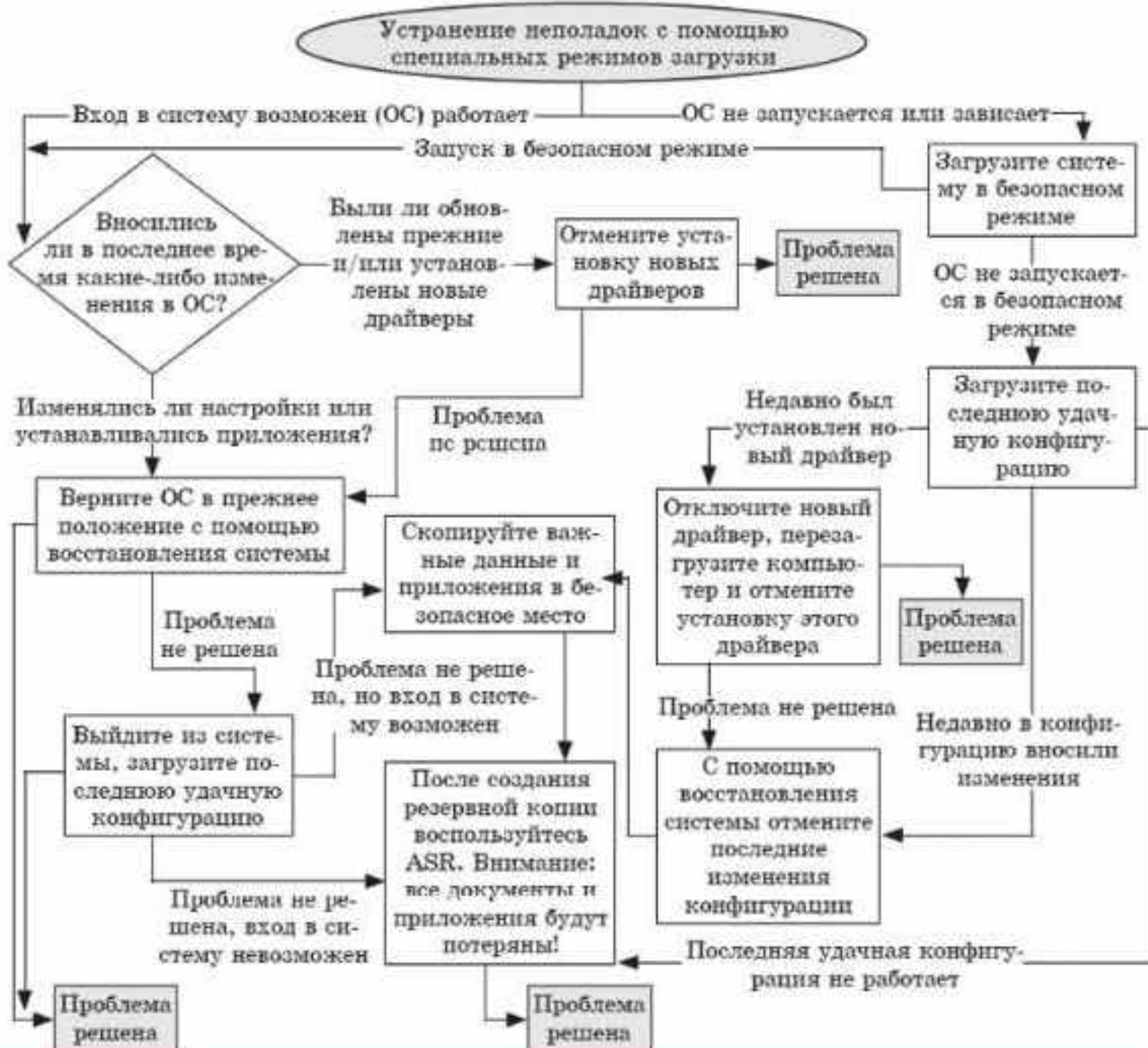
- Windows 10 (процессор с тактовой частотой 1 ГГц или выше, 1 Гб ОЗУ для 32-разрядных систем ; 2 Гб ОЗУ для 64-разрядных систем);
- Windows 7 (процессор с тактовой частотой от 0,8 ГГц, 512 Мб оперативной памяти);

- Office 365 (процесор с тактовой частотой 1 ГГц или выше, 1 Гб ОЗУ для 32-разрядных систем; 2 Гб ОЗУ для 64-разрядных систем, операционная система Windows 7 или более поздняя версия, Windows Server 2008 R2 или Windows Server 2012;
- Office Профессиональный 2019 (Windows 10, процесор: 1,6 ГГц, 2-ядерный, ОЗУ: 4 Гб, 2 Гб (32-разрядная версия);
- Adobe Premiere Pro 2.0 (Intel Pentium 4 1,4 ГГц или Pentium 4 3,4 ГГц, 512 Мб оперативной памяти);
- Мини-игры (процесор 1,7 ГГц , RAM 512 Мб, видеопамять 128 Мб).

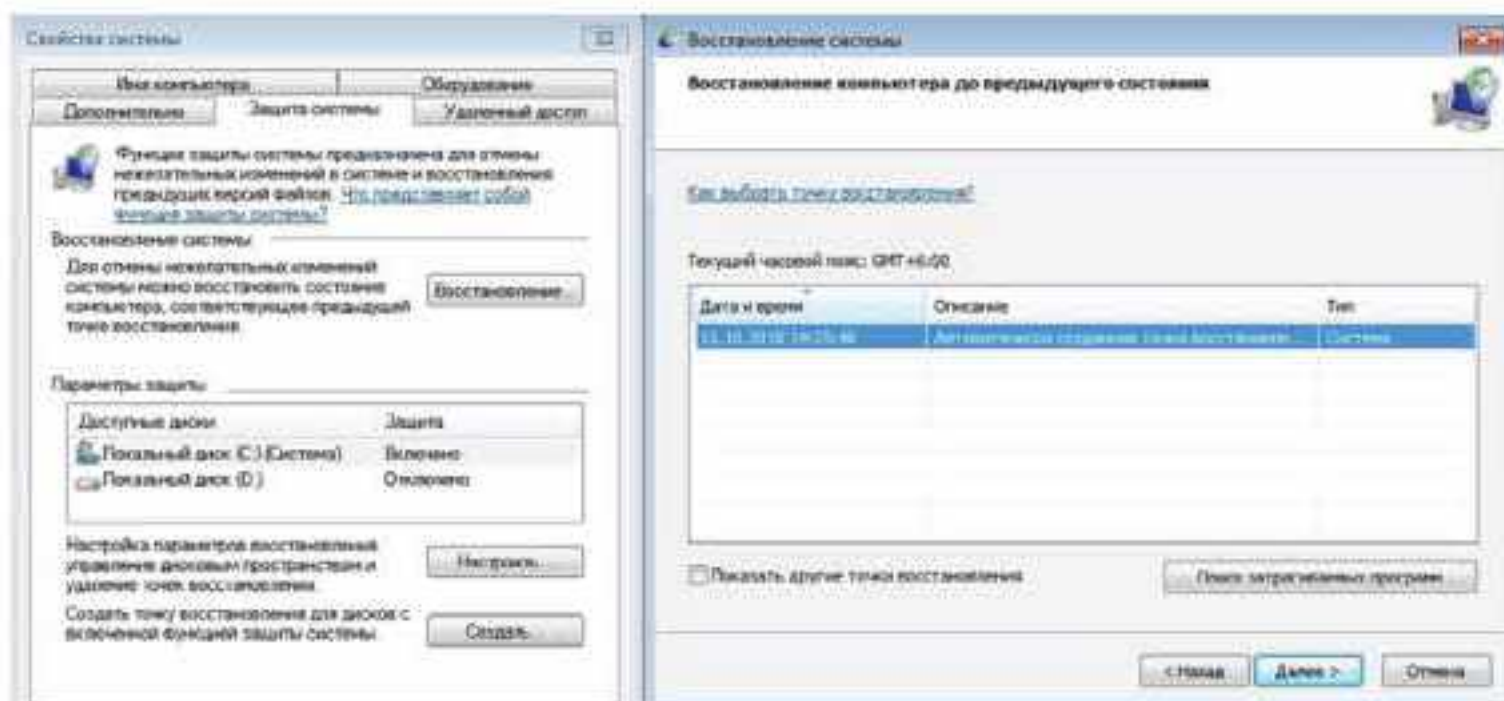


Как быстро выявить причину сбоя в работе персонального компьютера самому пользователю и восстановить работоспособность операционной системы? Рассмотрите на схеме, как можно устранить неполадки с помощью специальных режимов загрузки. На схеме показано, как решить проблему, если операционная система не запускается или виснет.

Устранение неполадок с помощью специальных режимов загрузки



Самый легкий способ устранения неполадок операционной системы, если вы создали контрольную точку восстановления (см. рис. ниже). При восстановлении операционной системы выберите последнюю дату контрольной точки. Система возвращает все настройки.



Восстановление системы

2.3. Расчет стоимости компьютера

Вы научитесь:

- выбирать конфигурацию компьютера в зависимости от его назначения;
- выбирать программное обеспечение в зависимости от потребности пользователя;
- разрабатывать и исследовать модели процессов (физических, биологических, экономических) в электронных таблицах.

Ключевые слова

Аппаратные средства
Расчет стоимости
компьютера
Комплектующие ком-
пьютера

Аппараттық құралдар
Компьютердің бағасын
есептеу
Компьютердің құрамдас
бөліктері

Hardware
Computer cost
calculation
Computer accessories



Дайте общую трактовку ключевых слов и предположите, как они будут фигурировать в последующем тексте по теме.



Компьютер – это своеобразный конструктор. Для него мы можем самостоятельно выбирать комплектующие (материнскую плату, видеокарту, оперативную память и т. д.) с техническими характеристиками, которые нужны именно нам. С чем это связано? Это связано с тем, что компьютерная сфера развивается очень быстро. Поэтому не имеет смысла покупать самый мощный и дорогой компьютер. Через некоторое время самый дорогой компьютер будет стоить в два раза дешевле. При приобретении компьютера необходимо знать, для каких целей вы приобретаете его и сколько денег вы готовы потратить.



Произведите расчет стоимости компьютера среднего уровня.



Шаг за шагом

1. Составьте список тех задач, которые планируете решить с помощью компьютера. Это определит, насколько мощный компьютер вам нужен. Например, стандартный офисный компьютер:

- работа с офисными приложениями;
- чтение книг и информации в Интернете;
- просмотр видеофайлов;
- прослушивание музыки и аудиокниг;
- хранение и просмотр фотографий;
- работа с 2D-графикой;
- простые игры.

Расчет стоимости компьютера		
Устройство	Тип комплектующих	Стоимость (тг)
Процессор (CPU)	Intel Core i3-3240, 3.4 GHz (Ivy Bridge), 2C/4T, 3 MB L3, HD2500/650, 65W, S 1155, OEM	36470
Материнская плата	MB Socket1151, MATX, iH110 (VGA+DVI, SB, GNIC), Gigabyte GA-H110M-S2PV, 2DDR4, PCIE16, 2PCI, PCIE1	16920
Оперативная память (RAM)	Kingston KVR16N11S8I-4, DDR3, 4 GB, DIMM <PC3-12800/1600MHz> CL11, 8 chip box	14035
Жесткий диск	Western Digital Caviar Blue, 1000 GB, HDD SATA WD10EZRX, 5400rpm, 64MB cache, SATA 6 Gb/s	17275
Видеокарта, звуковая карта	Интегрированные — встроены в материнскую плату	
Корпус	Корпус MATX Chenbro PC710, Slim, Черный-серебристый, Case (300W), (low profile), FSP 300W, black-silver	11576
Монитор	LG 20M38A-B, LCD 19.5" Black, 1600x900 (LED), 5ms, 200 cd/m2, 5M:1, D-Sub	29990
Клавиатура	USB, Rapoo NK2500, Черный, KeyBoard black	2499
Мышь	Мышь Ritmix RMW-505, Черный, Mouse Wireless Optical Mouse, USB, 1000dpi, 2+1 buttons, 1AA, black	1080
Итого:		129845

Рис. 2.7. Прайс-лист комплектующих

Помимо этого нужно подключить дополнительные устройства: принтер, сканер, аудио-колонки, модем, гарнитуру.

2. В электронной таблице Excel на Листе 1 составьте таблицу конфигурации компьютера, где укажите стоимость комплектующих (рис. 2.7).

3. Подсчитайте общую стоимость, используя функцию **Автосумма**. Вы увидите, сколько потратите на аппаратные средства компьютера.

4. Выберите в прайс-листе дополнительные устройства и посчитайте их стоимость (Лист 2) (рис. 2.8).

Расчет стоимости дополнительных устройств			
1	2	3	4
Устройство	Модель	Стоимость (тг)	
3	Принтер	Лазерное МФУ HP LaserJet Pro M130a MFP Laser printer/scanner/copier (A4, 600dpi, 128MB, 22ppm), USB	47952
4	Акустическая система	Ritmix SP-2030 (2.0), Черный-оранжевый SPK active RMS 2.5Wx2, USB power, Black-orange	1 270
5	Вебкамера	Global, T-89, WebCamera CMOS, 800x600, 2.0MP, USB	4622
6	Модем	Маршрутизатор ADSL TP-Link TD-W8151N, Modem/Router, 1 x 10/100Base-T, Wi-Fi b/g/n(150), ADSL2+, splitter, ext PS	7 061
7	Гарнитура	A4 Tech HS-24, Черный, Гарнитура 16 Ohm, 97dB, 20-20000Hz, 2m cable, black	2 441
8	Итого:		63346
9			

Рис. 2.8. Прайс-лист дополнительных устройств

5. Определите список программного обеспечения на Листе 3, которое будет установлено на компьютер (рис. 2.9). Обратите внимание на системные требования программ к оборудованию компьютера. К ним относятся наиболее важные для работы программы параметры компьютера: частота процессора, объем оперативной или видеопамяти.

Расчет стоимости программного обеспечения				
1	2	3	4	5
Группа	Программа	Стоимость (тг)	Примечание	
3	Системные программы	OS Microsoft Windows 10 Home, 32 bit, Russian SP1	40730	
4	Офисные приложения	Microsoft Office	17524	
5	Проигрыватели и плееры	Windows Media Player	-	в составе Windows
6	Просмотр изображений	Picasa	-	Бесплатная версия
7	Обработка графики	Adobe Photoshop	-	
8	Обработка видео	Microsoft Move Maker	-	компонент Windows
9	Браузеры	Google Chrome	-	Бесплатная версия
10	Архиваторы	Rar; Zip	-	
11	Итого		58254	
12				

Рис. 2.9. Прайс-лист программного обеспечения

6. Создайте на Листе 4 итоговую таблицу комплектации компьютера, используя стоимость комплектующих (рис. 2.7), дополнительных устройств (рис. 2.8) и программного обеспечения (рис. 2.9). Для заполнения ячейки C4 данными используйте формулу =Лист1!C13, для ячейки C5 – формулу =Лист2!C9, для ячейки C6 – формулу =Лист3!C12. В итоге таблица (рис. 2.10) заполнится данными. В ячейке C7 посчитайте, сколько всего будет стоить ваш компьютер с помощью функции =СУММ(C4:C6).

№	Комплектация	1-й вариант	2-й вариант	3-й вариант
1	Комплектующие компьютера	129845		
2	Дополнительные устройства	63346		
3	Программное обеспечение	58254		
	Всего	251445		

Рис. 2.10. Итоговая таблица комплектации компьютера

Рассмотрите еще несколько вариантов комплектации компьютера, дополнительных устройств и программного обеспечения. Затем сравните варианты и выберите тот, при котором комплектация компьютера будет оптимальной, а затраты на покупку минимальные.

Знание и понимание

1. Какие прикладные программы лучше использовать в следующих ситуациях:

- 1) вы собираете информацию о своей родословной и оформляете ее в виде компьютерной презентации;
- 2) писатель пишет свой новый роман;
- 3) фирма готовит рекламу своего товара;
- 4) сотрудники налоговой инспекции готовят отчет по сбору налогов;



щую для работы с видеозаписями. Упор в таком компьютере делается на мощность центрального процессора и количество оперативной памяти, в то время как видеокарта играет незначительную роль.

Результаты занесите в предложенную ниже таблицу.

№ п/п	Изображение компонента	Наименование компонента	Цена в тенге
1		процессор AMD Athlon II X2 245, ADX245OCK23GM, 2.90 ГГц, 2 МБ, Socket AM3, OEM	
	Итого		

Проанализируйте полученные результаты.



Синтез

Оценивание



4. Компьютеры какой конфигурации и с каким программным обеспечением лучше использовать в следующих ситуациях:

- 1) режиссер приступает к съемкам фантастического фильма, насыщенного спецэффектами;
- 2) мальчик играет в компьютерную игру-стратегию;
- 3) железнодорожная компания решила усовершенствовать процесс продажи билетов;
- 4) создатели мобильного телефона пробуют различные варианты дизайна;
- 5) сотрудник фирмы поехал отдыхать, взяв экскурсионный тур на автобусе, и продолжает получать электронную почту и новости из Интернета?

5. Проект. Используя информацию из дополнительного материала «Служба технической поддержки учащимися», составьте график технического обслуживания в вашей школе в электронной таблице Excel.

- Сформулируйте и впишите в таблицу операции по обслуживанию компьютеров, которые будет выполнять ваша служба технической поддержки (с указанием даты проведения). Для этого вам потребуется календарь учебного года, точное количество обслуживаемых компьютеров и примерная оценка времени, необходимого для проведения каждой операции.
- В столбец «Операция» впишите все операции по обслуживанию, за проведение которых отвечает служба технической поддержки вашей школы.
- Оцените, сколько времени (в часах) потребуется для проведения каждой операции на всех компьютерах. Для этого умножьте время, которое занимает операция на одном компьютере, на общее число компьютеров (если ее требуется провести на всех компьютерах).
- Заполните столбец «Периодичность».
- Заполните столбец «Начало», указав планируемые даты начала выполнения операций.
- Спланируйте проведение операций по обслуживанию, сделав пометки в календаре учебного года.
- Сделайте заключение об относительности ценности идеи по организации службы технической поддержки учащимися.

6. Представьте модель компьютера будущего. Укажите его предполагаемые технические характеристики, принципы работы и области применения.

7. Проект «Новости компьютерного мира».

Планируемый результат. Тетрадь (альбом) с подборкой сообщений, иллюстраций, с заметками из периодической электронной прессы.

Учебно-педагогическая задача. Используя материалы периодической электронной печати, собрать и оформить тетрадь (альбом). Проанализировать представленный материал и снабдить его собственными комментариями типа: «Транспортировка мониторов и телевизоров требует особой осторожности. Но в скором времени при транспортировке монитор и телевизор можно будет скатать в трубочку и спокойно нести под мышкой. Я мечтаю купить себе такой монитор».

Представьте проект, обосновав критериями отбора материала.



Служба технической поддержки учащимися

Службу технической поддержки можно организовать в любой школе группой учащихся. Ниже представлена таблица операций по техническому обслуживанию компьютеров.

Операции по техническому обслуживанию компьютеров

Название	Описание	Значение	Периодичность
<i>Общие процедуры</i>			
Очистка клавиатуры и мыши	Выполняется с помощью слегка влажной тряпки или баллончика со сжатым воздухом. Для очистки мыши необходимо извлечь ее шарик (кроме оптических мышей)	Своевременная очистка предотвращает повреждение оборудования	Раз в три месяца
Очистка экрана монитора	Выполняется с помощью рекомендованных растворителей или специальных салфеток	Пользователи не будут пытаться очистить экран неподходящими для этого растворителями	Раз в неделю
Проверка износа кабелей и проводов	Следует проверять, нет ли чрезмерного перегиба и обрыва кабелей, а также состояние и отсутствие перегрузки розеток	Своевременная проверка проводов предотвращает их повреждение и позволяет обойтись без ремонта	Раз в месяц
<i>Обслуживание оборудования и дисков</i>			
Проверка и обновление драйверов	С помощью Мастеров Windows выполняется установка новых драйверов, найденных на сайте Windows Update и сайтах производителей оборудования	Эта процедура обеспечивает использование оптимального набора драйверов	Раз в три месяца
Дефрагментация жестких дисков	Выполняется с помощью утилиты дефрагментации. Ускоряет обращение к диску.	Обеспечивает максимально эффективное использование диска	Раз в три месяца

	Возможна автоматизация этой операции с помощью Мастера планирования заданий		
Очистка диска	Позволяет удалить с диска ненужные и редко используемые файлы	Позволяет увеличить доступный объем свободного места за счет ненужных файлов	Раз в один-три месяца
<i>Обслуживание системы безопасности</i>			
Установка обновлений ОС	Если не настроена автоматическая загрузка с сайта Windows Update и установка обновлений, то операция выполняется вручную	Обеспечивает установку обновлений, критичных для защиты и эффективной работы компьютера	Раз в неделю
Полная проверка компьютера на вирусы	Если не настроена автоматическая проверка, то операция выполняется вручную, в большинстве антивирусных программ проверка компьютера может быть автоматизирована	При наличии актуальных антивирусных баз обеспечивает поиск и удаление найденных вирусов	Раз в неделю
Обновление антивирусных баз	Загрузка с веб-сайта разработчика антивирусной программы и установка последних антивирусных баз (в большинстве антивирусных программ может быть автоматизирована)	Обеспечивает распознавание и удаление недавно появившихся вирусов	Раз в две недели
Обзор веб-сайтов в поисках сведений о новых вирусах	Позволяет узнать о появлении новых вирусов и при необходимости тут же получить соответствующие базы	Обеспечивает своевременную защиту от атак новых вирусов	Ежедневно

Итоговые задания к разделу «Выбираем компьютер»

1. В данное предложение вставьте пропущенные слова: «Конфигурацию компьютера можно назвать _____ и _____ тогда, когда все возможности его процессора, памяти, видеокарты, шин для обмена информацией дают возможность обрабатывать данные максимально быстро, минуя простейшие простои, для каждого типа решаемых задач»:

- a) конструктивной;
- b) рабочей;
- c) сбалансированной;
- d) оптимальной;
- e) логической.

2. Что влияет на приобретение компьютера той или иной конфигурации:

- a) устройства и программы компьютера должны приобретаться по высокой цене;
- b) конфигурация компьютера должна соответствовать классу решаемых на нем задач;
- c) конфигурация компьютера должны быть оснащена самыми мощными аппаратными средствами;
- d) ничего не влияет;
- e) необходимо учесть все приведенные факторы.

3. Базовая конфигурация компьютера включает в себя:

- a) процессор, внутренняя память, внешняя память, устройства ввода и вывода;
- b) арифметико-логическое устройство, устройство управления, монитор;
- c) микропроцессор, ВЗУ, ОЗУ, ПЗУ, монитор, принтер, мышь;
- d) системный блок, монитор, клавиатура, мышь;
- e) только системный блок и монитор.

4. Характеристиками этого устройства являются тактовая частота, разрядность, производительность:

- a) процессор;
- b) материнская плата;
- c) оперативная память;
- d) жесткий диск
- e) встроенная память.

5. На этом устройстве располагаются разъемы для процессора, оперативной памяти, слоты для установки контроллеров:

- a) жесткий диск;



11. Укажите три устройства, которые размещаются на материнской плате:

- a) оперативная память (ОЗУ);
- b) жесткий диск;
- c) процессор;
- d) блок питания;
- e) постоянное запоминающее устройство (ПЗУ).

12. Установите соответствие.

- | | |
|----------------------------------------|-------------------------|
| 1) базовое программное обеспечение; | a) служебные программы; |
| 2) системное программное обеспечение; | b) Windows; |
| 3) прикладное программное обеспечение. | c) MS Office; |
| | d) BIOS; |
| | e) Lingvo; |
| | f) Photoshop. |

13. Программное обеспечение компьютера – это:

- a) совокупность всех программ, позволяющих организовать решение задач на компьютере;
- b) операционная система;
- c) список имеющихся программ;
- d) программа для работы компьютера;
- e) встроенная память.

14. Программное обеспечение делится на:

- a) прикладное;
- b) компьютерное;
- c) системное;
- d) процессорное;
- e) комплексное.

15. Средства контроля и диагностики относятся:

- a) к операционным системам;
- b) к системам программирования;
- c) к пакетам прикладных программ;
- d) к сервисному программному обеспечению;
- e) к операционным оболочкам.

16. Облачные технологии (вычисления) (англ. cloud computing) – это:

- a) вычислительный ресурс для электронного хранения;
- b) вычислительный ресурс для электронного хранения, обработки и совместной работы с данными в сети Интернет;
- c) вычислительный ресурс для электронного хранения, обработки и совместной работы на диске;

- d) вычислительный ресурс для работы с данными в сети Интернет;
- e) вычислительный ресурс для работы с электронной почтой.

17. Сетевой этикет – это:

- a) правила поведения при работе в сети;
- b) правила поведения и общения пользователей;
- c) общение в сети, которого придерживается большинство пользователей;
- d) правила поведения, общения в сети, традиции и культуры интернет-сообщества, которых придерживается большинство пользователей;
- e) общение в сети узкого круга лиц.

18. Системное программное обеспечение – это:

- a) программы для организации совместной работы устройств компьютера как единой системы;
- b) программы, предназначенные для выполнения определенных задач и рассчитанные на непосредственное взаимодействие с пользователем;
- c) программы для организации удобной системы размещения программ на диске;
- d) набор программ для работы устройства системного блока компьютера;
- e) набор драйверов.

19. Прикладное программное обеспечение – это:

- a) программы для организации совместной работы устройств компьютера как единой системы;
- b) программы, предназначенные для выполнения определенных задач и рассчитанные на непосредственное взаимодействие с пользователем;
- c) программы для организации удобной системы размещения программ на диске;
- d) набор программ для работы устройства системного блока компьютера;
- e) набор вспомогательных программ.

20. Какая программа обязательна для установки на компьютер:

- a) система программирования;
- b) прикладные программы общего назначения;
- c) прикладные программы специального назначения;
- d) сервисные программы;
- e) операционная система?

Проекты к разделу II

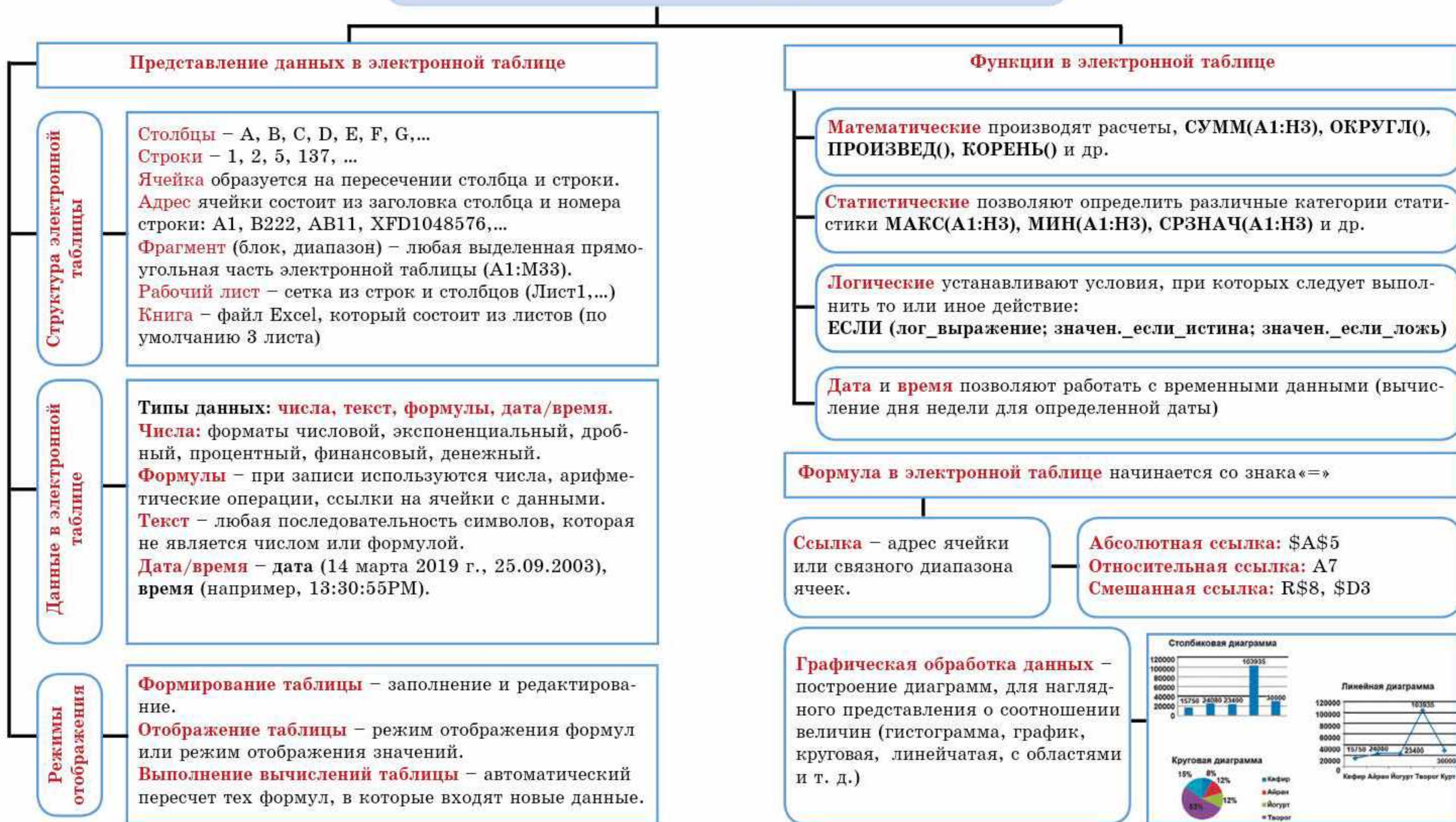
1. Проект «Собери компьютер» (совместная работа с документами в Google, конфигурация компьютера с учетом ценовой политики).
2. Исследование «Я выбираю ОС» (выбор программного обеспечения).

СИСТЕМА ОСНОВНЫХ

ПОНЯТИЙ РАЗДЕЛА III

БАЗЫ

ДАННЫХ



Раздел III

БАЗЫ ДАННЫХ

3.1. Базы данных

Вы научитесь:

– объяснять термины «базы данных, запись, поле».

Ключевые слова

Базы данных
Запись
Поле

Деректер базасы
Жазба
Өріс

Database
Record
Field



Дайте общую трактовку ключевых слов и предположите, как они будут фигурировать в последующем тексте по теме.



Для реализации автоматизированного сбора, обработки, хранения и использования данных применяется **информационная система**. Основой многих современных информационных систем (информационно-справочные системы) являются **базы данных**.

Базы данных характеризуются большим объемом хранимых и обрабатываемых данных.



База данных (БД) – это организованная совокупность данных, предназначенная для хранения, изменения и обработки взаимосвязанной информации, преимущественно больших размеров.

В большинстве случаев базу данных можно рассматривать как **информационную модель** некоторой реальной системы. Примером может быть *электронный журнал школы, национальная образовательная база данных (НОБД), книжный фонд библиотеки, кадровый состав любой организации, учебный процесс в школе, электронный архив судебных решений* и многое другое.



Приведите примеры использования баз данных из повседневной жизни.



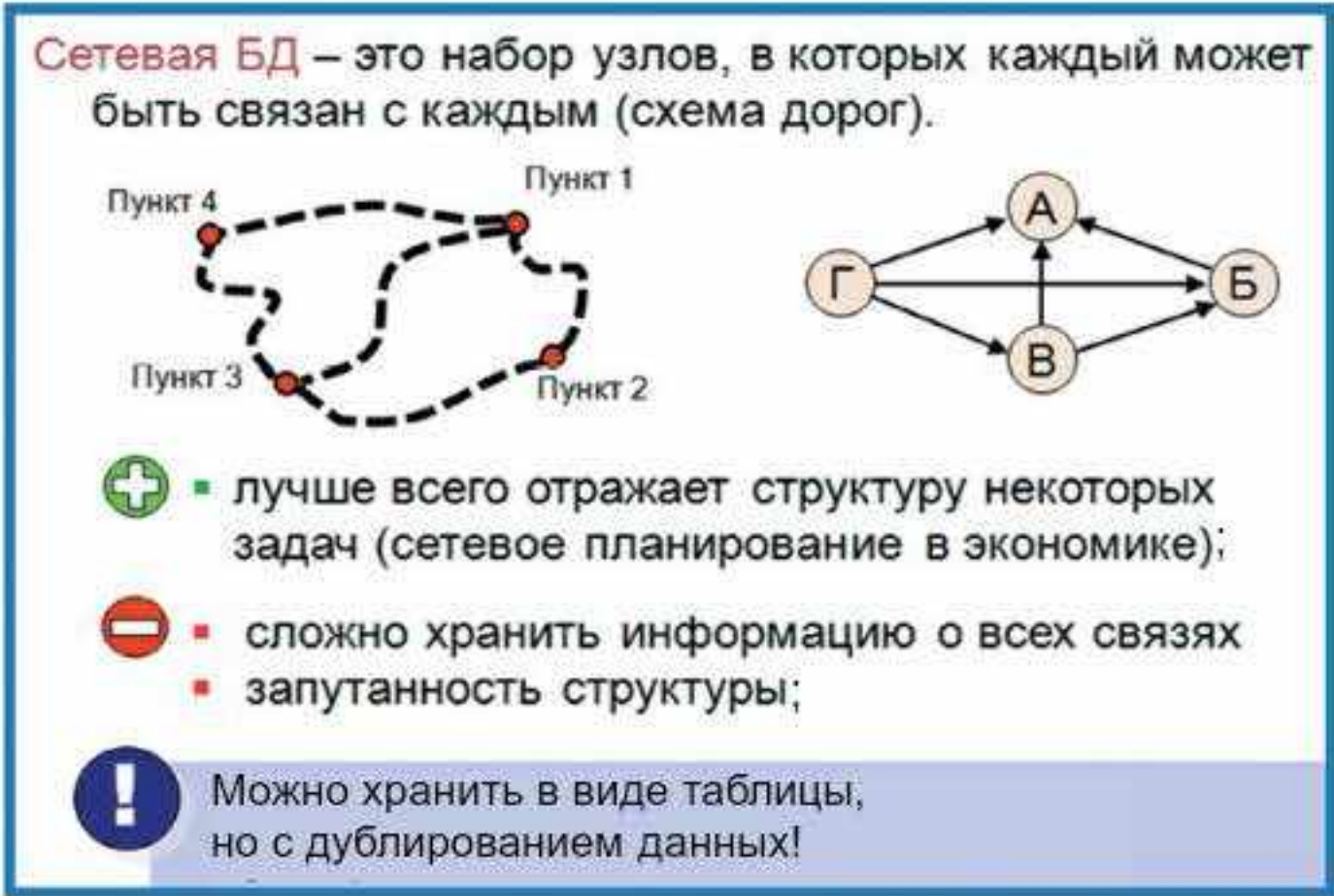


Рис. 3.1. Сетевая база данных

Базы данных в зависимости от структурирования данных делятся на *сетевые* (рис. 3.1), *иерархические* (рис. 3.2) и *табличные*, или *реляционные* (рис. 3.3).



Рис. 3.2. Иерархическая база данных

Достоинства «+» иерархической базы данных	Недостатки «-» иерархической базы данных
<ul style="list-style-type: none"> - эффективное использование памяти компьютера; - хорошие показатели времени при выполнении основных операций над данными. 	<ul style="list-style-type: none"> - громоздкость для обработки информации со сложными логическими связями; - сложность понимания для обычного пользователя.

Самые распространенные базы данных – *табличные*, иначе они называются *реляционными* (англ. *relation* – «отношение»). Отсюда происходит название *реляционные базы данных*.

Табличные БД содержат перечень объектов одного типа


Модель – картотека


Примеры:

- записная книжка;
- каталог в библиотеке.

Алибеков Рустам
 Достык пр., д. 32, кв. 11
 355-77-77

Фамилия	Имя	Адрес	Телефон
Алибеков	Рустам	Достык пр., д. 32, кв. 11	355-77-77
		Кунаева ул., д. 25, кв. 12	233-66-44

 1) самая простая структура
 2) все другие типы БД используют таблицы

 во многих случаях – дублирование данных:

А.С. Пушкин	Сказка о царе Салтане	20 стр.
А.С. Пушкин	Сказка о золотом петушке	12 стр.

Рис. 3.3. Табличная, или реляционная, база данных

Таблица в реляционной базе данных является объектом и служит для ввода, хранения и обработки информации. Таблица состоит (рис. 3.4) из *записей (строк)* и *полей (столбцов)*.



Запись – это строки таблицы, в которых записывается и хранится информация. В таблице может быть *любое количество записей* (ограничение – объем носителя) или *ни одной записи* (пустая таблица).

Поля – это столбцы таблицы, образующие структуру базы данных. Для каждого поля задается *имя поля, тип данных, перечень свойств, описание*.

Имя поля – это уникальное имя. Имена полей могут содержать любые символы, за исключением точки (.), восклицательного знака (!), апострофа (') и квадратных скобок ([]) и не должны начинаться с пробела.

	Поле 1	Поле 2	Поле 3	Поле 4	Поле 5
Запись 1	ИЗД_№	АВТОР	НАЗВАНИЕ	ГОД_ИЗД	ИЗДАТЕЛЬСТВО
Запись 2	0001230	К. Муратова	Тесты	2012	Атамұра
Запись 3	0001256	И. Аденов	Физика	2017	Мектеп
Запись 4	0001237	И. Кузнецова	ИКТ	2018	Лаборатория
.....	

Рис. 3.4. База данных Домашняя библиотека

Для каждой таблицы реляционной базы данных определяют **первичный ключ**.



Под **первичным ключом** понимают поле или набор полей, однозначно (уникально) идентифицирующих запись.

Иначе можно сказать, что первичный ключ не должен повторяться в разных записях. Например, в базе данных Домашняя библиотека ключом может быть выбран **инвентарный номер** книги, который не может совпадать у разных книг.



Вспомните типы данных в программировании в электронной таблице Excel. Что такое *тип данных*? Какие бывают типы данных?



В зависимости от цели создания базы данных строится структура будущей таблицы, состоящей из данных разного типа.

Типы данных задают характеристику и диапазон набора значений, которые могут содержаться в поле. Возможны следующие типы данных:

- **Числовой** – тип данных для хранения числовых значений, используемых в математических расчетах. Это могут быть целые или действительные числа.

- **Текстовый** – тип данных, используемый для хранения символьных или числовых данных, не требующих вычислений.

- **Дата/Время** – тип данных для хранения календарных дат и текущего времени.

- **Денежный тип** – тип данных для хранения денежных значений.

- **Логический тип** – оперирует двумя значениями ИСТИНА и ЛОЖЬ. Используется для построения логических выражений.



В табл. 3.1 дан фрагмент базы данных **Погода**. Назовите, какие типы данных полей используются в ней. Укажите различие между ними.



Таблица 3.1

День	Осадки	Температура (°С)	Давление (мм рт. ст.)	Влажность (%)
15.03.2018	Дождь	-3,5	746	67
17.03.2018	Без осадков	0	750	62
17.03.2018	Дождь	1,0	746	100

Для строчного представления структуры таблицы применяется форма:

Имя_таблицы (ИМЯ_ПОЛЯ_1, ИМЯ_ПОЛЯ_2, ..., ИМЯ_ПОЛЯ_N).

Имена полей, составляющие **первичный ключ**, подчеркиваются. Приведем примеры строчного представления таблицы для рис. 3.6 и табл. 3.1.

Домашняя_библиотека (ИНВ_№, АВТОР, НАЗВАНИЕ, ГОД_ИЗД, ИЗДАТЕЛЬСТВО).

Погода (ДЕНЬ, ОСАДКИ, ТЕМПЕРАТУРА, ДАВЛЕНИЕ, ВЛАЖНОСТЬ).

Для полей таблиц **Домашняя_библиотека** и **Погода** могут быть установлены следующие типы данных:

Символьный: АВТОР, НАЗВАНИЕ, ИЗДАТЕЛЬСТВО.

Числовой: ИНВ_№, ОСАДКИ, ТЕМПЕРАТУРА, ДАВЛЕНИЕ, ВЛАЖНОСТЬ.

Дата/Время: ГОД_ИЗД, ДЕНЬ.



Знание и понимание



1. Что такое *база данных*?
2. Для чего предназначены базы данных? Выберите верный ответ:
 - для выполнения вычислений на компьютере;
 - для осуществления хранения, поиска и сортировки данных;
 - для принятия решений.
3. Какие варианты классификации баз данных существуют?

4. Почему реляционный тип базы данных является наиболее распространенным?
5. Что такое *запись в реляционной базе данных*?
6. Что такое *поле, тип поля*? Какие бывают типы полей?
7. Для чего определяют первичный ключ для таблицы реляционной базы данных?
8. Составьте и запишите свою версию рассказа, употребив все предложенные ключевые термины по теме.



Применение

Анализ



9. Определите первичные ключи и типы полей в следующих отношениях:
- Самолеты** (НОМЕР_РЕЙСА, МЕСТО_ОТПРАВЛЕНИЯ, МЕСТО_ПРИБЫТИЯ).
- Уроки** (ДЕНЬ_НЕДЕЛИ, НОМЕР_УРОКА, КЛАСС, ПРЕДМЕТ, УЧИТЕЛЬ).
- Адрес** (ПОЧТОВЫЙ_ИНДЕКС, ОБЛАСТЬ, РАЙОН, НАСЕЛЕННЫЙ_ПУНКТ, УЛИЦА, ДОМ, КВАРТИРА).
10. Опишите структуру записей (имена полей, типы полей, первичный ключ) для баз данных:
- 1) города_Казахстана;
 - 2) выпускники_школы;
 - 3) справочная_служба_169;
 - 4) ГИС (городской информационный сервис).



Синтез

Оценивание



11. Систематизируйте информацию по теме Базы данных, составив интеллект-карту.
12. Сделайте выводы по теме «Базы данных», написав эссе.



Система управления базами данных (СУБД) – это система (совокупность языковых и программных средств), позволяющая создавать базы данных, осуществлять доступ к ним, создавать, изменять, удалять и манипулировать сведениями по запросам пользователей.

Современная СУБД содержит в своем составе программные средства создания баз данных, средства работы с данными и сервисные средства, пользовательский интерфейс и т. д.



Система управления базами данных



Схема работы в СУБД

3.2. Правила создания базы данных в электронной таблице Excel

Вы научитесь:

– создавать базу данных в электронной таблице.

Ключевые слова

Двухмерная таблица
Структура таблицы
Форма данных

Екі өлшемді кесте
Құрылымдық кесте
Деректер формасы

A two-dimensional table
Table structure
Form of data



Дайте общую трактовку ключевых слов и предположите, как они будут фигурировать в последующем тексте по теме.



Объясните своими словами, что означает термин «двухмерная таблица» Excel.



Таблицей **базы данных** является обычная двухмерная таблица Excel, созданная с соблюдением определенных правил. **Столбцы** электронной таблицы Excel – **поля** базы данных, а **строки** – это **записи** базы данных.

Перед тем как создавать базу данных, необходимо **структурировать** информацию в виде таблицы. Каждое поле (столбец) базы данных будет иметь уникальное имя и содержать информацию об одном признаке для всех записей. Совокупность всех полей образует **структуру** базы данных.

Чтобы воспользоваться информацией из базы данных, необходимо научиться самостоятельно ее создавать, осуществлять поиск информации, заменять и дополнять различными сведениями.

Приведем основные правила при создании таблицы для базы данных в электронной таблице Excel.

Правила создания базы данных Excel

1. Рекомендуется создание и ведение базы данных на отдельном листе электронной таблицы, без других данных.
2. Создание списка для таблицы. Списком называют таблицу, обязательным атрибутом которой является строка заголовков. Список выполняет функцию базы данных.
3. Первая строка в списке используется для задания имен полей базы (заголовка таблицы), которые должны быть краткими. Имя поля должно располагаться в пределах одной ячейки.
4. Каждое поле (столбец) должно быть одного определенного типа данных. Это или все числовые значения, или текстовые, или значения **Дата**.
5. Строку заголовка базы данных обрамляют в границы и выделяют другим шрифтом, чем все основные данные.
6. Запрещается объединение диапазонов ячеек в таблице базы данных.
7. Каждая запись (строка) базы данных должна содержать ячейку с заполненными данными (не должно быть пустых строк).

Для качественной базы данных необходимо правильно сформулировать задание – четко понимать, что должна уметь делать таблица.

В электронной таблице Excel можно автоматически создавать встроенную **форму данных** для диапазона или таблицы. Форма представляет собой диалоговое окно, в котором заголовки столбцов отображаются в виде подписей. Рядом с каждой подписью расположено пустое текстовое поле, в которое можно вводить данные для полей.

Форма данных упрощает *перемещение* между столбцами, *ввод*, *поиск* с помощью навигации, а также *изменение* и *удаление* строк. Если в ячейке содержится формула, то в форме данных отображается только результат ее вычисления. Саму формулу невозможно изменить с помощью формы данных. Форму данных нельзя распечатать.



Для создания базы данных используем готовую структуру табл. 3.2 «Численность молодежи от 14 до 28 лет» по статистическим данным Комитета по статистике Министерства национальной экономики Республики Казахстан.



Таблица 3.2

Численность молодежи в возрасте от 14 до 28 лет

Области, города	Численность молодежи, тыс., 2013	Численность молодежи, тыс., 2014	Численность молодежи, тыс., 2015	% к общей численности, 2013	% к общей численности, 2014	% к общей численности, 2015
1	2	3	4	5	6	7
Республика Казахстан	4376100	4293353	4206300	25,9	25	24,1
Акмолинская область	173451	169558	165154	23,7	23,1	22,4
Актобе	224666	219333	213867	28,2	27,1	26
Алматинская область	496710	486710	450795	25,5	24,5	23,5
Атырау	148295	146263	144152	26,7	25,8	24,8
Западно-Казахстанская область	157654	153930	149595	25,5	24,7	23,7
Тараз	271018	265921	260631	25,3	24,5	23,7
Караганды	331018	322431	314966	24,3	23,5	22,9
Костанай	215122	208279	201440	24,4	23,6	22,9
Кызылорда	193377	191797	189637	26,6	25,9	25,2
Мангыстау	154944	153916	152652	27,3	26,2	25,2
Туркестанская область	733294	729251	723936	27,4	26,7	26
Павлодар	175801	170439	164465	23,5	22,6	21,8
Северо-Казахстанская область	125098	120677	116212	21,6	21	20,3
Восточно-Казахстанская область	327564	316595	304972	23,5	22,7	21,9
Нур-Султан	228589	218589	213350	28,9	26,8	25
Алматы	423436	419664	440476	28,7	27,8	26,8

Шаг за шагом

1. Создайте заголовок таблицы и заполните данные для первой записи **Республика Казахстан** в электронной таблице, придерживаясь правил создания базы данных.

2. Для ввода информации в две строки используйте сочетание клавиш **Alt+Enter** или кнопку **Перенос текста** из группы **Выравнивание**.

3. Формат выделенной области можно задать разными способами: командой **Формат ячейки...** на вкладке **Главная** из группы **Ячейки** либо из контекстного меню выбрать **Формат ячейки...** (на рис. 3.5: слева вызов команды из Главного меню, справа – из контекстного).

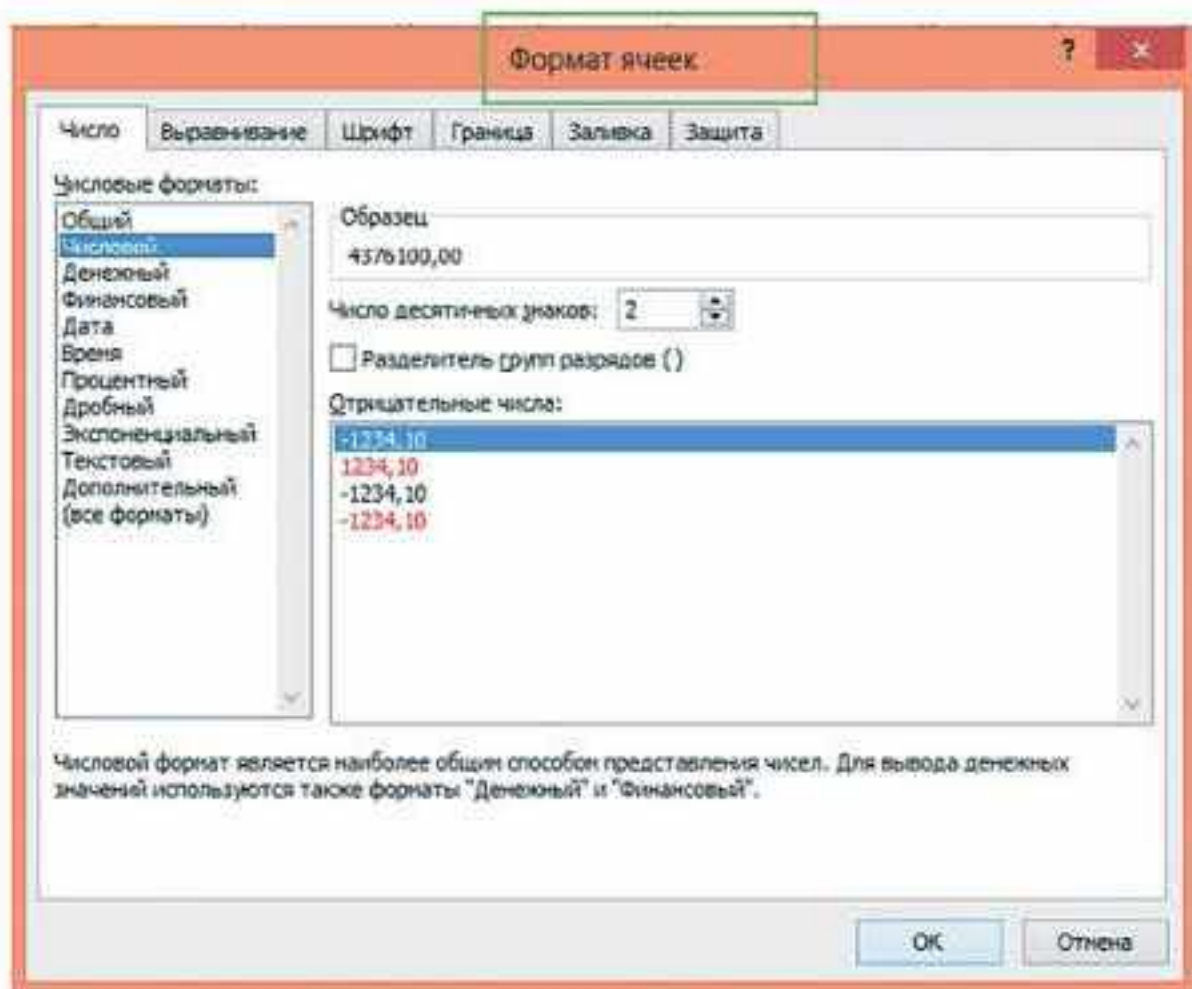
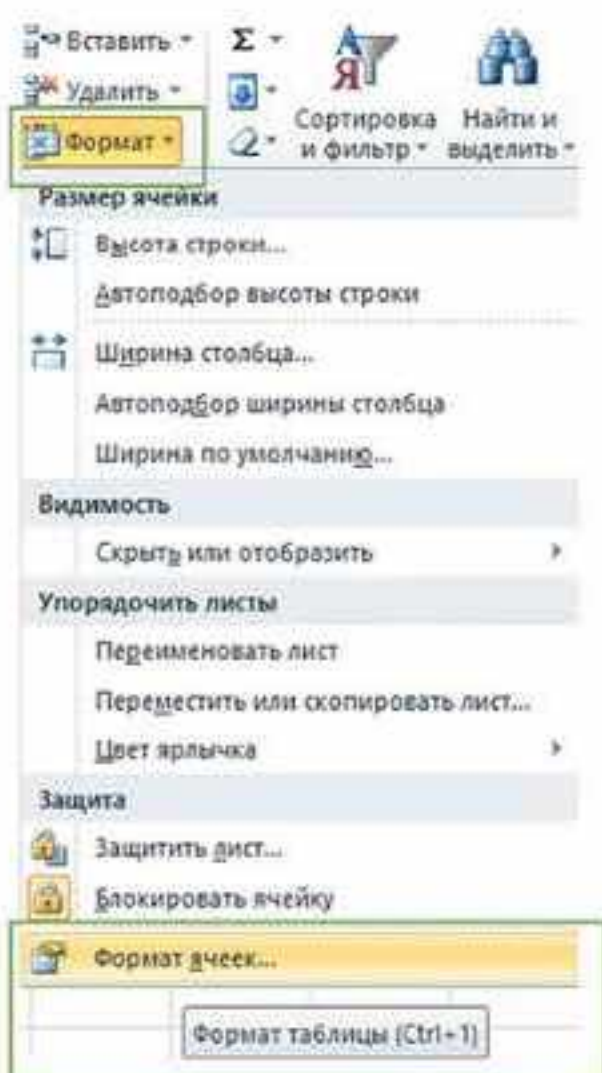


Рис. 3.5. Формат ячеек

4. Задайте **числовой формат** для данных поля **В** в процентах к общей численности населения. Для этого в диалоговом окне **Формат ячеек...** на вкладке **Число** выберите из списка **Числовые форматы**: формат **Числовой**. **Число десятичных знаков**: установите – 1 (рис. 3.6).

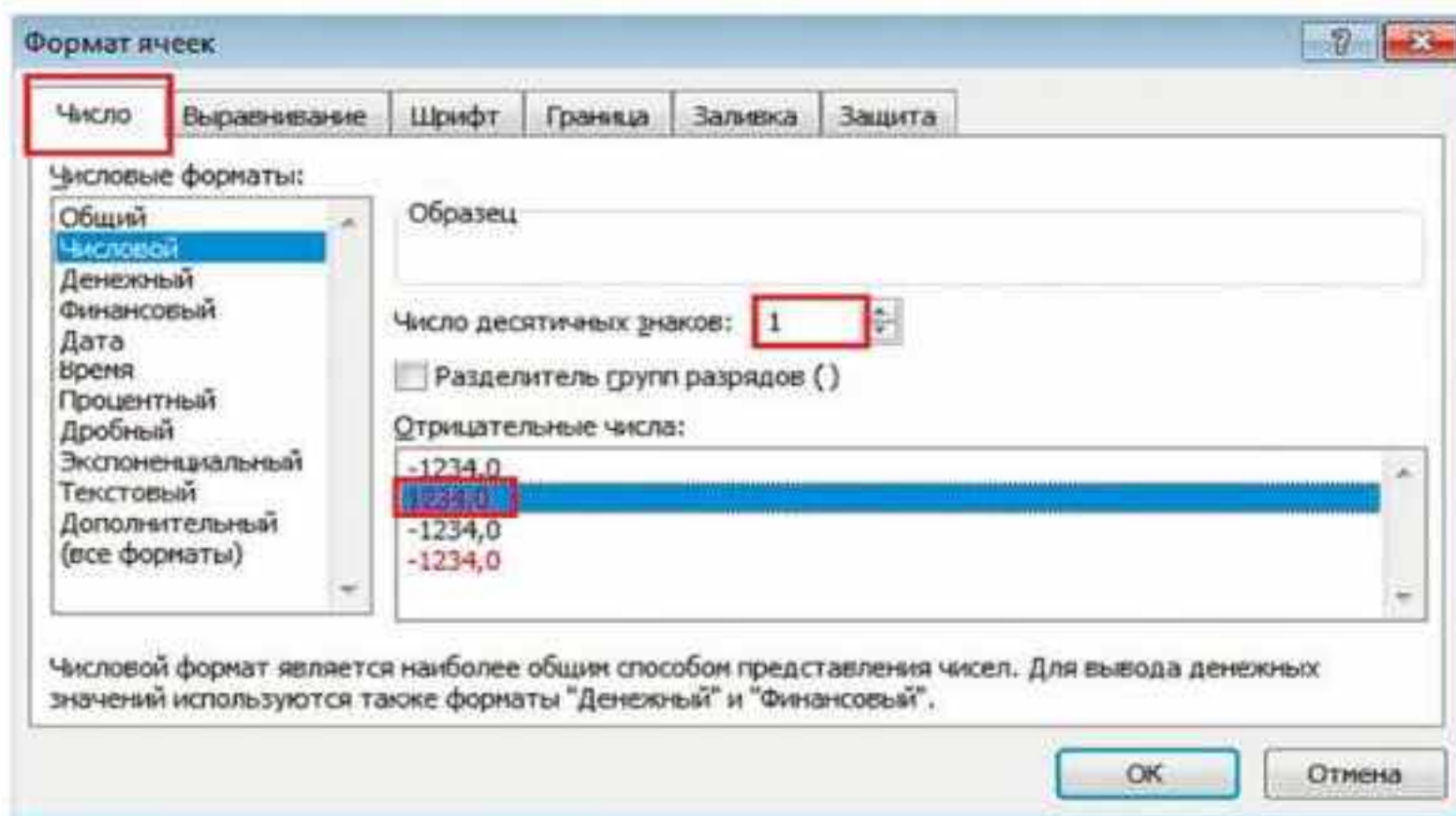


Рис. 3.6. Числовой формат для данного поля **В** в процентах к общей численности населения

5. После установки типа данных в поле выполните форматирование данных с помощью вкладок **Выравнивание**, **Шрифт**, **Граница** из диалогового окна **Формат ячейки** (рис. 3.7) или **Форматировать как таблицу** из группы **Стили** с вкладки **Главная**.

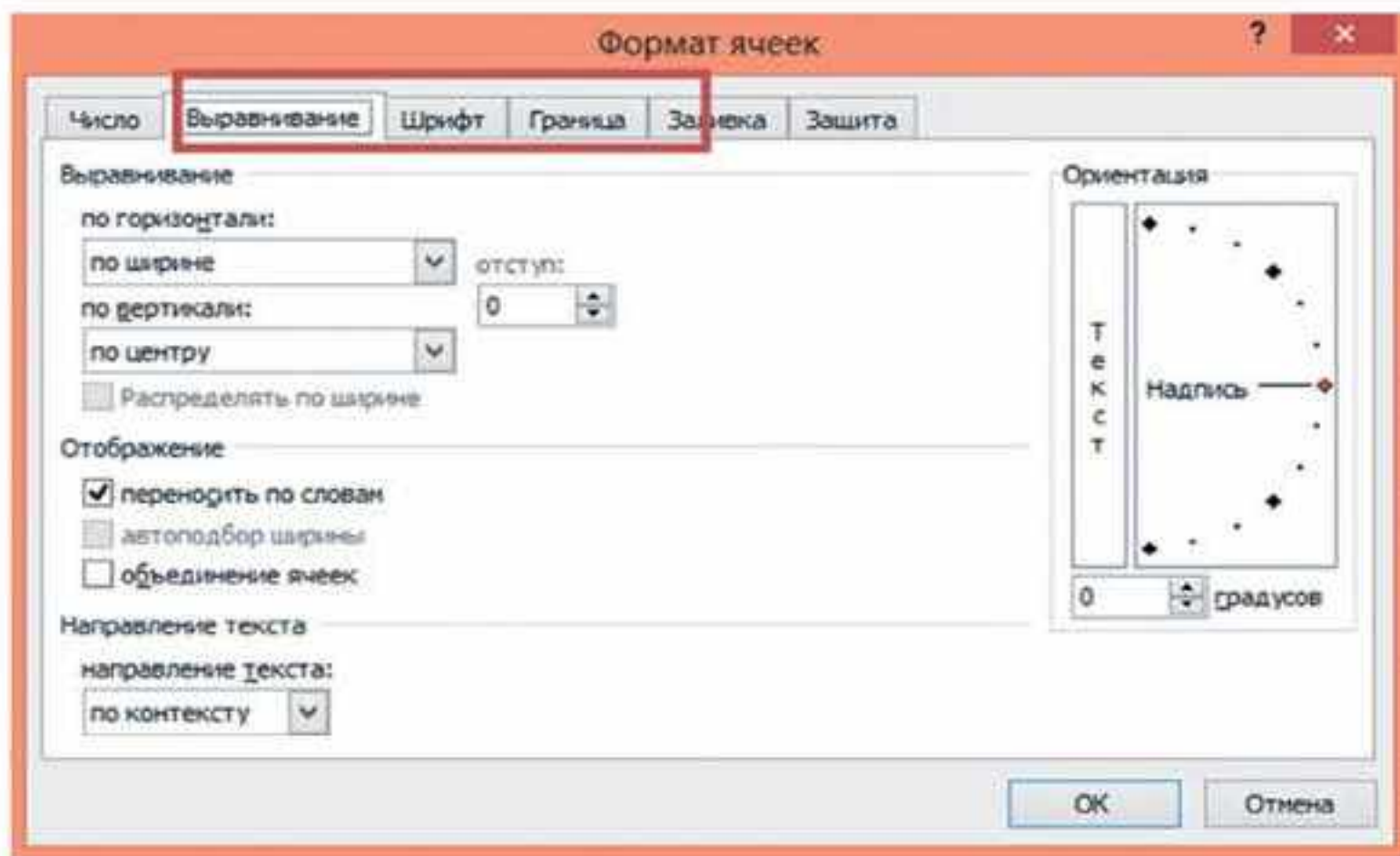




Рис. 3.7. Вкладки для форматирования **Выравнивание**, **Шрифт**, **Граница**


6. Для создания **формы данных** и ввода записей выполните алгоритм: Добавьте кнопку **формы**  на панель **Быстрого доступа**, выполнив следующие действия:

– щелкните стрелку, расположенную рядом с панелью **Быстрого доступа**, и выберите пункт **Другие команды**;

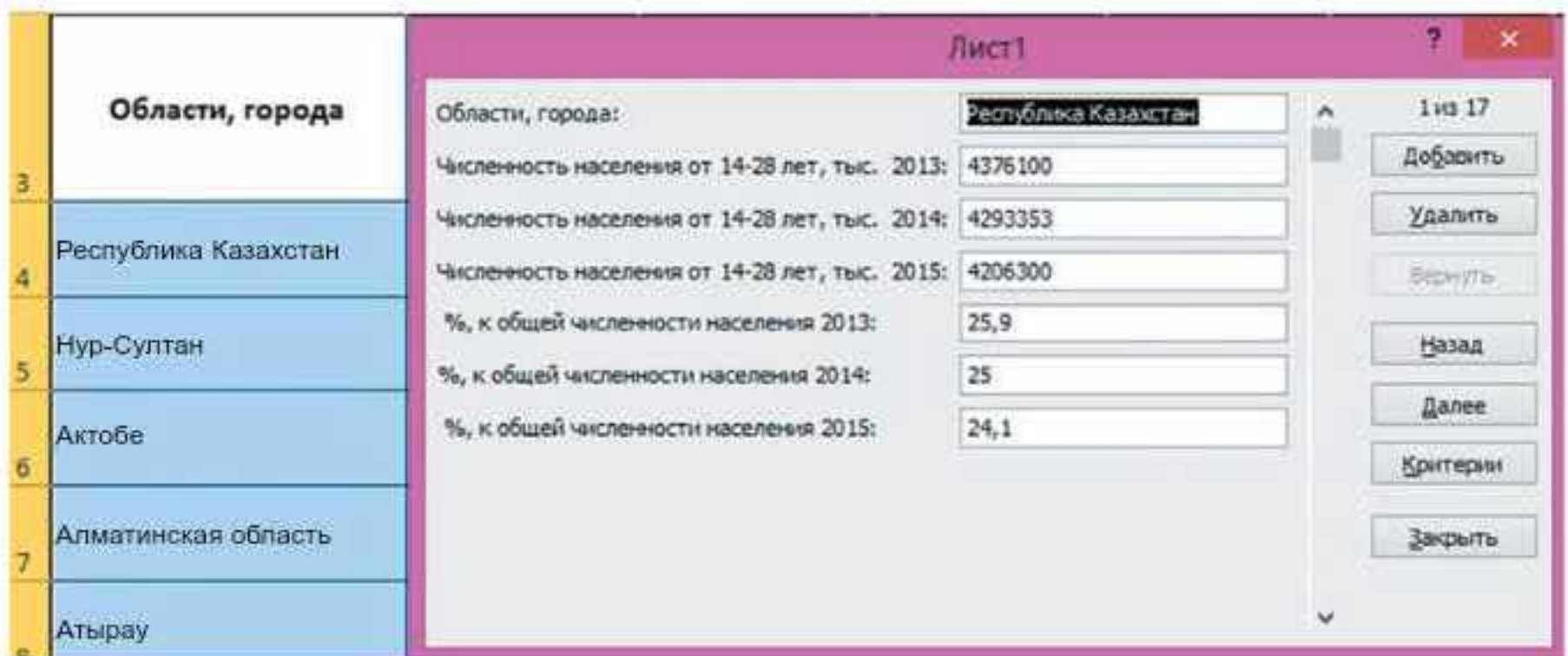
– в списке **Выбрать команды** из выберите **Все команды**, в списке выберите кнопку **формы** ;

– нажмите кнопку **Добавить**, а затем нажмите кнопку **ОК**.

7. Щелкните по ячейке в диапазоне или таблице, куда нужно добавить форму.

8. Нажмите кнопку **формы**  на панели **Быстрого доступа**, появится окно формы данных (рис. 3.8).

9. Ввод данных должен завершаться нажатием на клавишу Enter.



Области, города	Численность населения от 14-28 лет, тыс. 2013:	Численность населения от 14-28 лет, тыс. 2014:	Численность населения от 14-28 лет, тыс. 2015:	% к общей численности населения 2013:	% к общей численности населения 2014:	% к общей численности населения 2015:
Республика Казахстан	4376100	4293353	4206300	25,9	25	24,1
Нур-Султан						
Актобе						
Алматинская область						
Атырау						

Рис. 3.8. Форма данных



Поэкспериментируйте.

Почему не рекомендуется объединять диапазоны ячеек в таблице базы данных?



Для поиска строки с помощью панели навигации используйте форму данных.

1. Попробуйте выполнить последовательный переход от одной строки к другой, щелкнув нужную стрелку на полосе прокрутки в форме данных.

2. Перейдите через десять строк, щелкнув полосу прокрутки между стрелками.

3. Перейдите к следующей строке в диапазоне или таблице, нажимая кнопку **Далее**.

4. Перейдите к предыдущей строке в диапазоне или таблице, нажимая кнопку **Назад**.



Попробуйте использовать форму данных для добавления и изменения записей, удаления строк в таблице **Численность населения от 14 до 28 лет**.



Добавление строки

1. Для добавления строки данных в форме нажмите на кнопку **Добавить**.

2. Введите данные для новой строки. Переход к следующему полю производится с помощью клавиши **Tab**, а к предыдущему – с помощью сочетания клавиш **Shift + Tab**.

3. После ввода данных нажмите клавишу **Enter**. При добавлении строк с помощью формы данных таблица расширяется вниз. Строка будет добавлена в нижнюю часть диапазона или таблицы.

4. Добавьте запись в форме данных другим способом, нажав на кнопки **Критерии**, затем **Правка** (рис. 3.9).

	Численность	Численность	Численность	%, к общей	%, к	
3	<div style="border: 1px solid gray; padding: 5px;"> <p style="text-align: center;">Лист1</p> <p>Области, города: <input type="text"/></p> <p>Численность молодежи, тыс. 2013: <input type="text"/></p> <p>Численность молодежи, тыс. 2014: <input type="text"/></p> <p>Численность молодежи, тыс. 2015: <input type="text"/></p> <p>%, к общей численности 2013: <input type="text"/></p> <div style="float: right; border: 1px solid gray; padding: 5px;"> <p>Новая запись</p> <p>Добавить</p> <p>Удалить</p> <p>Вернуть</p> </div> </div>					
4						Республика Казахстан
5						Нур-Султан
						Актобе
						Алматинская область

Актобе

Алматинская область

Атырау

Западно-Казахстанская

Рис. 3.9. Добавление строк

Изменение данных в строке

1. Найдите строку, которую нужно изменить.

2. Изменив данные, нажмите кнопку **Enter**. Строка будет обновлена.

3. Переход к следующей строке производится автоматически.

4. Отмените любые изменения с помощью кнопки **Вернуть**.

5. Измените запись в форме данных другим способом, нажав на кнопку **Критерии**, затем **Правка** (рис. 3.10).

Удаление данных в строке

Рис. 3.10. Изменение данных

1. Найдите в форме данных строку, которую нужно удалить, применяя **Поиск строки с помощью навигации**.

2. Нажмите кнопку **Удалить**. Откроется диалоговое окно для подтверждения операции (рис. 3.11). После подтверждения отменить удаление строки невозможно.

Заккрытие формы данных

4	Республика Казахстан	4376100	4293353	4206300	25,9	25
5	Нур-Султан					
6	Актобе					
7	Алматинская область					

Microsoft Excel

⚠ Запись, выведенная на экран, будет удалена.

[Были ли сведения полезными?](#)

Рис. 3.11. Удаление строки

Чтобы закрыть форму данных и вернуться к листу, нажмите кнопку **Заккрыть**.



Знание и понимание



1. Почему в электронной таблице Excel можно создавать реляционные базы данных?
2. Можно ли между заголовком таблицы и записями оставлять пустую строку? Как удалить пустую строку в электронной таблице Excel?
3. Какие типы данных можно использовать в полях базы данных?
4. Может ли существовать таблица без записей?
5. Как увеличить или уменьшить количество десятичных знаков в электронной таблице Excel?
6. Что нужно сделать, чтобы разместить имя поля в двух строчках?
7. Для чего используется форма данных?
8. Как добавить, изменить записи в форме данных?
9. Как можно удалить записи в форме данных?
10. Составьте и запишите свою версию рассказа, употребив все предложенные ключевые термины по теме.



Применение

Анализ



11. Создайте базу данных Процессоры (Intel, AMD) в электронной таблице Excel из строчного представления структуры таблицы (данные скопируйте из прайс-листов компьютерных фирм, применив поисковые системы Интернета).



Синтез

Оценивание



12. Как известно, на каждого учащегося класса у классного руководителя должно быть мини-досье со следующими данными:
 - 1) номер по порядку;
 - 2) фамилия;
 - 3) имя;
 - 4) отчество;
 - 5) дата рождения;
 - 6) домашний адрес;
 - 7) телефон;
 - 8) фамилия, имя, отчество родителей;
 - 9) место работы родителей, рабочий телефон.
 - Попробуйте представить эти данные в виде базы данных.
 - Введите пять записей с помощью формы данных.

- Примените команду **Критерии** и подстановочные знаки для быстрого поиска информации по запросу.
- Приведите аргументы «за» и «против» применения подстановочных знаков для поиска информации.

13. Оцените степень своей готовности и способности применения ключевых слов на казахском и английском языках при передаче содержания темы, общении друг с другом и выражении своего мнения.

14. Проведите мини-исследование. Используя интернет-ресурсы, совершите виртуальную экскурсию на производство. Получите наглядное представление о рабочем месте, просмотрев видеоматериалы с рассказами лучших работников конкретных производств выбранных профессий. Составьте базу данных. Например, в поле **Профессии** включите: инженер (1380 просмотров), системный программист (2005 просмотров), учитель в средней школе (358 просмотров), врач (1287 просмотров). Какое направление профориентации, судя по результатам вашего исследования, оказалось приоритетным? Применяя электронную таблицу Excel, сделайте прогноз о том, какие профессии окажутся невостребованными в ближайшем будущем.



3D-карты в Excel

В электронную таблицу Excel версии 2013 встроен магазин приложений для скачивания и установки надстройки Power Map: **Вставка – Магазин (Insert – Office Apps)**. В Excel 2016 Power Map встроена в программу и называется **3D-карта (3D-map)**.



Для визуализации и анализа данных с географической привязкой к регионам, выполните алгоритм создания **3D-карты**.

1. Создайте файл с таблицей данных, которые нужно визуализировать. Например, таблицу с корректной информацией по разным городам и странам.
2. Выполните: **Главная – Форматировать как таблицу**.
3. Выделите диапазон данных для анализа.
4. Нажмите на вкладке **Вставка** кнопку **3D-карта**. Открывается окно создания и редактирования **3D-карты**, состоящее из одной вкладки **Главная**. На ней находятся команды добавления, удаления, редактирования карты.

3.3. Методы поиска информации

Вы научитесь:

– осуществлять поиск, сортировку и фильтрацию данных.

Ключевые слова

Поиск информации
Критерий
Сортировка
Фильтр

Ақпаратты іздеу
Критерий
Сұрыптау
Сүзгі

Search information
Criterion
Sorting
Filter



Дайте общую трактовку ключевых слов и предположите, как они будут фигурировать в последующем тексте по теме.



Если в базе данных увеличивается количество записей, то поиск информации затрудняется. Поэтому основной функцией в любой базе данных является **поиск информации по определенным критериям**. Электронная таблица Excel упрощает этот процесс путем поиска данных в форме **по критерию и фильтрации данных**.

В базах данных электронной таблицы Excel можно производить следующие операции:

- фильтрацию записей для поиска, отображения или скрывания значений в одном или нескольких полях (столбцах) данных;
- сортировку записей (строк) таблиц по возрастанию или убыванию данных в указанных полях (ячейках).



Попробуйте выполнить поиск информации с помощью панели навигации, критериев и фильтрации.



Шаг за шагом

Для поиска строки по определенным условиям нажимаем на кнопку **Критерии** и вводим в форму данных **условие сравнения**. Все элементы, начинающиеся с условия сравнения, фильтруются.

К примеру, при вводе текста *Макс* как критерия электронная таблица Excel найдет «Максимальный», «Максимум», «Максим», «Максут» и др.

При создании условий сравнения для фильтров, а также при поиске и замене содержимого в качестве условия отбора можно использовать подстановочные знаки (табл. 3.3). Подстановочные знаки существенно расширяют возможности поиска.

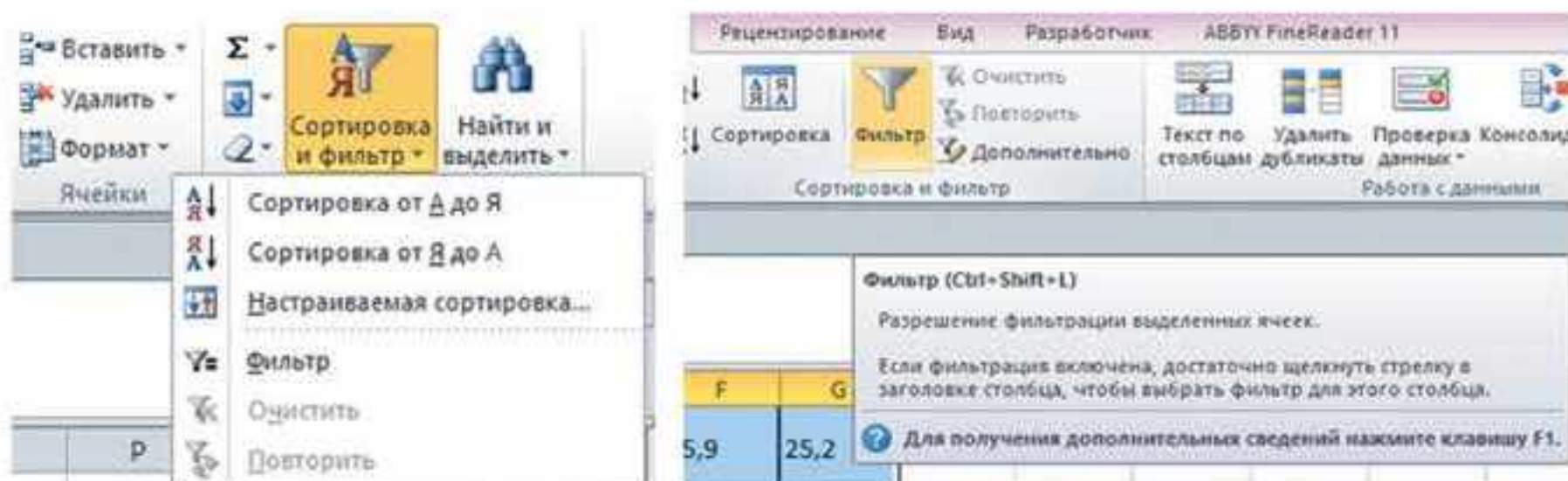
Таблица 3.3

Подстановочные знаки

Подстановочные знаки	Поиск
? (вопросительный знак)	Любой символ (один) Пример: условию «бар?н» соответствуют результаты «барин» и «барон»
* (звездочка)	Любое количество символов Пример: условию «*-восток» соответствуют результаты «северо-восток» и «юго-восток»
~ (тильда), за которой следует ?, * или ~	Вопросительный знак, звездочку или тильду Пример: условию «ан91~?» соответствует результат «ан91?»

Поиск информации с помощью фильтра осуществим с применением базы данных **Численность молодежи**.

1. Перед тем как устанавливать **Фильтр**, выделите всю таблицу и установите фильтр.



1. Фильтр с вкладки Главная

2. Фильтр с вкладки Данные

Рис. 3.12. Установка Фильтра

2. Используйте команду **Фильтр** с вкладки **Главная**, группы **Сортировка и фильтр**, либо с вкладки **Данные** команда **Фильтр** из группы **Сортировка и фильтр** (рис. 3.12). Если фильтрация включена, то для выбора данных достаточно щелкнуть стрелку в заголовке столбца и выбрать фильтр.

После включения фильтрации в заголовке таблицы у каждого столбца появятся **раскрывающиеся списки** (рис. 3.13), где можно будет выбрать нужное значение.

	Области, города	Численность молодежи, тыс.	Численность молодежи, тыс.	Численность молодежи, тыс.	%, к общей численности	%, к общей численности	%, к общей численности
3		2013	2014	2015	2013	2014	2015
4	Республика Казахстан	4376100	4293353	4206300	25,9	25	24,1
5	Нур-Султан	173451	169558	165154	23,7	23,1	22,4
6	Актобе	224666	219333	213867	28,2	27,1	26
7	Алматинская область	496710	486710	450795	25,5	24,5	23,5
8	Атырау	148295	146263	144152	26,7	25,8	24,8
9	Западно-Казахстанская область	157654	153930	149595	25,5	24,7	23,7

Рис. 3.13. Отфильтрованный список

Для примера откройте в базе данных список поля **Области, города** и поставьте галочку только возле города **Нур-Султан**. В итоге получится отфильтрованный список по г. Нур-Султан (рис. 3.14).

Таблица 3.2. Численность молодежи в возрасте от 14 до 28 лет							
	Области, города	Численность молодежи, тыс.	Численность молодежи, тыс.	Численность молодежи, тыс.	%, к общей численности	%, к общей численности	%, к общей численности
3		2013	2014	2015	2013	2014	2015
19	Нур-Султан	228589	218589	213350	28,9	26,8	25

Рис. 3.14. Установка фильтра по г. Нур-Султан

Например, необходимо узнать данные больше 25,0 в 2015 году по полю процент к общей численности населения. Для этого нужно установить числовые фильтры (рис. 3.15).

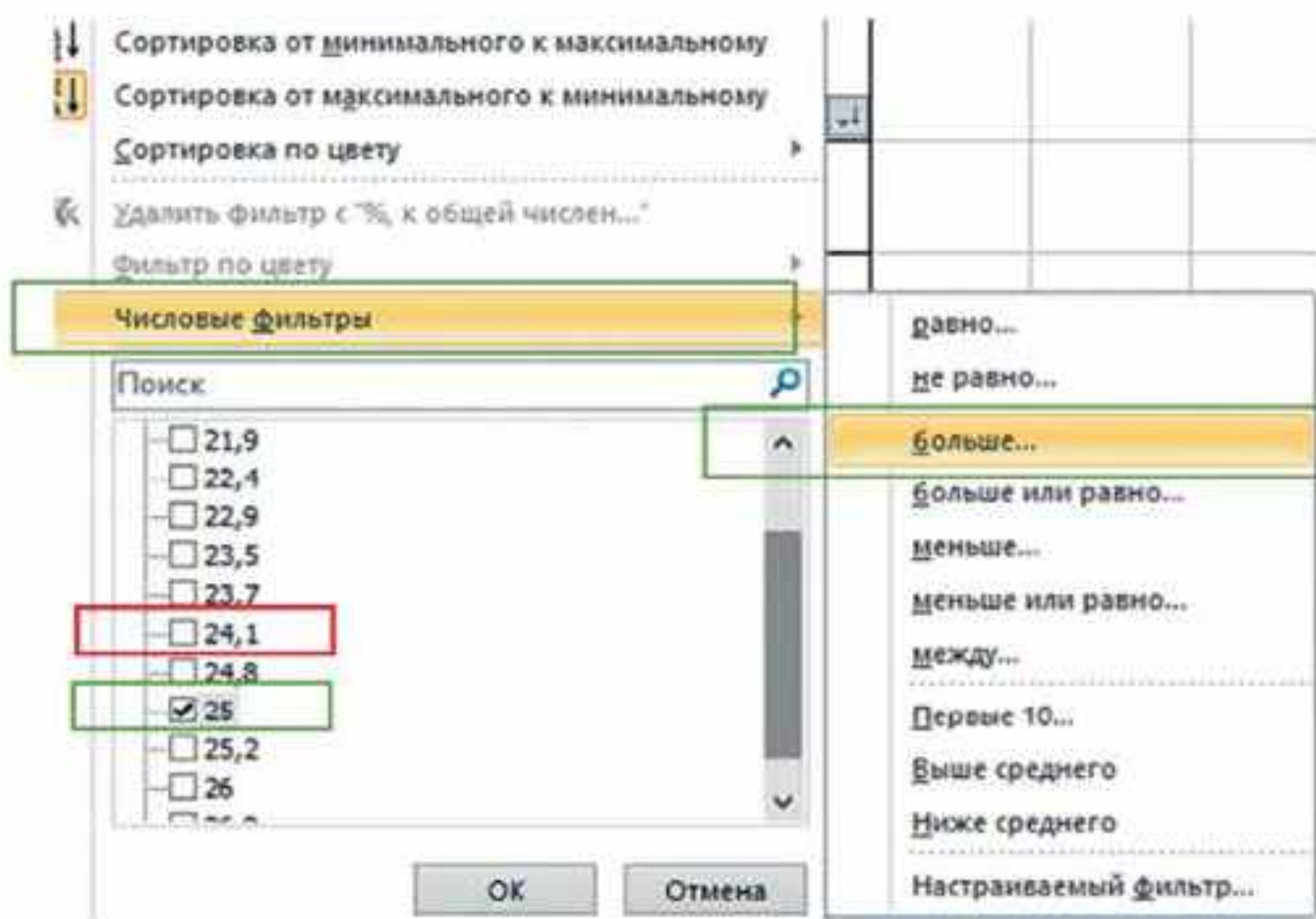


Рис. 3.15. Числовые фильтры

Устанавливаем в диалоговом окне **Пользовательский автофильтр** с параметром больше и значением 25,0 (рис. 3.16).

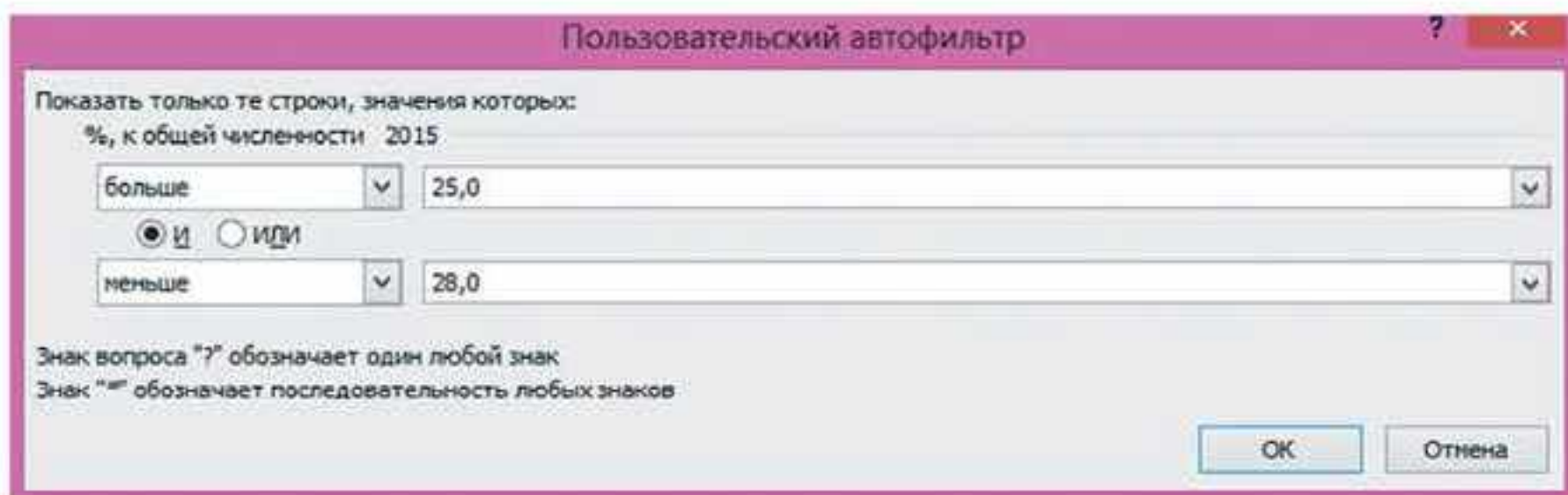


Рис. 3.16. Пользовательский автофильтр

На рис. 3.17 мы видим отфильтрованные данные по установленным параметрам.

	Области, города	Численность молодежи, тыс. 2013	Численность молодежи, тыс. 2014	Численность молодежи, тыс. 2015	%, к общей численности 2013	%, к общей численности 2014	%, к общей численности 2015
3							
4	Алматы	423436	419664	440476	28,7	27,8	26,8
5	Актобе	224666	219333	213867	28,2	27,1	26
6	Туркестанская область	733294	729251	723936	27,4	26,7	26
7	Кызылорда	193377	191797	189637	26,6	25,9	25,2
8	Мангыстау	154944	153916	152652	27,3	26,2	25,2

Рис. 3.17. Отфильтрованные данные

При работе с фильтрами можно пользоваться подстановочными знаками или такими условными обозначениями, как: * – любое количество любых символов, ? – один любой символ.

Например, нужно определить названия городов, которые оканчиваются на букву «а». Для этого введите в поле **Область, города** в строку **Поиск** фильтр с подстановочными знаками *а (рис. 3.18).

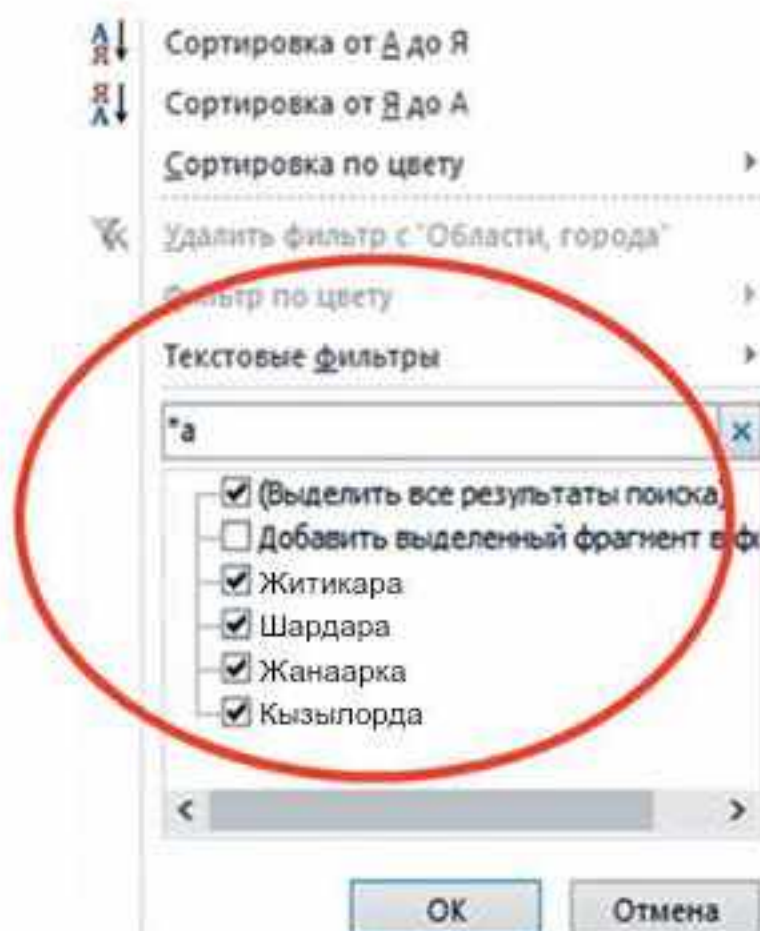


Рис. 3.18. Фильтр с условным обозначением

Если требуется определить, в каком городе Республики Казахстан за 2015 год процент численности молодежи ниже среднего, то выбираем Числовые фильтры и Ниже среднего (рис. 3.19).

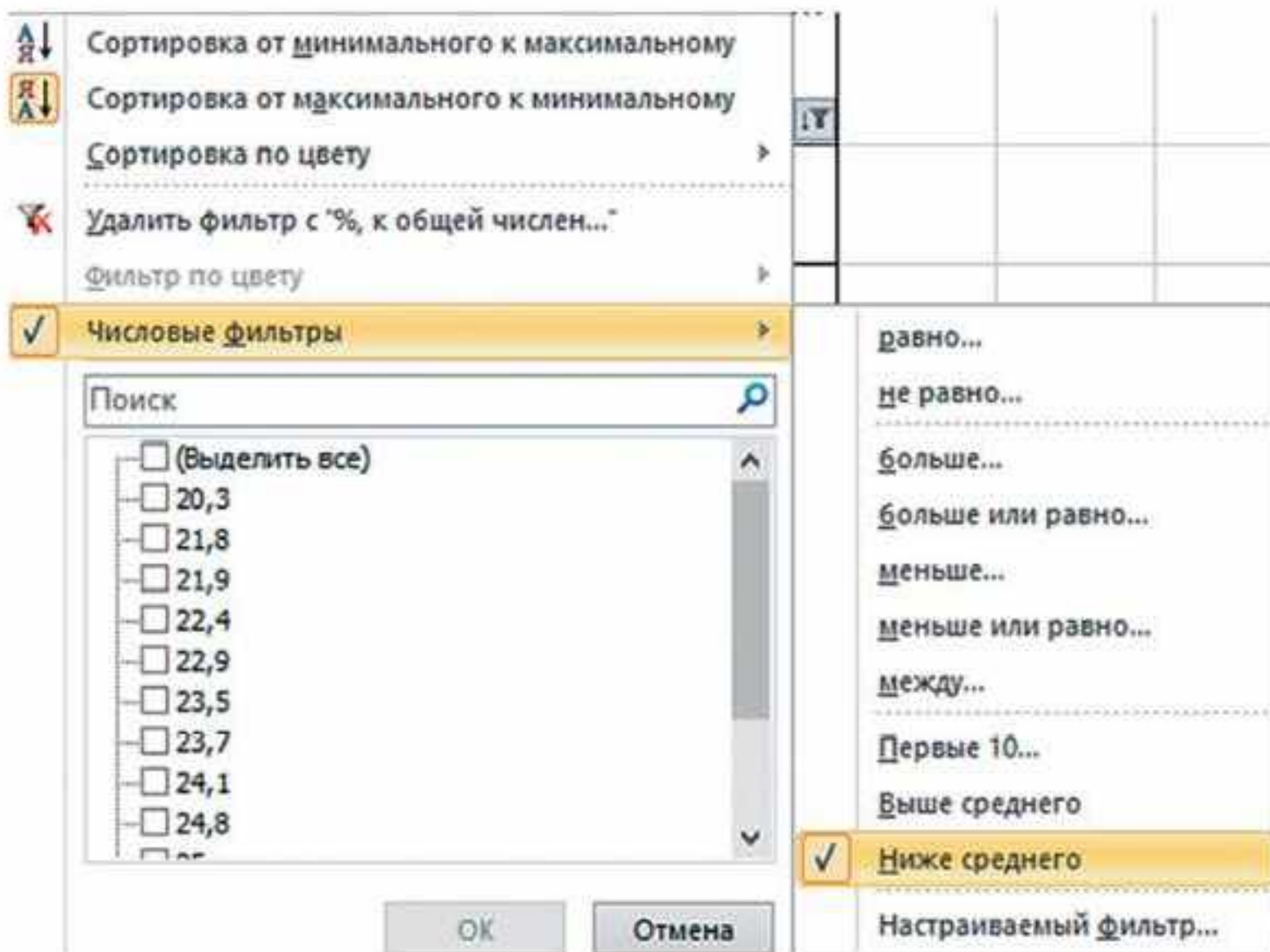


Рис. 3.19. Числовые фильтры Ниже среднего

В итоге получится отфильтрованная таблица (рис. 3.20).

Области, города	Численность молодежи, тыс.	Численность молодежи, тыс.	Численность молодежи, тыс.	% к общей численности	% к общей численности	% к общей численности
	2013	2014	2015	2013	2014	2015
3						
12	157654	153930	149595	25,5	24,7	23,7
13	271018	265921	260631	25,3	24,5	23,7
14	496710	486710	450795	25,5	24,5	23,5
15	331018	322431	314966	24,3	23,5	22,9
16	215122	208279	201440	24,4	23,6	22,9
17	173451	169558	165154	23,7	23,1	22,4
18	327564	316595	304972	23,5	22,7	21,9
19	175801	170439	164465	23,5	22,6	21,8
20	125098	120677	116212	21,6	21	20,3

Рис. 3.20. Отфильтрованная таблица по фильтру Ниже среднего



Поэкспериментируйте.

Какой фильтр нужно подключить, чтобы узнать, в каком городе наибольший процент молодежи?

Каким образом можно снять фильтр?

Отфильтруйте данные в нескольких столбцах.



Кроме фильтрации данных, с помощью команды **Фильтр** можно выполнить **сортировку**. В данной базе данных **сортировку** применим для **упорядочивания** данных:

- от **А** до **Я** от минимального к максимальному;
- от **Я** до **А** от максимального к минимальному;
- сортировка по цвету.

Перед тем как выполнить сортировку, выделите поле ячейки, по которому будете упорядочивать данные. Например, для сортировки данных по полю **% к общей численности 2015** выделите поле. Затем выберите с вкладки **Главная** из группы **Редактирование** команду **Сортировка и фильтр** либо с вкладки **Данные** команду **Сортировка** из группы **Сортировка и фильтр**. Выберите сортировку **От наименьшего к наибольшему** (рис. 3.21).

% к общей численности 2015
20,3
21,8
21,9
22,4
22,9
22,9
23,5
23,7

Рис. 3.21. Сортировка от А до Я



Знание и понимание



1. Как можно быстро найти информацию в большой базе данных?
2. Какие подстановочные знаки вы знаете?
3. Придумайте условие запроса, используя подстановочные знаки.
4. С помощью каких команд осуществляется поиск, фильтрация и сортировка данных?
5. Какие операции можно производить в базе данных электронной таблицы Excel?
6. Как можно осуществить поиск данных с помощью фильтра?
7. Каким образом можно установить числовые фильтры?
8. Для чего применяется сортировка? Какие виды сортировки вы знаете?
9. Составьте и запишите свою версию рассказа, употребив все предложенные ключевые термины по теме.



Применение

Анализ



10. Занесите на лист следующие данные:

	А	В	С
1	Ромб	Плоская фигура	
2	Шар	Тело	
3	Конус	Тело	
4	Трапеция	Плоская фигура	
5	Куб	Тело	
6	Квадрат	Плоская фигура	
7	Призма	Тело	
8	Параллелепипед	Тело	
9	Треугольник	Плоская фигура	
10	Параллелограмм	Плоская фигура	
11	Пирамида	Тело	

Скопируйте данные на другой лист и там представьте их в виде таблицы.

	А	В	С
1	Ромб	Плоская фигура	
2	Трапеция	Плоская фигура	
3	Квадрат	Плоская фигура	
4	Треугольник	Плоская фигура	
5	Параллелограмм	Плоская фигура	
6	Шар	Тело	
7	Конус	Тело	
8	Куб	Тело	
9	Призма	Тело	
10	Параллелепипед	Тело	
11	Пирамида	Тело	

Каким образом отсортированы данные?



лениями. Это поможет школьникам осознанно выбрать будущую профессию.

Необходимо создать базу данных, содержащую информацию об образовательном пространстве, в электронной таблице, которая оборудована:

- 1) компьютерами,
- 2) конструкторами по робототехнике,
- 3) наборами для занятий по микроэлектронике,
- 4) 3D-принтерами.

Поиск необходимой информации в базе данных осуществляйте с помощью условия отбора **подстановочных знаков**.



Как ограничить доступ к информации?

Программа Excel предоставляет возможность защитить информацию в таблицах путем ограничения доступа к ней.

Ниже на рисунке приведены различные параметры, доступные для защиты данных в Excel.



Вкладка **Рецензирование** для защиты информации

Для того чтобы невозможно было открыть или изменить файл Excel, его можно заблокировать с помощью пароля. Приведем несколько вариантов.

Шифрование файла. Вы можете задать пароль и заблокировать файл Excel. Это предотвращает открытие файла другими пользователями. Однако если вы забудете пароль, то открыть файл уже не сможете.

Задавание пароля для открытия или изменения книги. Вы можете задать пароль, который будет требоваться для открытия или изменения книги. Используйте этот вариант, если нужно предоставить некоторым пользователям доступ только для чтения или доступ на изменение.

3.4. Сортировка и фильтрация данных

Вы научитесь:

– осуществлять поиск, сортировку и фильтрацию данных.

Ключевые слова

*Настраиваемая
сортировка
Фильтрация*

*Бапталатын
сұрыптау
Сүзгілеу*

*Customizable sorting
Filter*



Дайте общую трактовку ключевых слов и предположите, как они будут фигурировать в последующем тексте по теме.



Сортировку данных применяют как для их анализа, так и для упорядочивания, визуализации и принятия правильных решений при работе с информацией. Сортировать записи таблицы можно по любому полю, части поля или задав настраиваемую сортировку. Например, сортировать данные можно (рис. 3.22):



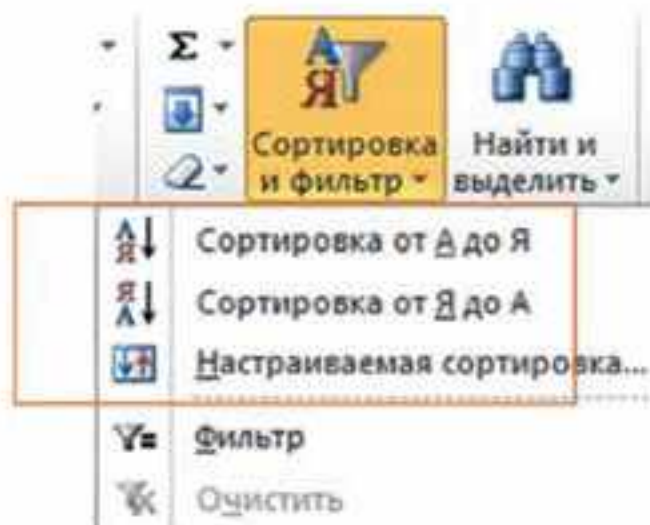
Рис. 3.22. Сортировка разных данных

В большинстве случаев сортировка применяется к полям таблицы (можно сортировать данные и по строкам).



Поэкспериментируйте.

Отсортируйте таблицу **Численность молодежи** в порядке убывания названия городов и областей.



Шаг за шагом

Выделите ячейку в столбце таблицы, который нужно отсортировать. На вкладке **Главная** в группе **Редактирование** выберите команду **Сортировка и фильтр** (рис. 3.23).

– Чтобы отсортировать значения по возрастанию, нажмите кнопку **Сортировка от А до Я** (рис. 3.24).

– Чтобы отсортировать значения по убыванию, нажмите кнопку **Сортировка от Я до А**.

Рис. 3.23. Сортировка

	А	В	С	Д
4	Акмолинская область	173451	169558	165154
5	Актобе	224666	219333	213867
6	Алматинская область	496710	486710	450795
7	Алматы	423436	419664	440476
8	Нур-Султан	228589	218589	213350
9	Атырау	148295	146263	144152
10	Восточно-Казахстанская область	327564	316595	304972
11	Тараз	271018	265921	260631
12	Западно-Казахстанская область	157654	153930	149595
13	Караганды	331018	322431	314966
14	Костанай	215122	208279	201440
15	Кызылорда	193377	191797	189637
16	Мангыстау	154944	153916	152652
17	Павлодар	175801	170439	164465
18	Республика Казахстан	4376100	4293353	4206300
19	Северо-Казахстанская область	125098	120677	116212
20	Туркестанская область	733294	729251	723936

Рис. 3.24. Отсортированная таблица по возрастанию

Чтобы отсортировать числовые значения нестандартным образом, можно использовать пункт меню **Настраиваемая сортировка...**, которая осуществляется через диалоговое окно **Сортировка** (рис. 3.25).

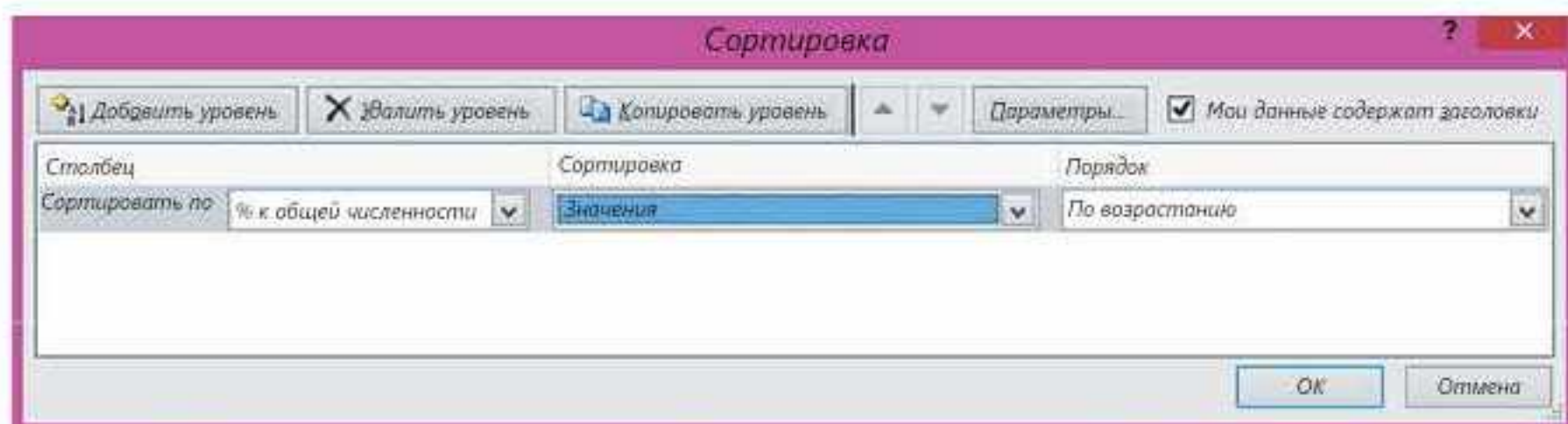


Рис. 3.25. Настраиваемая сортировка

В диалоговом окне **Сортировка** выбираются:

- столбец (поле);
- сортировка (значения) по полю – % к общей численности 2015;
- порядок (сортировка по возрастанию или убыванию).

На рис. 3.26 отображен результат отсортированной таблицы.

Области, города	Численность молодежи, тыс. 2013	Численность молодежи, тыс. 2014	Численность молодежи, тыс. 2015	% к общей численности 2013	% к общей численности 2014	% к общей численности 2015
Северо-Казахстанская	125098	120677	116212	21,6	21	20,3
Павлодар	175801	170439	164465	23,5	22,6	21,8
Восточно-Казахстанская	327564	316595	304972	23,5	22,7	21,9
Акмолинская область	173451	169558	165154	23,7	23,1	22,4
Караганды	331018	322431	314966	24,3	23,5	22,9
Костанай	215122	208279	201440	24,4	23,6	22,9
Алматинская область	496710	486710	450795	25,5	24,5	23,5
Тараз	271018	265921	260631	25,3	24,5	23,7
Западно-Казахстанская область	157654	153930	149595	25,5	24,7	23,7
Республика Казахстан	4376100	4293353	4206300	25,9	25	24,1
Атырау	148295	146263	144152	26,7	25,8	24,8
Нур-Султан	228589	218589	213350	28,9	26,8	25
Кызылорда	193377	191797	189637	26,6	25,9	25,2
Мангыстау	154944	153916	152652	27,3	26,2	25,2

Рис. 3.26. Результат отсортированной таблицы



Поэкспериментируйте.

Создайте базу данных «Химические элементы» по рис. 3.27.

1. Какой будет порядок строк в таблице после сортировки по возрастанию в поле **Год открытия** + **Автор**?
2. Произведите сортировку по полю **Название** по убыванию и запишите порядок записей.
3. Установите фильтр и выберите только те химические элементы, которые были открыты во **Франции**.
4. Проведите фильтрацию по нескольким полям.
5. С помощью **Настраиваемой сортировки** отсортируйте данные по двум полям **Название** и **Место открытия**. Проанализируйте полученный рис. 3.28.

	A	B	C	D	E	F
	№	Название	Символ	Год открытия	Автор	Место открытия
1						
2	1	Америций	Am	1945	Г. Сиборг	США
3	2	Дубний	Db	1970	Г.Н.Флёров	СССР
4	3	Германий	Ge	1886	К. Винклер	Германия
5	4	Полоний	Po	1898	Склодовская-Кюри	Франция
6	5	Рутений	Ru	1844	К.Клаус	Россия
7	6	Галлий	Ga	1875	Ф. Лекон де Буабодран	Франция
8	7	Водород	H	1766	Кавендиш	Англия
9	8	Радий	Ra	1998	Склодовская-Кюри	Франция

Рис. 3.27. База данных «Химические элементы»

	A	B	C	D	E	F
	№	Название	Символ	Год открытия	Автор	Место открытия
2	7	Водород	H	1766	Кавендиш	Англия
3	3	Германий	Ge	1886	К. Винклер	Германия
4	5	Рутений	Ru	1844	К.Клаус	Россия
5	2	Дубний	Db	1970	Г.Н.Флёров	СССР
6	1	Америций	Am	1945	Г. Сиборг	США
7	4	Полоний	Po	1898	Склодовская-Кюри	Франция
8	6	Галлий	Ga	1875	Ф. Лекон де Буабодран	Франция
9	8	Радий	Ra	1998	Склодовская-Кюри	Франция

Рис. 3.28. Сортировка по двум полям

	A	B	C	D	E	F
1	№	Названи	Символ	Год открытия	Автор	Место открытия
2	5	Америций	Am	1945	Г. Сиборг	США
3	1	Водород	H			ия
4	7	Галлий	Ga			ИТАЛИЯ
5	2	Германий	Ge			Германия
6	4	Дубний	Db			Россия
7	6	Полоний	Po			Польша
8	8	Радий	Ra			Франция
9	3	Рутений	Ru	1844	К.Клаус	Россия

1	№	Названи	Символ	Год открытия	Автор	Место открытия
2	1	Водород	H	1766	Кавендиш	Англия
3	2	Германий	Ge	1886	К. Винклер	Германия

Сортировка по двум полям

- Проанализируйте полученную базу данных после сортировки по двум полям.
- Почему записи расположились в порядке: 5, 1, 7, 2, 4, 6, 8, 3?



Синтез

Оценивание



10. Примените к базе данных «Химические элементы» **Автофильтр**, выделив весь диапазон таблицы.
- В раскрывшемся списке по полю **Автор** установите сортировку от минимального к максимальному.
 - Затем в поле **Место открытия** установите **Текстовые фильтры** начинается с... буквы **А**.

	A	B	C	D	E	F	G	H	I
1	N	Названи	Симво	Год открытия	Автор	Место открытия			
2	5	Америций	Am	1945	Г. Сиборг				
3	4	Дубний	Db	1970	Г.Н.Флёров				
4	2	Германий	Ge	1886	К. Винклер				
5	3	Рутений	Ru	1844	К.Клаус				
6	1	Водород	H	1766	Кавендиш				
7	6	Полоний	Po	1898	Склодовская-Кю				
8	8	Радий	Ra	1998	Склодовская-Кю				
9	7	Галлий	Ga	1875	Ф. Леон де Буа				
10									
11									

Автофильтр для базы данных Химические элементы

11. Заполните таблицу, содержащую информацию о планетах Солнечной системы, согласно рисунку «Солнечная система».

Планета	Период	Расстояние	Диаметр	Масса	Спутники
Венера	0,615	108	12	4,86	0
Меркурий	0,241	58	4,9	0,32	0
Земля	1	150	12,8	6	1
Плутон	247,7	5900	2,8	0,1	1
Нептун	164,8	4496	50,2	103,38	2
Марс	1,881	288	6,8	0,61	2
Уран	84,01	2869	49	87,2	14
Юпитер	11,86	778	142,6	1906,98	16
Сатурн	24,96	1426	120,2	570,9	17

Солнечная система

Единицы измерения, используемые в таблице:

- период обращения по орбите, в земных годах;
- среднее расстояние от Солнца, в млн. км;
- экваториальный диаметр, в тыс. км;
- масса, в 10^{24} кг.

1) С использованием **Автофильтра** осуществить поиск планет, начинающихся на букву С или букву Ю с массой менее $600 \cdot 10^{24}$ кг.

2) Выполните команды **Данные** → **Фильтр** → **Автофильтр**. Нажмите на кнопку в поле **Планета**. Выберите пункт **Условие**.

– В диалоговом окне задайте критерий отбора: **равно с*** или **равно ю***.

– Нажмите на кнопку на поле **Масса**. Выберите пункт **Условие**.

– В диалоговом окне задайте критерий: **меньше 600**.

– Выполните команды меню **Данные** → **Фильтр** → **Отобразить все**.

3) Осуществите поиск планет, имеющих экваториальный диаметр менее 50 тыс. км и массу менее $50 \cdot 10^{24}$ кг.

4) Осуществите поиск планет, находящихся от Солнца на расстоянии не менее 100 млн. км, имеющих массу в диапазоне от $3 \cdot 10^{24}$ кг до $600 \cdot 10^{24}$ кг, а также не более 2 спутников.



Поворот листа или замена строк на столбцы

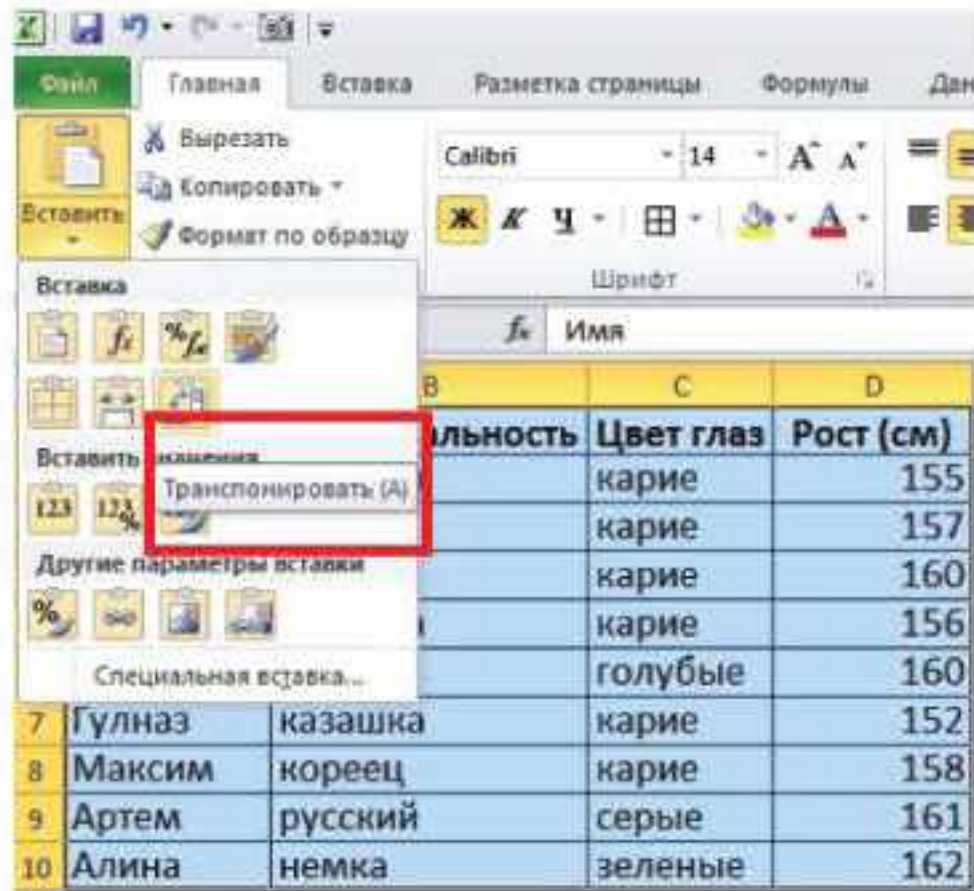
Допустим, при проектировании структуры таблицы вы допустили ошибку. Вам необходимо заголовки столбцов разместить по строкам или строки по столбцам. Для этого в Excel предусмотрена функция, позволяющая осуществить «поворот» листа на 90 градусов. При повороте содержимое строк перемещаются в столбцы.

	А	В	С	Д
1	Имя	Национальность	Цвет глаз	Рост (см)
2	Айнура	казашка	карие	155
3	Айдар	казах	карие	157
4	Серик	казах	карие	160
5	Мадина	казашка	карие	156
6	Анна	русская	голубые	160
7	Гулназ	казашка	карие	152
8	Максим	кореец	карие	158
9	Артем	русский	серые	161
10	Алина	немка	зеленые	162

Исходная таблица

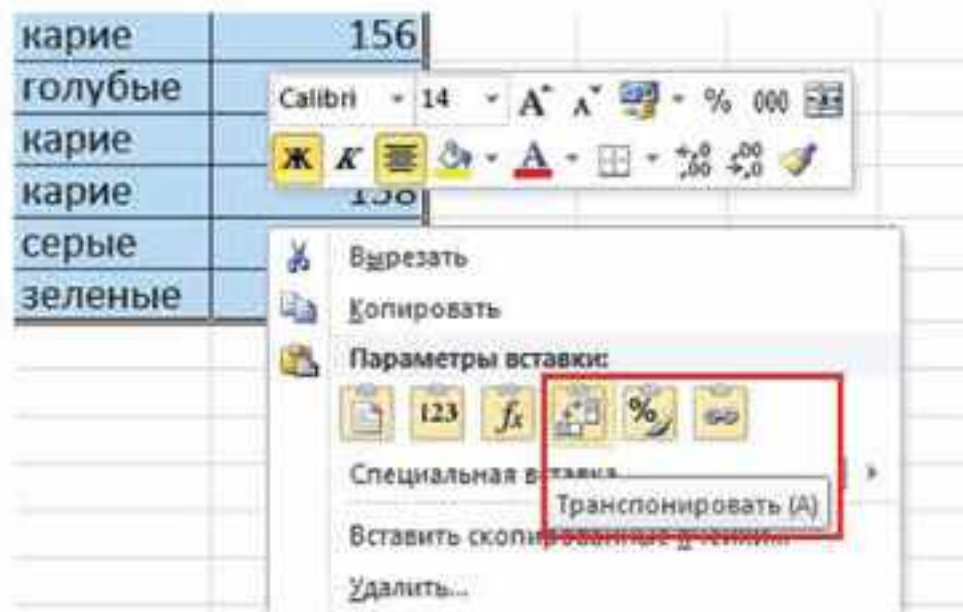
Алгоритм поворота таблицы:

1. Выделите ячейки с данными в диапазоне **A1:D10**.
2. Скопируйте данные в буфер обмена любым способом, например **Ctrl + C**.
3. Перейдите к пустому листу или свободному пространству текущего листа и вставьте данные комбинацией клавиш **Ctrl + V** или другим способом **Вставить** с вкладки **Главная**.
4. В меню **Параметры Вставки** из вкладки **Главная** выберите вариант **Транспонировать**.



Транспонирование из главного меню

5. Можете использовать как вариант поворота контекстное меню.



Вставка с транспонированием из контекстного меню

При этом форматирование сохраняется, а формулы изменяются в соответствии с новым положением ячеек.

	L	M	N	O	P	Q	R	S	T	U
1	Имя	Айнура	Айдар	Серик	Мадина	Анна	Гулназ	Максим	Артем	Алина
2	Национальность	казашка	казах	казах	казашка	русская	казашка	кореец	русский	немка
3	Цвет глаз	карие	карие	карие	карие	голубые	карие	карие	серые	зеленые
4	Рост (см)	155	157	160	156	160	152	158	161	162

Результат после поворота

3.5. Работа с базой данных

Вы научитесь:

- создавать базу данных в электронной таблице;
- осуществлять поиск, сортировку и фильтрацию данных.

Ключевые слова

Форматировать как таблицу

Таблица с заголовками

Пользовательский автофильтр

Кесте ретінде форматтау

Тақырыптар кестесі

Пайдаланушылық автофильтр

Format as table

Table with headlines

Custom
AutoFilter



Дайте общую трактовку ключевых слов и предположите, как они будут фигурировать в последующем тексте по теме.



Для работы с большим количеством информации применяется реляционная база данных. В таких базах данных отношения между информацией в таблицах четко определены. Это позволяет быстро находить и получать информацию, отображать одну и ту же информацию различным образом.



Создайте базу данных **БД_Продажи** в электронной таблице Excel. Таблица содержит поля: **Код продавца, Фамилия и имя, Адрес, Город, 1 квартал, 2 квартал, 3 квартал, 4 квартал – Формат ячеек... – денежный – обозначение: казахский.**



Шаг за шагом

После создания таблицы необходимо придать ей красивый вид. Для этого выделите весь диапазон таблицы вместе с заголовками. Перейдите во вкладку **Главная** в группу **Стили**. Нажмите кнопку **Форматировать**



Поэкспериментируйте.

1. Добавьте еще две записи в форме данных.
2. Выполните сортировку по возрастанию по полю таблицы 4 квартал.
3. Для поиска данных сформулируйте запросы с подстановочными знаками.
4. Выполните фильтрацию данных по выделенным ячейкам. Допустим, что в таблице нужно оставить только фамилии тех, кто работает в г. Нур-Султан.



Выполните пользовательский автофильтр согласно рис. 3.31.

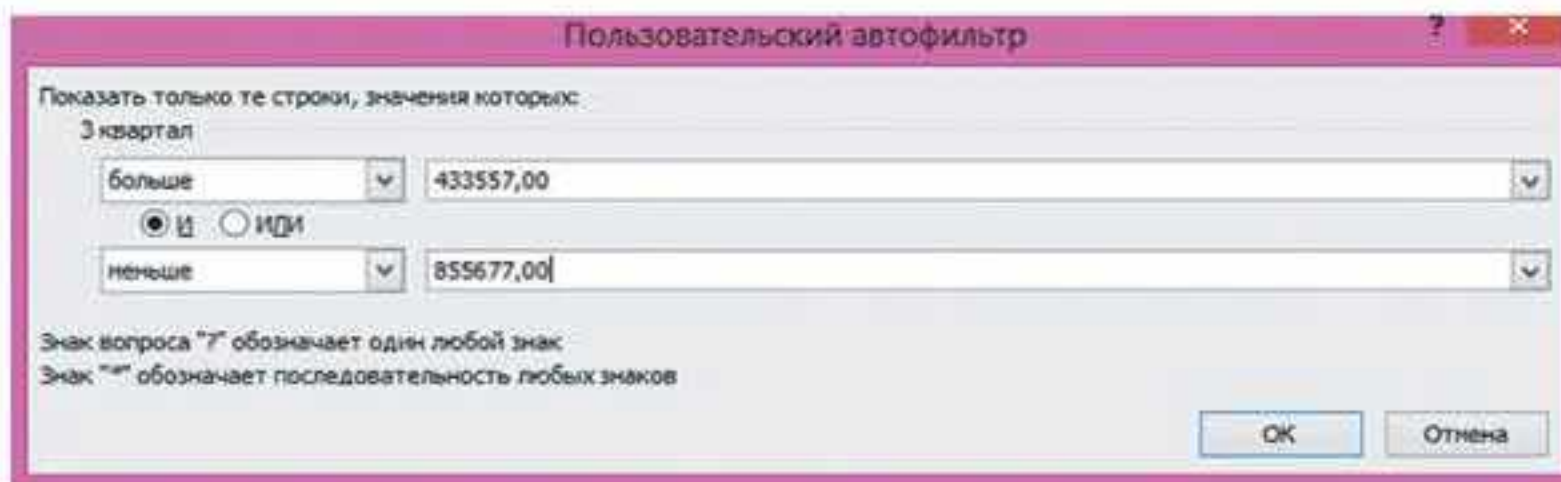


Рис. 3.31



Знание и понимание

1. На приведенном ниже рисунке изображен кроссворд. Составьте вопросы к словам, размещенным по горизонтали и вертикали.



Кроссворд Базы данных

2. Составьте и запишите свою версию рассказа, употребив все предложенные ключевые термины по теме.



Применение

Анализ

3. Представьте себе, что вы менеджер супермаркета. Вам необходимо создать базу данных овощей и фруктов по полям:

- наименование товара,
- количество в килограммах,
- цена за 1 килограмм в тенге (укажите актуальные цены в вашем регионе),
- стоимость.

Используйте формулу для подсчета стоимости товара.

Выполните сортировку по возрастанию по полю **Товары**.

Выведите информацию о яблоках (**Количество**, **Цена за 1 кг**), а также **Стоимость** всей продукции.

Составьте **Автофильтр** для фильтрации данных по **Цене** больше 500 тенге и меньше 1000 тенге.

4. Создайте в электронной таблице Excel таблицу **Части света** из текста «Какая часть света наибольшая как по площади, так и по численности населения?».

«Из шести основных частей света на нашей планете Азия является самой большой, как по площади, так по численности населения. Она занимает площадь 44,58 млн. км² и насчитывает более 4 млрд. человек. Второй по величине частью света с площадью 42,55 млн. км² является Америка (включая Северную и Южную Америки), а далее по убыванию находятся Африка (30,22 млн. км²), Антарктида (14,11 млн. км²), Европа (10,18 млн. км²), Австралия и Океания (8,51 млн. км²). Если распределять части света по численности населения, то второе место принадлежит Африке (1,1 млрд. чел.), а Америка с населением 953,7 млн. чел. занимает третью позицию.

Затем следуют Европа (742,5 млн. чел.), Австралия и Океания (24,2 млн. чел.), и наконец, Антарктида (около 5000 временного населения).

Если объединить Европу и Азию в одну часть света, известную как Евразия, то она будет далеко впереди, как по размеру, так и по численности населения».

1) Отформатируйте таблицу с помощью стилей. В заголовке таблицы введите информацию в две строки с помощью клавиши Alt+Enter.

Части света	Территория, кв. км	Население, чел.	Плотность, чел./кв. км
Азия	44 580 000	4 000 000 000	?

- Для поля **Часть света** выбираете тип данных – текстовый.
 - В полях **Территория** и **Население** тип данных – числовой, 0 десятичных знаков, установить разделитель групп разрядов.
 - В поле **Плотность** тип данных – числовой и один десятичный знак. Выполните различные действия в базе данных с помощью формул, функций электронной таблицы Excel.
- 2) Расчет плотности.
 - 3) Сведения по заданной части света.
 - 4) Какая часть света имеет наибольшую/наименьшую плотность, территорию или население.
 - 5) Общая площадь суши.
 - 6) Количество населения на Земле.
 - 7) Насколько больше населения в одной части света по сравнению с другой?
 - 8) Сколько частей света начинаются на букву А?
 - 9) У какой части света самое длинное название?



Синтез

Оценивание

5. Имеется табличная база данных «Питательная ценность продуктов».

Питательная ценность продуктов

№	Наименование продукта	Белки	Жиры	Углеводы	Количество калорий
1	Хлеб ржаной	5,5	0,6	39,3	190,0
2	Говядина	16,0	4,3	0,5	105,0
3	Судак свежий	10,4	0,2	0	44,0
4	Картофель свежий	1,0	0,1	13,9	63,0
5	Капуста свежая	0,9	0,1	3,5	20,0
6	Белые грибы	33,0	13,6	26,3	224,2
7	Рыжики соленые	21,85	3,75	47,75	183,7

1. Определите ключевое поле таблицы:

- 1) Наименование продукта.
- 2) Белки.

- 3) Жиры.
- 4) Белки, Жиры, Углеводы.
- 5) Количество калорий.

2. Сформулируйте условие отбора, позволяющее получить наименование продуктов, калорийность которых менее 100, не содержащих углеводов:

- 1) Количество калорий > 100 и Углеводы $= 0$.
- 2) Количество калорий > 100 или Углеводы $= 0$.
- 3) Количество калорий < 100 и Углеводы $= 0$.
- 4) Количество калорий < 100 или Углеводы > 0 .
- 5) Количество калорий < 100 или Углеводы $= 0$.

3. Запишите порядок строк в таблице после сортировки по возрастанию в поле Количество калорий:

- 1) 1, 2, 3, 4, 5, 6.
- 2) 5, 4, 1, 3, 2, 7, 6.
- 3) 3, 2, 5, 4, 6, 1, 7.
- 4) 4, 5, 3, 1, 7, 2, 6.
- 5) 5, 3, 4, 2, 7, 1, 6.

4. Какие записи удовлетворяют условию отбора Белки > 15 и Жиры < 10 и Количество калорий > 100 :

- 1) 2, 6, 7;
- 2) таких нет;
- 3) 6, 7;
- 4) 2, 7;
- 5) 3, 4, 5?

5. Произведите сортировку по полю Количество калорий + Жиры по возрастанию и запишите порядок записей.

6. Аналитическое задание «Доходы и расходы».

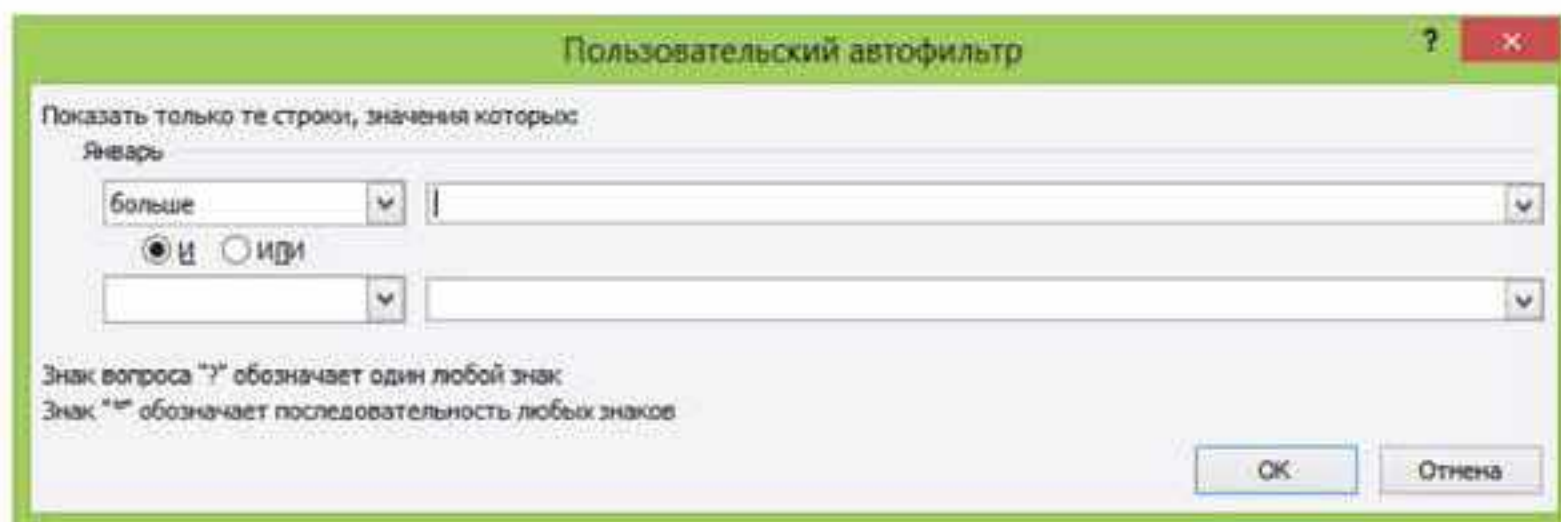
Ниже приведена таблица «Доходы и расходы», созданная в электронной таблице Excel.

Вопросы для исследования

1. Какие поля нужно добавить в таблицу, чтобы в ней отображались расчеты за год?
2. Какие формулы будут находиться в ячейках C22, D22, C24, D24?
3. Для чего используются фильтры в таблице?

Доходы и расходы						
Доходы	Январь	Февраль	Всего	Среднее (мес)	Среднее (год)	
Зарплата на карточку	100000	80000	180000	90000	15000	
Итого	100000	80000	180000	90000	15000	
Расходы	Январь	Февраль	Всего	Среднее (мес)	Среднее (год)	
Коммунальные услуги	15000		15000	15000	1250	
Еда	10000		10000	10000	833	
Проезд	3000		3000	3000	250	
Интернет	6500		6500	6500	542	
Сотовая связь	1300		1300	1300	108	
Одежда	7000		7000	7000	583	
Хозяйственные расходы	4500		4500	4500	375	
Покупка техники	0		0	0	0	
Парикмахерская	2500		2500	2500	208	
Обучение	3000		3000	3000	250	
Подарки и дни рождения	5000		5000	5000	417	
Прочие	3000		3000	3000	250	
Итого	60800	0	60800	30400	5067	
Остаток	39200	80000	119200	59600	9933	

4. Выберите **Числовые фильтры**, а затем **больше...**, появится **Пользовательский автофильтр** (см. рис. ниже). Составьте условие, чтобы в таблице отображались только те значения, которые больше 20 000.



5. Проанализируйте, какие формулы применяются в таблице, находящиеся в диапазоне **F3:H24**.
6. Оцените таблицу «Доходы и расходы».
7. Предложите свои идеи по оптимизации и модернизации таблицы «Доходы и расходы».



Современные базы данных

В основу Автоматизированных информационных систем (АИС) положены базы данных, которые появились в 60-х годах XX века. Первоначально базы данных появились в военной промышленности, а затем перешли в бизнес.

Первым базам данных нашли применение в химии, ядерной физике, космонавтике и других науках, требовавших систематического подхода к работе с информацией.

Последующая эволюция компьютеризации привела человечество к тому, что базы данных оказались в разработке фактически во всех областях занятости человека и стали повседневно использоваться на разных экономических объектах: от сельского хозяйства до финансовых систем.

Последним нововведением применения баз данных стала всемирная сеть Интернет, которая по всей сути является самой крупной и обширной базой данных.

Мы рассмотрим несколько приложений нового поколения, определяющих потребности в средствах разработки баз данных и возможностях их применения в современном мире.

1. Применение баз данных в системе наблюдения Земли. Система наблюдения Земли (EOS – Earth Observing System) представляет собой множество спутников, которые запускает NASA начиная с 1998 года. Их назначение – сбор информации, необходимой для исследователей, занятых изучением долгосрочных тенденций состояния атмосферы, океанов, земной поверхности. Спутники поставляют информацию в объеме 1/3 Пбайт (Petabyte – 10^{15} байт) в год.

В дальнейшем предполагается, что доступ к ней будут иметь даже школьники, которые смогут знакомиться с моделями формирования погодных условий, с воздействием вулканических явлений и т. п.

2. В настоящее время все большее распространение находит использование и применение баз данных в военной отрасли. В связи с применением в военном деле элементов робототехники и искусственного интеллекта крайне актуальным представляется интеграция систем баз данных в управление отдельными боевыми машинами и создание комплексов, направленных на взаимодействие их между собой.

3. В издательском бизнесе в ближайшем будущем ожидается ряд глубоких перемен. Становится возможным хранение книг и статей в электронном виде и их оперативная доставка потребителям по высокоскоростным сетевым каналам. Само понятие публикации существенно расширяется – документ может содержать графические, аудио- или видеовключения, аннотацию, другие сопроводительные элементы.

4. Внедрение современных информационных технологий в области здравоохранения окажет кардинальное воздействие на такие характеристики медицинского обслуживания, как стоимость, качество, повсеместная доступность.

5. Базы данных в системе электронной коммерции.

Итоговые задания по разделу «Базы данных»

1. Базы данных – это:

- а) совокупность электронных таблиц и всего комплекса аппаратно-программных средств для их хранения; изменения и поиска информации; для взаимодействия с пользователем;
- б) организованная совокупность данных, предназначенная для длительного хранения во внешней памяти компьютера и постоянного применения;
- в) программное обеспечение, управляющее хранением и обработкой данных;
- г) настраиваемые диалоговые окна, сохраняемые в компьютере в виде объектов специального типа;
- д) системное программное обеспечение, управляющее системой.

2. По характеру хранимой информации базы данных бывают:

- а) иерархические;
- б) централизованные;
- в) фактографические;
- г) локальные;
- д) распределенные.

3. По способу хранения информации базы данных бывают:

- а) фактографические;
- б) распределенные;
- в) табличные;
- г) документальные;
- д) сетевые.

4. Запись базы данных – это:

- а) столбцы таблицы, образующие структуру базы данных;
- б) набор данных об одном объекте;
- в) строки таблицы, содержащие определенную информацию;
- г) заголовок таблицы базы данных;
- д) строка заголовка реляционной таблицы.

5. Поле, значение которого не повторяется в различных записях, называется:

- а) первичным ключом;
- б) типом поля;
- в) внешним ключом;

- d) главной записью;
- e) составным ключом.

6. Какие данные могут быть первичным ключом базы данных сотрудников фирмы:

- a) номер дома;
- b) цвет волос;
- c) рост, вес;
- d) цвет глаз;
- e) номер паспорта?

7. В зависимости от структурирования информации базы данных бывают:

- a) централизованные;
- b) документальные;
- c) сетевые;
- d) распределенные;
- e) иерархические.

8. В табличных базах данных полем называется:

- a) столбцы таблицы, образующие структуру базы данных;
- b) строка заголовка реляционной таблицы;
- c) набор данных об одном объекте;
- d) строки таблицы, содержащие определенную информацию;
- e) заголовок таблицы базы данных.

9. Укажите количество полей и записей в базе данных «Учащиеся».

| *Фамилия* | *Имя* | *Школа* | *Класс* | *Адрес* | *Год рождения* | - | *Хасанов* |
Аят | *ШГ №36* | *9-А* | *Алматы* | *2003* | | *Фиоктистов* | *Иван* | *ОШ №106* | *9-Б* |
 | *Алматы* | *2003* |

- a) 2 поля и 6 записей;
- b) 1 поле и 6 записей;
- c) 6 полей и 2 записи;
- d) 6 полей и 6 записей;
- e) 6 полей и 1 запись.

10. Определите, о каком типе данных идет речь: «...используется для хранения символьных или числовых данных, не требующих вычислений»:

- a) дата/время;
- b) логический;
- c) текстовой;
- d) числовой;
- e) денежный.

11. Определите, о каком типе данных идет речь: «...используется для хранения числовых данных, используемых в математических расчетах»:

- a) числовой;
- b) логический;
- c) текстовой;
- d) дата/время;
- e) денежный.

12. Тип поля реляционной базы данных определяется:

- a) именем поля;
- b) именем записи;
- c) типом ключа;
- d) типом данных;
- e) именем ячейки.

13. Определите, о каком типе данных идет речь: «...используется для хранения денежных величин»:

- a) логический;
- b) дата/время;
- c) текстовой;
- d) числовой;
- e) денежный.

14. Для чего применяется сортировка в базах данных:

- a) отбор записей, удовлетворяющих условиям поиска;
- b) вывод на печать упорядоченных записей;
- c) упорядочение записей по значениям одного или нескольких полей;
- d) для группировки различных данных;
- e) быстрый способ поиска данных для работы с ними?

15. Перечислите достоинства табличных баз данных:

- a) возможность видеть одновременно несколько записей
- b) содержит большое количество полей
- c) легко просматривать и редактировать
- d) возможность видеть одну только запись
- e) самая простая структура, все другие типы баз данных используют их

16. Установите соответствие.

- | | |
|--------------------------------------------------------------|-------------------|
| 1) в виде многоуровневой структуры; | a) сетевые; |
| 2) набор взаимосвязанных таблиц; | b) иерархические; |
| 3) набор узлов, в котором каждый может быть связан с каждым. | c) реляционные. |

17. Для чего используется фильтрация в базах данных:

- a) отбор записей, удовлетворяющих заданным условиям;
- b) вывод на печать упорядоченных записей;
- c) нахождение однородных данных для всех объектов;
- d) упорядочение записей по значениям одного или нескольких полей;
- e) для группировки различных данных?

18. Какие поля должны быть отражены в базе данных, описывающей хобби ваших одноклассников, если база данных позволяет получить ответы на следующие вопросы:

- возраст всех детей, увлекающихся компьютером;
- имена девочек, увлекающихся пением;
- фамилии мальчиков, увлекающихся хоккеем;

- a) имя, пол, хобби, возраст;
- b) фамилия, пол, хобби, возраст;
- c) фамилия, имя, возраст, хобби;
- d) имя, фамилия, пол, хобби, возраст;
- e) имя, пол, хобби, фамилия?

19. Сетевая база данных – это база данных:

- a) информация организована в виде прямоугольных таблиц;
- b) каждый элемент связан с каждым;
- c) один элемент считается главным, остальные подчиненными;
- d) записи расположены в произвольном порядке;
- e) принята свободная связь между элементами разных уровней.

20. Иерархическая база данных – это база данных, в которой:

- a) информация организована в виде прямоугольных таблиц;
- b) каждый элемент связан с каждым;
- c) один элемент считается главным, остальные подчиненными;
- d) записи расположены в произвольном порядке;
- e) принята свободная связь между элементами разных уровней.

Проекты к разделу III

1. Проект «География полезных ископаемых» (создание базы данных в MS Office Excel, поиск, сортировка и фильтрация информации).

2. Проект «Машиностроительные заводы Казахстана» (создание базы данных в MS Office Excel, поиск, сортировка и фильтрация информации).

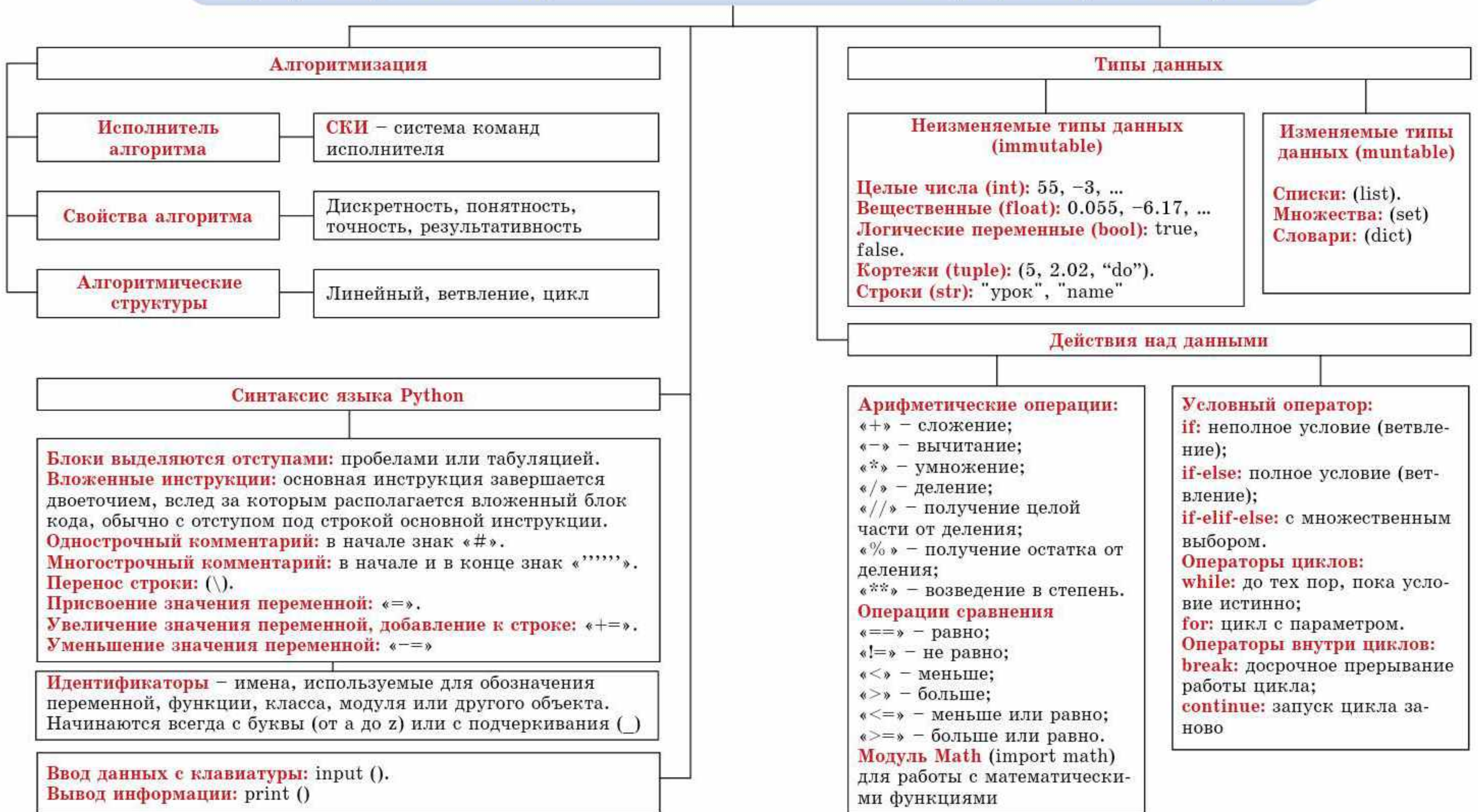
3. Проект «Демография в Казахстане» (создание базы данных в MS Office Excel, поиск, сортировка и фильтрация информации).

СИСТЕМА ОСНОВНЫХ

ПОНЯТИЙ РАЗДЕЛА IV

Программирование алгоритмов на

языке программирования Python



Раздел IV

ПРОГРАММИРОВАНИЕ АЛГОРИТМОВ НА ЯЗЫКЕ ПРОГРАММИРОВАНИЯ PYTHON (Пайтон)

4.1. Одномерный массив

Вы научитесь:

– создавать программы на языке программирования Python (пайтон) с использованием одномерных массивов.

Ключевые слова

Список

Массив

Одномерный массив

Элемент

Индекс

Тізім

Жиым

Бір өлшемді жиым

Элемент

Индекс

List

Array

One-dimensional array

Element

Index



Дайте общую трактовку ключевых слов и предположите, как они будут фигурировать в последующем тексте по теме.



Предположим, что вам необходимо обработать информацию о курсе тенге в долларах США с 20 по 31 декабря 2018 года (табл. 4.1).

Каким образом можно разместить данные на каждый день? Как можно обработать данные за несколько лет?



Приведем пример размещения данных из табл. 4.1 в отдельные переменные `day1`, `day2`, ..., `day11`.

Таблица 4.1

Курс тенге к доллару

Дата	Курс тенге к 1 доллару	Данные в отдельных переменных	Код программы	Результат
20.12.2018	372,49	day1 = 372.49; print (day1)	<pre>File Edit Format Run Options Win >>> day1 = 372.49; print (day1) day2 = 370.76; print (day2) day3 = 371.35; print (day3) day4 = 371.35; print (day4) day5 = 371.35; print (day5) day6 = 371.68; print (day6) day7 = 371.08; print (day7) day8 = 375.15; print (day8) day9 = 380.44; print (day9) day10 = 380.44; print (day10) day11 = 384.20; print (day11) >>></pre>	
21.12.2018	370,76	day2 = 370.76; print (day2)		372.49
22.12.2018	371,35	day3 = 371.35; print (day3)		370.76
23.12.2018	371,35	day4 = 371.35; print (day4)		371.35
24.12.2018	371,35	day5 = 371.35; print (day5)		371.35
25.12.2018	371,68	day6 = 371.68; print (day6)		371.35
26.12.2018	371,08	day7 = 371.08; print (day7)		371.68
28.12.2018	375,15	day8 = 375.15; print (day8)		371.08
29.12.2018	380,44	day9 = 380.44; print (day9)		375.15
30.12.2018	380,44	day10 = 380.44; print (day10)		380.44
31.12.2018	384.20	day11 = 384.20; print (day11)		380.44

Сколько переменных потребуется для размещения данных в программе?

Для каждой переменной (day1, day2, ..., day11) в памяти резервируется область, которая похожа на контейнер. В ней хранятся ссылки на значение (рис. 4.1), а не содержат его, как во многих языках программирования. Данный способ представления данных позволяет работать в программе с **отдельными переменными**.



Рис. 4.1. Контейнеры

Как можно обрабатывать информацию в **наборе переменных**? Для этого в языке программирования Python применяются **списки**, являющиеся аналогом термина «массив» в большинстве языков программирования. **Массив** может состоять из одной строки и нескольких столбцов, которые являются элементами массива. Такой массив называется **одномерным**.

Так как массивы в языке программирования Python представляют собой список элементов, то в дальнейшем при работе с одномерными массивами будем применять термин «списки».



Список (list) – это один из типов последовательностей в языке Python. В последовательностях хранятся упорядоченные наборы объектов, объединенные под общим именем.

Переменные, входящие в состав списка, называются элементами. Элементы списка заключены в квадратные скобки и разделены запятыми.



Зададим список с именем *s* (рис. 4.2), применяя данные из табл. 4.1:
`s = [372.49, 370.76, 371.35, 371.35, 371.35, 371.68, 371.08, 375.15, 380.44, 380.44, 384.20]`

```

File Edit Format Run Options Window Help
s = [372.49, 370.76, 371.35, 371.35, 371.35, 371.68, 371.08, 375.15, 380.44, 380.44, 384.20]
print(s)
  
```

Код программы

```

[372.49, 370.76, 371.35, 371.35, 371.35, 371.68, 371.08, 375.15, 380.44, 380.44, 384.2]
>>> |
  
```

Результат

Рис. 4.2. Ввод данных списком

Для эффективной работы с последовательностью элементов используется **индекс**. С его помощью описывают местоположение и обращение к элементу списка, заключенному в квадратные скобки. Нумерация индексов в списках начинается с нуля. Первому элементу последовательности присваивается индекс 0, второму – индекс 1, третьему – индекс 2, и т. д. Например: `s[0]=372.49`, `s[1]=370.76`, `s[2]=371.35`, ... `s[10]=384.20`.

Особенностью элементов списка в Python является то, что они могут быть разного типа. Например, в списке `lst=[5, 3.14, 'hello', 's']` применяются данные целого и вещественного типов, строки (рис. 4.3).

```

File Edit Format Run Options Window
lst=[5, 3.14, 'hello', 's']
print(lst)
  
```

```

Type "copyright", "credits"
>>>
===== RESTART: C:/Users/TAC/
[5, 3.14, 'hello', 's']
>>>
  
```

Рис. 4.3

Списки относятся к изменяемым типам. Это означает, что их содержимое можно добавлять, удалять и изменять без создания нового объекта. Существуют два способа создания пустых списков:

1) с вызовом класса `list`: `names = list ()`;

2) с использованием синтаксиса литералов в квадратных скобках `[]`:
`names = []`.

Как создавать заполненные списки? В языке программирования Python существует для этого специальная синтаксическая конструкция, которая называется генератором списков. Например, конструкция `[i for i in range(1,10)]`, заключенная в квадратные скобки, это генератор списка.

Внутри квадратных скобок можно выделить три части, отделенные друг от друга ключевыми словами `for` и `in`:

1) что делаем с элементом (добавляем в список);

2) что берем (элемент `i`);

3) откуда берем (из объекта `range`).

Функция `range ()` создает последовательность (диапазон) чисел. В качестве аргументов функция принимает начальное значение диапазона (по умолчанию 0), конечное значение (не включительно) и шаг (по умолчанию 1).

Разберем с помощью примера три части:

```
a = [5, -5, 2, -2, 7, 5, 3, 8]
```

```
b = [i**3 for i in a]
```

```
print(b)
```

```
[125, -125, 8, -8, 343, 125, 27, 512]
```

Берем каждый элемент из списка `a` и возводим его в генераторе списка в куб:

1) что делаем – возводим элемент в куб;

2) что берем – элемент;

3) откуда берем – из списка `a`.

Благодаря генератору списка запись программного кода становится короткой, чем при создании списка классическим способом. Приведем пример (табл. 4.2), в котором надо создать список, заполненный натуральными числами от 1 до 10.

Таблица 4.2

Способы создания списков

Классический способ	Способ – генератор списка
<pre>s = [] for i in range(1,10): s.append(i) print (s) [1, 2, 3, 4, 5, 6, 7, 8, 9]</pre>	<pre>s = [i for i in range(1,10)] print (s) [1, 2, 3, 4, 5, 6, 7, 8, 9]</pre>

Для эффективного решения разнообразных задач используются **функции** и **операции** для работы со списками. Рассмотрим некоторые из них (табл. 4.3).

Таблица 4.3

Специальные функции и операции для списков

Функции	Назначение	Примеры
append(x)	добавляет элемент x в конец списка	<pre>a = [1, 2, 3, 4, 5] a.append([6, 7, 8, 9, 10]) print (a)</pre> <p>Результат [1, 2, 3, 4, 5, [6, 7, 8, 9, 10]]</p>
a.extend(s)	добавляет все элементы списка s в конец списка a	<pre>a = [1, 2, 3, 4, 5] a.extend([6, 7, 8, 9, 10]) print (a)</pre> <p>Результат [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]</p>
sum(s)	возвращает сумму элементов в списке	<pre>s = [1,2,3,4,5] print(sum(s))</pre> <p>Результат 15</p>
Операции	Назначение	Примеры
lst1+lst2	Сложение или объединение списков – приписывание одного списка к другому	<pre>a = [9, 8, 7] b = [6, 5] c = a + b print(c)</pre> <p>В результате список: c = [9, 8, 7, 6, 5]</p>
lst * n	n кратное повторение списка lst (умножение списка lst на число n)	<pre>b = [6, 5] d = b * 3 print(d)</pre> <p>В результате список: d = [6, 5, 6, 5, 6, 5]</p>

Поэкспериментируйте.

Составьте программу создания и считывания списков. Напишите комментарии к каждой строке программы.

Введите код программы в IDLE – Сохраните – Запустите модуль – Выведите.








Шаг за шагом

Прежде всего нужно создать пустой список (не содержащий элементов, длины 0). В конец списка будут добавляться элементы с помощью метода `append`:

```
massiv=[]          # пустой список
for i in range(0,6): #
    a=input("Элемент массива: ")
    massiv.append(a)
print ("Исходный массив: ", massiv)
```

- С какой целью в программе используется оператор цикла `for`?
- Какую роль играют четыре пробела в программе перед операторами?
- Попробуйте убрать пробелы в программе. Каким будет результат программы?

  	<p style="text-align: center;">Знание и понимание</p> <ol style="list-style-type: none"> 1. Что такое <i>список</i>? 2. Может ли список содержать разнородные типы данных? 3. Для чего применяется имя списка? 4. Что такое <i>индекс</i>? Как начинается индексация элементов в списке? 5. Что такое <i>элемент списка</i>? Каким образом размещаются элементы в памяти? 6. Какова отличительная особенность списка в Python? 7. Какой список называется пустым? 8. Что является аналогом списка? 9. Как с помощью генератора списка можно создавать заполненные списки? 10. Для чего применяются методы в списках? Перечислите методы. 11. Составьте и запишите свою версию рассказа, употребив все предложенные ключевые термины по теме. 		
 	<table border="0" style="width: 100%;"> <tr> <td style="text-align: center; border: 1px solid black; border-radius: 10px; padding: 5px;">Применение</td> <td style="text-align: center; border: 1px solid black; border-radius: 10px; padding: 5px;">Анализ</td> </tr> </table> <ol style="list-style-type: none"> 12. Какой список будет выведен на экран после выполнения кода программы: <div style="border: 1px solid black; padding: 5px; margin: 10px 0;"> <pre>hi = ['bonjour', 'привет', 'hola', 'aloha', 'salem'] hi[1] = 'hello' print(hi)</pre> </div> 13. Что будет выведено на экран после выполнения блока функций? <div style="border: 1px solid black; padding: 5px; margin: 10px 0;"> <pre>print([4, 3] + [2]) print([0, 1] * 2)</pre> </div> 	Применение	Анализ
Применение	Анализ		

14. Дана программа в Python:

```
a = [] # пустой список
n = int(input()) # количество элементов в списке
for i in range(n):
    new_element = int(input()) # считывается очередной элемент
    a.append(new_element) # добавляется элемент в список
print(a)
```

– Введите код программы в **IDLE** – **Сохраните** – **Запустите модуль** – **Выведите**.

- Проанализируйте, как будет происходить считывание списка.
- Какое количество элементов в списке будет задано на входе?
- Сколько элементов списка будет находиться в отдельной строке?
- Куда будут добавляться элементы с помощью метода `append`?

15. Дан код программы:

```
numbers = [1, 2, 3, 4, 5]
strings = ['казахские', 'традиции:', 'асар', 'жылу', 'конакасы', 'конаккаде']
mylist = [numbers, strings]
print(mylist)
```

- 1) Сколько переменных созданы в программе?
- 2) Какие типы данных хранятся в списках?
- 3) Какой результат будет выведен на экране?
- 4) Сколько элементов будут храниться в третьем списке?



Синтез

Оценивание



16. Каким образом можно заполнить список числами, кратными 3 или 5, чтобы на экран был выведен список: [3, 5, 6, 9, 10, 12, 15, 18, 20]?

17. Ниже приведена программа, в которой генерируется список случайных целых чисел. Напишите комментарии к программе. Определите, что будет выведено на экран в результате выполнения данной программы.

```
import random
s = []
for i in range(10):
    s.append(int(random.random() * 100))
print(s)
chet = 0
nechet = 0
for i in s:
```

```

if i%2 == 0:
    chet += 1
else:
    nechet += 1
print ("Chet:", chet)
print ("Nechet:", nechet)

```

18. Оцените преимущества и недостатки использования списков при обработке большого набора данных.



Создание списка разными способами.

Приведем примеры создания списка разными способами.

1. Создание списка, заполненного одинаковыми элементами с использованием оператора повторения списка:

```
A = [0] * n
```

2. Создание списков, заполненных по сложным формулам с использованием генераторов, – выражения, позволяющие заполнить список некоторой формулой. Общий вид генератора следующий:

[выражение for переменная in список]

где переменная – идентификатор некоторой переменной, список – список значений.

2.1. Создать список, состоящий из n нулей с помощью генератора:

```
A = [0 for i in range (n)]
```

2.2. Создать список, заполненный квадратами целых чисел:

```
A = [i ** 2 for i in range(n)]
```

2.3. Заполнить список квадратами чисел от 1 до n с изменением параметров функции `range` на `range (1, n + 1)`:

```
A = [i ** 2 for i in range (1, n + 1)]
```

2.4. Список, заполненный случайными числами от 1 до 9 (используя функцию `randint` из модуля `random`):

```
A = [randint(1, 9) for i in range(n)]
```

2.5. Список, считанный стандартным вводом, – ввод числа элементов списка, которое будет использовано в качестве аргумента функции `range`, затем ввод заданного количества строк:

```
A = [input() for i in range(int(input()))].
```

4.2. Поиск элемента с заданными свойствами

Вы научитесь:

– создавать программы на языке программирования Python (пайтон) с использованием одномерных массивов.

Ключевые слова

Поиск элемента
Алгоритм поиска
Линейный поиск

Элементті іздеу
Іздеу алгоритмі
Сызықтық іздеу

Finding an item
Search Algorithm
Linear search



Дайте общую трактовку ключевых слов и предположите, как они будут фигурировать в последующем тексте по теме.



Очень часто при обработке информации необходимо решать задачи по поиску данных с определенными свойствами. Поиск информации является одним из наиболее часто выполняемых алгоритмов. Задачи по поиску данных разнообразны. Например, поиск данных разыскиваемого человека, телефонного номера, необходимого товара, поиск школы, угадывание числа в играх и т. д.



Поиск – это обработка некоторого множества данных с целью выявления подмножества данных, соответствующего критериям поиска.



Подумайте и приведите примеры использования алгоритмов поиска информации из вашей практики.



К основным алгоритмам, используемых для поиска элементов в списке, относятся (рис. 4.4):

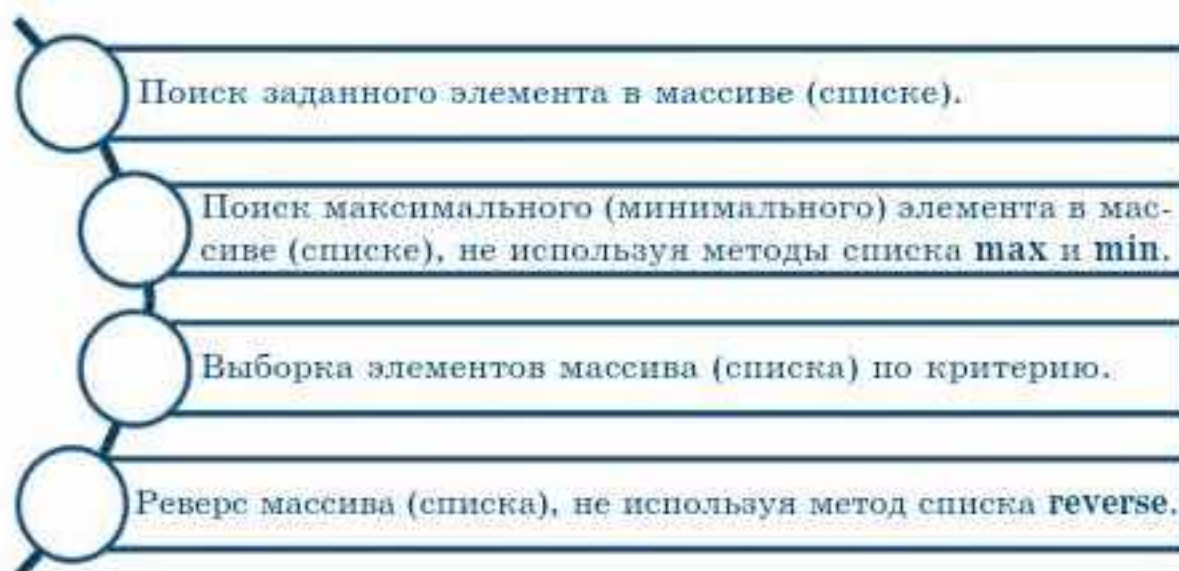


Рис. 4.4. Методы поиска элементов в списках

Самый простой способ поиска элементов списка – **линейный**, или **последовательный**. При линейном поиске последовательно проверяют все элементы списка и сравнивают их с ключевым значением.



Рассмотрим, как осуществляется поиск элементов в программах на Python.



1. Код программы поиска заданного элемента в списке.

Код программы	Алгоритм выполнения
<pre># задаем список arr = [100, -355, 111, -765, 123, -99] # исходное значение point = 111 # длина списка или количество элементов l = len(arr) for i in range(l): if arr[i] == point: print ("заданный элемент – содержит") break else: print ("заданный элемент – не содержит")</pre> <p>Результат заданный элемент – содержит</p>	<p>Для проверки заданного элемента, входящего в список, применим цикл for. Каждый элемент списка сравниваем со значением в переменной point (эталон). В условном операторе if проверяем условие. Если заданное число совпадает с элементом в списке, то выводится соответствующее сообщение: «заданный элемент – содержит», и мы выходим из цикла. Если элементы списка последовательно проверены и совпадения отсутствуют, то выполняются else и команда print ("заданный элемент – не содержит").</p>

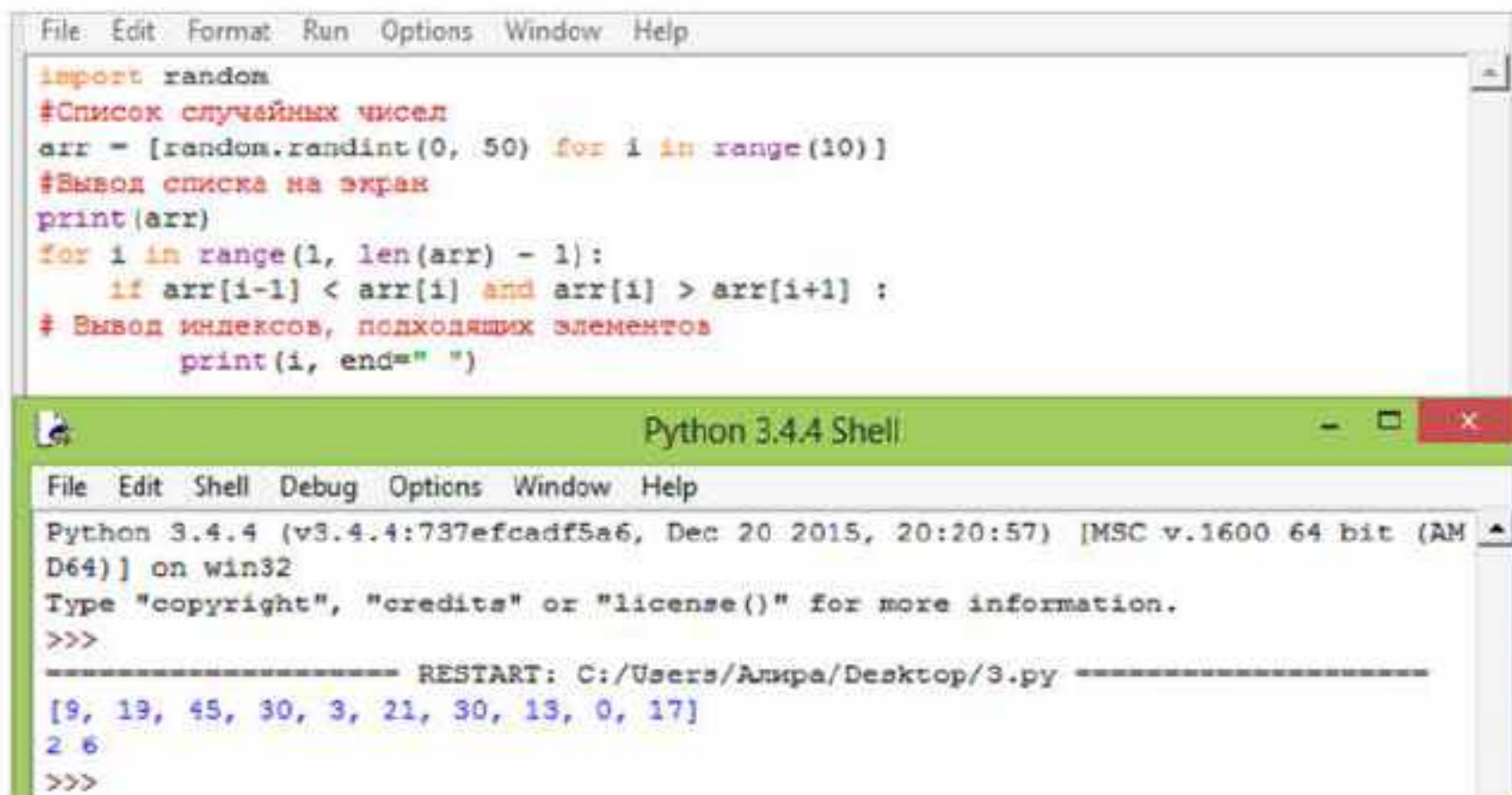
2. Поиск максимального (минимального) элемента в списке.

В жизни для решения задач часто необходимо найти **максимальный** (наибольший) или **минимальный** (наименьший) элемент массива. Например, определить рост самого высокого ученика в классе, какого числа была самая низкая температура зимой в Алматы, самую большую площадь территории среди евразийских стран. Ниже приведен пример, в котором выполняется поиск максимального элемента списка.

Код программы	Алгоритм выполнения
<pre># задаем список arr = [22, 8, 27, -55, 0, 15, 42, -5, 77, 99] # максимальный элемент равен arr[0] maximum = arr[0] for i in range(1, len(arr)): if arr[i] > maximum: maximum = arr[i] print(maximum)</pre> <p>Результат 99</p>	<p>Максимальный элемент сравнивается последовательно со всеми элементами списка со знаком > (больше). Если найдутся элементы списка больше arr [0], то значение максимума обновится. Иначе значение максимума останется равным arr [0].</p> <p>Чтобы найти минимальный элемент списка, необходимо знак поменять на < (меньше).</p>

3. Выборка элементов списка по критерию.

На языке программирования Python сложное условие **if** можно записать разными способами. Например, **if arr[i - 1] < arr[i] > arr[i + 1]**. Ниже на рис. 4.5 представлен результат программы.



```
File Edit Format Run Options Window Help
import random
#Список случайных чисел
arr = [random.randint(0, 50) for i in range(10)]
#Вывод списка на экран
print(arr)
for i in range(1, len(arr) - 1):
    if arr[i-1] < arr[i] and arr[i] > arr[i+1] :
# Вывод индексов, подходящих элементов
    print(i, end=" ")

Python 3.4.4 Shell
File Edit Shell Debug Options Window Help
Python 3.4.4 (v3.4.4:737efcadf5a6, Dec 20 2015, 20:20:57) [MSC v.1600 64 bit (AMD64)] on win32
Type "copyright", "credits" or "license()" for more information.
>>>
----- RESTART: C:/Users/Алира/Desktop/3.py -----
[9, 19, 45, 30, 3, 21, 30, 13, 0, 17]
2 6
>>>
```

Рис. 4.5

Также можно записать сложное условие **if**, применив логический оператор **and** (рис. 4.6):

if arr[i-1] < arr[i] and arr[i] > arr[i+1].

```

2.py - C:/Users/Алира/Desktop/2.py (3.4.4)
File Edit Format Run Options Window Help
import random
#Список случайных чисел
arr = [random.randint(0, 50) for i in range(10)]
#Вывод списка на экран
print(arr)
for i in range(1, len(arr) - 1):

```

Рис. 4.6

Мы видим, что разные способы записи сложного условия `if` в Python выполняют одну и ту же функцию задания.

Код программы	Алгоритм выполнения
<pre> import random #Список случайных чисел arr = [random.randint(0, 50) for i in range(10)] #Вывод списка на экран print (arr) for i in range(1, len(arr) - 1): if arr[i-1] < arr[i] and arr [i] > arr[i+1]: # Вывод индексов, подходящих эле- ментов print (i, end=" ") Результат [16, 23, 34, 23, 29, 13, 4, 19, 49, 25] 2 4 8 </pre>	<p>Вывести номера элементов списка, которые больше стоящих по соседству.</p> <p>Начиная со второго элемента массива <code>arr[1]</code> и до предпоследнего, сравниваем элементы списка с соседями.</p> <p>Если элемент списка подходит, то на экран в одну строчку выводятся его индексы.</p>

4. Кортежи.

Следующий тип данных (класс) – **кортеж (tuple)**. Кортеж условно называют неизменяемым «списком». К нему применимы многие списковые функции, кроме изменения. Кортежи используются, если элементы структуры данных не будут изменены в процессе работы программы.

С помощью кортежей можно присваивать значения одновременно двум переменным:

```
(x, y) = (10, 5) # результат не изменится, если убрать круглые скобки
print(x)
print(y)
```

Результат:

10

5

Кортеж нельзя изменить, но можно изменить список, входящий в кортеж:

```
kort = (1, [4, 5], '3')
print(kort[1])
```

Результат:

[4, 5]

```
kort[1][0] = '7'
print(kort)
```

Результат:

(1, ['7', 5], '3')

5. Реверс массива (списка), не используя метод списка reverse.



Код программы	Алгоритм выполнения
<pre>import random #Список случайных чисел arr = [random.randint (0, 50) for i in range (10)] #Вывод списка на экран print(arr) l = len(arr) for i in range(l//2): arr[i], arr[l - 1 - i] = arr[l - 1 - i], arr[i] print(arr)</pre>	<p>Принцип алгоритма заключается в следующем: мы меняем местами 0-й элемент с последним, 1-й – с предпоследним и т. д.</p> <p>В итоге количество таких обменов будет равно половине длины списка. Элементы поменяются местами по второму кругу и вернуться в первоначальное положение.</p> <p>Цикл выполняется 1/2 раз (10/2).</p>

Результат [50, 2, 42, 40, 2, 39, 26, 0, 44, 9] [9, 44, 0, 26, 39, 2, 40, 42, 2, 50]	
--------------------------------------------------------------------------------------------------	--

Кроме операций сложения и повторения списков, с которыми вы познакомились в § 4.1, используются **встроенные функции** для списков (табл. 4.4). Например, такие функции, как `min()` и `max()`, возвращают соответственно минимальный и максимальный элементы последовательностей. Метод `index()` возвращает индекс (номер) элемента в последовательности.

Таблица 4.4

Встроенные функции для списков

Функции	Назначение	Примеры
<code>len(list)</code>	вычисляет общую длину списка <code>list</code>	<code>print(len([1, 2, 3, 4, 5, 6, 7, 8, 9]))</code> Результат: 9
<code>max(list)</code>	возвращает элемент из списка <code>list</code> с максимальным значением	<code>print(max([1,2,3,4,5]))</code> Результат: 5
<code>min(list)</code>	возвращает элемент из списка <code>list</code> с минимальным значением	<code>print(min([1,2,3,4,5]))</code> Результат: 1
<code>index()</code>	возвращает индекс элемента <code>x</code> . Если элемент не найден, генерирует исключение <code>ValueError</code>	<code>a=[0.25, 111, -1, 111, 1, 1234.5, 555]</code> <code>print (a.index(1234.5)) # индекс</code> Код программы Результат 
<code>reverse()</code>	расставляет все элементы в списке в обратном порядке	<code>data = list[1,2,3,4,5,6,7,8,9]</code> <code>data.reverse()</code> <code>print(data)</code> Код программы Результат 

Продолжим знакомство с операциями проверки вхождения элементов в список (табл. 4.5), которые часто применяются при решении задач на языке программирования Python.

Операции работы со списком




Операция	Назначение	Примеры
<code>x in A</code>	Проверяет, содержится ли элемент в списке. Возвращает True или False	<code>print(5 in [1, 2, 3, 4, 5])</code> Результат: True
<code>x not in A</code>	Аналогично проверяет <code>not (x in A)</code>	<code>print(5 not in [1, 2, 3, 4, 5])</code> Результат: False

Задачи по программированию с использованием списков решаются, как правило, по следующему алгоритму (рис. 4.7).



Рис. 4.7. Алгоритм работы со списками

Если в процессе обработки значения элементов списка изменяются, то вывод осуществляется дважды: до и после обработки.

Знание и понимание

1. Что такое *поиск информации*?
2. Каким образом упорядочены данные в массивах?
3. Как называется самый простой способ поиска элементов в массиве?
4. Какие числа называют максимальными и минимальными?
5. Чем отличается условие поиска максимального и минимального элементов?
6. Какие встроенные функции используются для списков?
7. Почему встроенные функции облегчают написание программы?
8. Почему алгоритм работы со списками имеет одинаковую структуру?
9. Составьте и запишите свою версию рассказа, употребив все предложенные ключевые термины по теме.



Применение

Анализ

10. Дан код программы. Каким образом будет заполнен новый список в зависимости от значений исходного?

```
# исходный список
a = [17, -11, 18, 10, 0, 77, -1, 12, -5, 25]
# пустой список
b = []
# перебор элементов исходного списка
for i in a:
    if i > 0: # если элемент больше 0,
        b.append(1) # в список добавляется 1.
    elif i < 0: # если элемент меньше 0,
        b.append(-1) # в список добавляется -1.
    else: # если элемент равен 0,
        b.append(0) # в список добавляется 0.
# вывод исходного и производного списков
print(a)
print(b)
```

11. Дана строка $s = 'abc12d3e5f7g8'$, состоящая из цифр и букв. Необходимо из строки выбрать все цифры и сложить их в массив как числа $digits == [1, 2, 3, 5, 7, 8]$. Составьте программу с использованием цикла **for**.

12. Найдите произведение элементов списка, кратных 3 и оканчивающихся на 5. Если таких элементов нет, сообщите об этом.

```
import random
#Создадим список случайных чисел
arr = [random.randint(0, 50) for i in range(10)]
#Выведем его на экран
print(arr)
#При поиске произведения – присваиваем переменной значение 1
p = 1
for i in range(len(arr)):
    if arr[i] % 3 == 0 and arr[i] % 10 == 5:
        p *= arr[i]
#Если произведение равно единице, то таких элементов нет
if p == 1:
    print("Нет таких элементов")
else:
    print(p)
[46, 47, 10, 10, 20, 47, 15, 48, 6, 16]
720
[48, 22, 31, 28, 37, 35, 36, 29, 42, 44]
Нет таких элементов
```



Синтез

Оценивание



13. В ботаническом саду растут деревья разной высоты. Для исследования необходимо методом случайной выборки определить 10 деревьев, имеющих высоту от 1 до 50 метров. Затем, используя этот список, нужно осуществить поиск дерева с максимальной высотой и соответственно его индекс, а также дерева с минимальной высотой и его индекс. Составьте программу на языке программирования Python.

Алгоритм поиска максимального элемента в списке будет целого типа. Максимальный элемент можно хранить в переменной `arr[max]`, а его индекс – в переменной `max`. Простейший алгоритм поиска сводится к последовательному сравнению со значением `max` всех элементов массива `arr`, начиная со второго в цикле `for` с параметром `i`.

На каждом шаге цикла будем проверять условие `arr[i] > arr[max]`. Если оно выполнится (очередной элемент больше `max`), то значение этого элемента примем в качестве максимального `max`, а его индекс присвоим переменной `max=i`. После перебора всех элементов массива переменная `arr[max]` будет содержать значение максимального элемента массива, а переменная `i` – его индекс. Аналогично для минимального значения самостоятельно проверьте условие:

`if arr[i] < arr[min].`

14. Найдите произведение самого большого и самого маленького элементов списка.

15. Заполните список случайными числами и выполните реверс.

16. Найдите максимальный и минимальный рост 100 спортсменов в списке, в котором даны случайные числа.



Задачи для углубленного изучения

1) Задана строка: «Казахская степь занимает большую часть Северного и Центрального Казахстана, широким поясом растянувшись более чем на 2000 км между Каспийским морем на западе и горами Алтая на востоке». Удалите из строки все слова, начинающиеся на букву «м». Результат выведите на экран в виде строки.

2) Дан список случайных целых чисел. Замените все нечетные числа списка нулями, выведите их количество.

3) Список содержит положительные и отрицательные числа температуры в январе 2019 года в Алматы. Требуется положительные числа

поместить в один список, а отрицательные – в другой. Списки выведите на печать.

4) Напишите программу, которая запрашивает у пользователя две строки и формирует из них список. Если строки состоят только из чисел, то программа добавляет в середину списка сумму введенных чисел. Иначе добавляется строка, образованная из слияния двух введенных ранее строк. Итоговая строка выводится на экран.

5) Найдите произведение элементов списка, кратных 6 и оканчивающихся на 8. Если таких элементов нет, то сообщите об этом.

6) Напишите программу-игру. Компьютер загадывает случайное число, пользователь пытается его угадать. Пользователь вводит числа до тех пор, пока не угадает или не введет слово «Выход». Компьютер сравнивает число с введенным и сообщает пользователю, больше оно или меньше загаданного.

7) Дан произвольный текст. Определите в нем номер первого самого длинного слова.

8) Дан произвольный текст. Напечатайте все имеющиеся в нем цифры, определите их количество, сумму и найдите максимальное.

4.3. Перестановка элементов

Вы научитесь:

– создавать программы на языке программирования Python (пайтон) с использованием одномерных массивов.

Ключевые слова

Перестановка

Факториал

Дополнительная переменная

Орын ауыстыру

Факториал

Қосымша айнымалы

Retirement

Factorial

Additional variable



Дайте общую трактовку ключевых слов и предположите, как они будут фигурировать в последующем тексте по теме.



Большинство задач на обработку списка сводятся к комбинированию следующих алгоритмов (рис. 4.8).

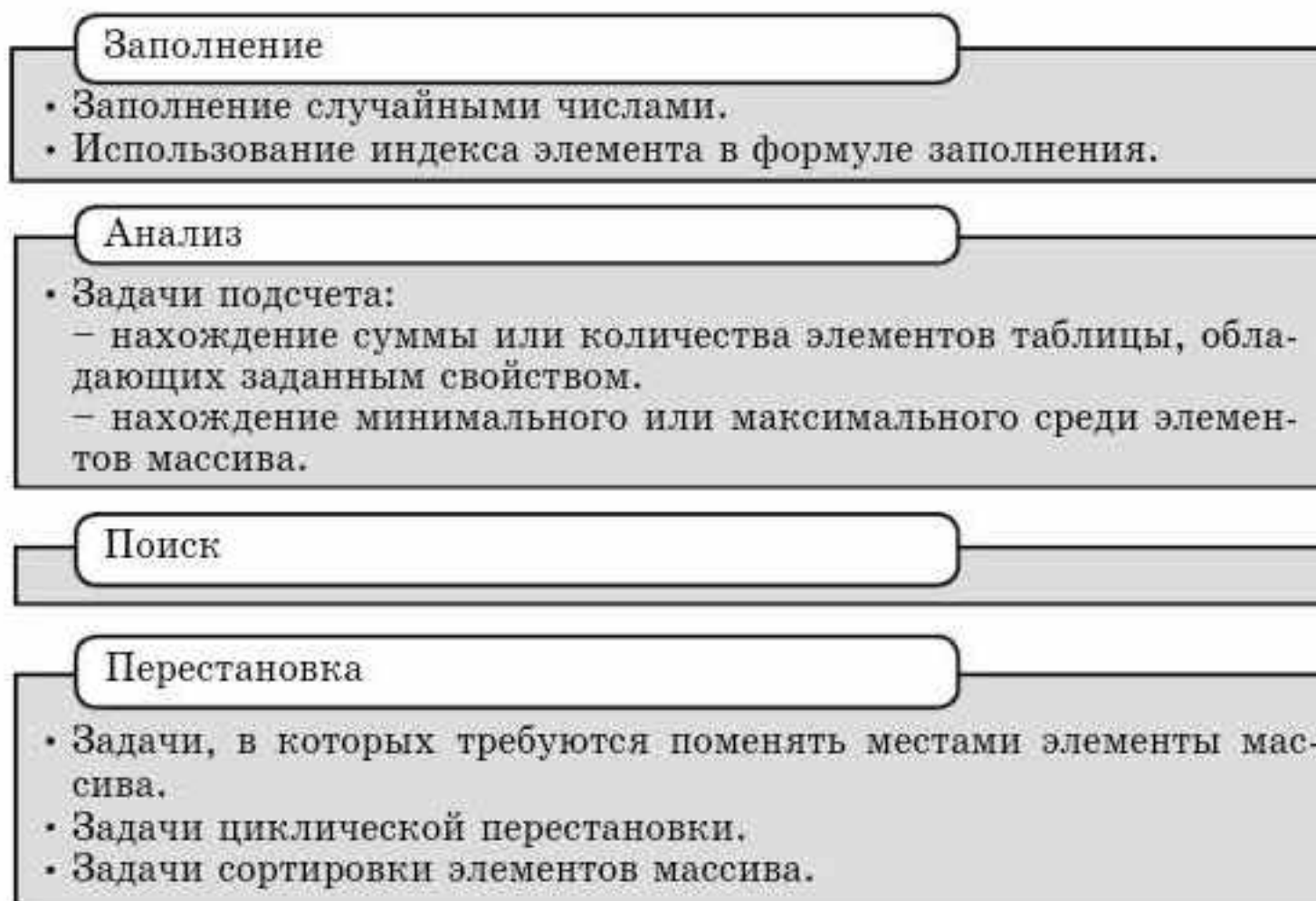


Рис. 4.8. Алгоритмы для обработки списков



Перестановками называются выборки элементов, отличающиеся только порядком расположения элементов, но не самими элементами.

Число перестановок на множестве из n элементов определяется по формуле: $P_n = n \cdot (n-1) \cdot (n-2) \dots 3 \cdot 2 \cdot 1 = n!$

$n!$ (n -факториал) – обозначение для краткой записи произведения всех натуральных чисел от 1 до n включительно. Например, перестановка пяти книг будет равна: $P_5 = 5! = 1 \cdot 2 \cdot 3 \cdot 4 \cdot 5 = 120$.



Приведите примеры из жизни, когда в списках используются перестановки элементов.



Алгоритм перестановки очень часто применяется при решении практических задач. Каким образом происходит перестановка элементов в списке (массиве)?

Наглядный способ для понимания – представить переменную в виде кружки с чаем. Чтобы поменять содержимое кружки (рис. 4.9), нужно использовать третью пустую кружку в качестве посредника или дополнительной переменной.



Рис. 4.9. Перестановка местами

Обозначим переменными $a[1]$ и $a[2]$ две кружки с чаем. Поменяем содержимое кружек между собой посредством третьей кружки, обозначенной как переменная k . Перельем в третью кружку k содержимое второй кружки $a[2]$, тогда получим $k = a[2]$. В освобожденную кружку $a[2]$ перельем содержимое кружки $a[1]$, т. е. получим $a[2] = a[1]$. Переменная $a[1]$ освободится, в нее переливаем содержимое кружки-посредника k , получим: $a[1] = k$. Таким образом, содержимое кружек поменяется, соответственно, элементы меняются местами.

<p>Рассмотрим перестановку генерирования комбинаций элементов последовательности abc. В результате должны получиться следующие списки последовательностей: abc, acb, bac, bca, cab, cba. Используем простой способ перестановки элементов в списке с применением инструмента itertools.permutations из стандартной библиотеки.</p>	<p>1-й способ</p> <pre>import itertools print(list(itertools.permutations(['a', 'b', 'c'])))</pre> <p>Результат</p> <pre>[('a', 'b', 'c'), ('a', 'c', 'b'), ('b', 'a', 'c'), ('b', 'c', 'a'), ('c', 'a', 'b'), ('c', 'b', 'a')]</pre> <p>2-й способ</p> <pre>import itertools for i in itertools.permutations('abc',3): print(''.join(i))</pre> <p>Результат</p> <pre>abc bca acb cab bac cba</pre>
---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

В предыдущей теме вы решали задачи, в которых нужно было найти максимальное и минимальное значения списка. А как можно поменять их местами?



Попробуйте решить задачу, состоящую из двух частей:
 1) поиск минимума и максимума (их индексов) в списке;
 2) перестановка местами минимума и максимума в списке.



Шаг за шагом

Выполните первую часть – поиск минимума и максимума в списке.

1. Присвойте переменной первый индекс списка.
2. Переберите все элементы списка в цикле, начиная со второго. Проверьте условие: если второй элемент списка меньше первого, то переменной нужно присвоить текущий индекс.
3. Поиск максимума выполняется аналогично. Проверяется, если второй элемент списка больше первого, то переменной нужно присвоить текущий индекс.

Выполните вторую часть – перестановка местами минимума и максимума списка.

1. Присвойте буферной (дополнительной) переменной значение минимума.
2. Запишите по индексу минимума максимум списка.
3. Запишите по индексу максимума значение, хранимое в буферной переменной.

Код программы

```
# создание списка из 20 случайных чисел в диапазоне от 0 до 100
from random import random
n = 20
array = [0]*n
for i in range(n):
    array[i] = int(random()*100)
    print(array[i],end=' ')
print()
mini = 0
maxi = 0
# поиск минимального и максимального элементов списка
for i in range(n):
    if array[i] < array[mini]:
        mini = i
    elif array[i] > array[maxi]:
        maxi = i
```



```

print('array[%d]=%d array[%d]=%d' % (mini+1, array[mini], maxi+1,
array[maxi]))
# перестановка местами минимального и максимального элемен-
тов списка
buffer = array[mini]
array[mini] = array[maxi]
array[maxi] = buffer
for i in range(20):
    print(array[i],end=' ')
print()

```

Ниже представлены откомпилированная программа и результат.

```

File Edit Format Run Options Window Help
# создание списка 20 случайных чисел в диапазоне от 0 до 100
from random import random
n = 20
array = [0]*n
for i in range(n):
    array[i] = int(random()*100)
    print(array[i],end=' ')
print()
mini = 0
maxi = 0
# поиск минимального и максимального элементов списка
for i in range(n):
    if array[i] < array[mini]:
        mini = i
    elif array[i] > array[maxi]:
        maxi = i
print('array[%d]=%d array[%d]=%d' % (mini+1, array[mini], maxi+1, array[maxi]))
# перестановка местами минимального и максимального элементов списка
buffer = array[mini]
array[mini] = array[maxi]
array[maxi] = buffer
for i in range(20):
    print(array[i],end=' ')
print()

=== RESTART: C:/Users/Алира/Desktop/Задачи Питон/перестановка max и min.py ===
67 38 4 75 57 21 69 37 31 94 25 58 23 24 77 74 19 62 20 95
array[3]=4 array[20]=95
67 38 95 75 57 21 69 37 31 94 25 58 23 24 77 74 19 62 20 4
>>>

```

Проанализируйте.

Можно ли применить представленную ниже программу для перестановки минимального и максимального элементов в списке? Ответ прокомментируйте. Дайте оценку двум способам решения одной задачи.



```

from random import random
n = 20
array = [0]*n
for i in range(n):
    array[i] = int(random()*100)
    print(array[i],end=' ')
print()
mini = min(array)
maxi = max(array)
imini = array.index(mini)
imaxi = array.index(maxi)
print('array[%d]=%d array[%d]=%d'%(imini+1,mini,imaxi+1,maxi))
array[imini],array[imaxi] = array[imaxi],array[imini]
for i in range(20):
    print(array[i],end=' ')
print()

```



Знание и понимание



1. Как вы понимаете термин «перестановка»?
2. Когда применяются перестановки в списках (массивах)?
3. По какой формуле можно определить количество перестановок? Сколько может быть возможных перестановок n предметов из комбинаторики (область математики)?
4. Почему для перестановки элементов списка используется третья переменная? Приведите пример из жизни.
5. Дан целочисленный список, состоящий из трех элементов – 1, 2, 3. Сколько возможных перестановок можно выполнить для трех элементов? Для четырех?
6. В списке, состоящем из 10 элементов $a=[1, 3, 5, 7, 9, 8, 6, 4, 2, 10]$, нужно поменять местами пятый и девятый элементы. Правильно ли записан фрагмент перестановки элементов списка? Запишите новый список с переставленными элементами.


```

b = a[5]
a[5] = a[9]
a[9] = b

```
7. Сколько и какие сочетания перестановок будут выведены на экран после выполнения программы?


```

import itertools
a=[0,1,2]
print (list(itertools.permutations(a)))

```
8. Составьте и запишите свою версию рассказа, употребив все предложенные ключевые термины по теме.



и максимальный расходы электроэнергии, используя обработку списка. Также узнайте, на сколько процентов отличаются минимальный и максимальный расходы от среднемесячного, и поменяйте их местами.



Задания по тематике окружающего мира.

1) Десять экологов сдали экзамен по охране окружающей среды Восточно-Казахстанской области. Найдите в массиве, сколько экологов будут охранять среду отлично, сколько – хорошо, а сколько – удовлетворительно.

2) Школьники на уроке физической культуры встают в круг и начинают считалку. Каждый раз, после того как выбывает n -й ребенок, круг смыкается. Считалка повторяется, пока не останется один ребенок. Обработайте список, чтобы узнать, кто останется.

3) Показатели дневной и ночной температуры воздуха проверяются ежедневно и записываются в таблицу. Если в течение трех дней подряд показатели среднесуточной температуры составляют менее 8 градусов Цельсия, начинается отопительный сезон. Определите, какого числа в этом году был начат отопительный сезон, если известно, что это произошло в текущем месяце. Используйте обработку массивов.

4) Роща ценных деревьев согдийского ясеня расположена в живописном уголке Шарынского каньона. Роща выжила после ледникового периода 10 тыс. лет назад. Известны координаты каждого дерева. Требуется огородить рощу от браконьеров забором прямоугольной формы, потратив на это как можно меньше денег. Найдите периметр охватывающего прямоугольника, обработав массив (список).

4.4. Двумерный массив

Вы научитесь:

– создавать программы на языке программирования Python с использованием двумерных массивов.

Ключевые слова

Двумерный массив

Екі өлшемді жиым

Two-dimensional array

Матрица

Матрица

Matrix

Строка

Жол

Line

Столбец

Баған

Column



Дайте общую трактовку ключевых слов и предположите, как они будут фигурировать в последующем тексте по теме.



В § 4.1 вы познакомились с одномерными массивами. В зависимости от количества измерений массивы бывают **одномерные**, **двумерные**, **n-мерные**. В языке программирования Python таблицу можно представить в виде **списка строк**.

Это могут быть таблицы результатов экзамена, умножения, расписание самолетов, поездов и т. д.



Данные, хранящиеся в прямоугольной таблице, называются **матрицами**, или **двумерными массивами** (two-dimensional array).

Двумерный массив является массивом ссылок на одномерные массивы. Как вам известно, элементы одномерного массива индексируются одиночным целым числом. А элементы двумерного массива индексируются двумя целыми числами. Первое число – это определение строки, второе – столбца.

После создания массива любой его объект можно использовать в программе везде, где используется имя переменной. Для этого указывают имя массива и целочисленный индексом в квадратных скобках.

Для примера создадим числовую таблицу, состоящую из двух строк и трех столбцов. В одной строке записано `arr = [[11, 22, 33], [44, 55, 66]]`.

Первая строка списка <code>arr[0] = [11, 22, 33]</code>	Вторая строка списка <code>arr[1] = [44, 55, 66]</code>
<code>arr[0][0] = 11</code>	<code>arr[1][0] = 44</code>
<code>arr[0][1] = 22</code>	<code>arr[1][1] = 55</code>
<code>arr[0][2] = 33</code>	<code>arr[1][2] = 66</code>

В двумерных массивах (списках) для **обработки** и **вывода** данных используются два вложенных цикла:

- по номеру строки – первый цикл;
- по элементам внутри строки – второй цикл.

Самый простой вывод матрицы можно осуществить одним оператором `print(a)`. Рассмотрим примеры для вывода двумерного массива на экран.



Необходимо построчно вывести на экран двумерный массив (список). Массив состоит из чисел, разделенных внутри одной строки пробелами.

Проанализируйте, каким образом осуществляется работа первого и второго циклов в кодах программ 1, 2. Чем они различаются?

Шаг за шагом

Код программы 1. Цикл по индексу списка	Код программы 2. Цикл по значениям списка
<pre>arr=[[11, 22, 33], [44, 55, 66]] for i in range(len (arr)): for j in range(len (arr[i])): print("arr[" ,i,j, "]=",arr[i][j], end=' ') print() ===== RESTART: C:/Users arr[0 0]= 11 arr[0 1]= 22 arr[0 2]= 33 arr[1 0]= 44 arr[1 1]= 55 arr[1 2]= 66 >>></pre>	<pre>arr=[[11, 22, 33], [44, 55, 66]] for row in arr: for elem in row: print(elem, end=' ') print() ===== RESTART: C:/Users/Алира 11 22 33 44 55 66 >>></pre>



В представленных кодах программ (1, 2) использованы два вложенных цикла для подсчета суммы всех чисел в матрице. Результаты выполнения двух программ одинаковые.

Проанализируйте программные коды и сделайте вывод, какая из них обрабатывает списки по индексу. Какая программа обрабатывает списки по значениям строк?



Код программы 1

```
arr=[[1,2,3], [4,5,6]]
sum = 0
for i in range (len(arr)):
    for j in range (len(arr[i])):
        sum += arr[i][j]
        print (sum, end=' ')

===== RESTART: C:/Users
1 3 6 10 15 21
>>>
```

Код программы 2

```
arr=[[1, 2, 3], [4, 5, 6]]
sum = 0
for row in arr:
```

```
for elem in row:
    sum += elem
    print(sum, end=' ')
```

```
===== RESTART: C:/Users
1 3 6 10 15 21
>>> |
```



Знание и понимание



1. Что вы понимаете под термином «двумерный массив»?
2. Сколько измерений имеет двумерный массив?
3. В виде чего можно представить таблицу в языке программирования Python?
4. Каким образом индексируется двумерный массив?
5. Приведите примеры двумерных массивов из жизни.
6. Сколько циклов нужно использовать в двумерном массиве при обработке и выводе данных?
7. Пусть дан двумерный массив, состоящий из n строк таблицы. Каждая строка в свою очередь содержит m чисел, разделенных пробелами. Каким образом можно вводить и выводить данные?
8. Составьте и запишите свою версию рассказа, употребив все предложенные ключевые термины по теме.



Применение

Анализ



9. Дан квадратный массив из n строк и n столбцов. Необходимо элементам, находящимся на главной диагонали, присвоить значение 1. Главная диагональ проходит из левого верхнего угла в правый нижний угол. Элементам, которые находятся выше главной диагонали, присвоить значение 0. А элементам, находящимся ниже главной диагонали, – значение 2. Приведем квадратный массив для $n = 4$:

Индексы массива	Элементы массива			
[0],[0] [0],[1] [0],[2] [0],[3]	1	0	0	0
[1],[0] [1],[1] [1],[2] [1],[3]	2	1	0	0
[2],[0] [2],[1] [2],[2] [2],[3]	2	2	1	0
[3],[0] [3],[1] [3],[2] [3],[3]	2	2	2	1

Например, $A[0], [0]=1$; $A[1], [1]=1$; $A[2], [2]=1$; $A[3], [3]=1$.

По условию задачи элементы, которые лежат выше главной диагонали, – это элементы $A[i][j]$, для которых $i < j$. Для элементов ниже

главной диагонали – $i > j$. Таким образом, сравнивая значения i и j , определяем значение $A[i][j]$.

Ниже представлены два алгоритма решения задачи. Какой из способов будет лучшим?

1-й способ

```
for i in range(n):
    for j in range(n):
        if i < j:
            A[i][j] = 0
        elif i > j:
            A[i][j] = 2
        else:
            A[i][j] = 1
```

2-й способ

```
for i in range(n):
    A[i][i] = 1
for i in range(n):
    for j in range(i + 1, n):
        A[i][j] = 0
for i in range(n):
    for j in range(0, i):
        A[i][j] = 2
```

10. Дано число n . Создайте массив размером $n \times n$ и заполните его по следующему правилу:

- числа на диагонали, идущей из правого верхнего в левый нижний угол, равны 1.
- числа, стоящие выше этой диагонали, равны 0.
- числа, стоящие ниже этой диагонали, равны 2.

Полученный массив выведите на экран. Числа в строке разделяйте одним пробелом.

Примеры

Входные данные

4

Выходные данные

```
0 0 0 1
0 0 1 2
0 1 2 2
1 2 2 2
```


11. Пусть задан массив, в котором $n = 5$ строк, $m = 6$ столбцов. Элементы в массиве разделены пробелами:

```
0 0 0 0 0 0
0 1 2 3 4 5
0 2 4 6 8 10
0 3 6 9 12 15
0 4 8 12 16 20
```

Создайте программу, вычисляющую элементы в строке i и столбце j по формуле: $A[i][j]=i*j$. Для создания массива используйте генератор: `[[i*j for j in range(m)] for i in range(n)]`.



Синтез

Оценивание



12. С трех метеостанций, расположенных в разных регионах Казахстана, получены значения температуры воздуха за 4 дня.

Номер станции	1-й день	2-й день	3-й день	4-й день
1	-10	-12	-18	-15
2	15	18	20	17
3	5	7	11	13

В двумерном массиве запись показаний выглядит следующим образом:

$t[1],[1]=-10$	$t[1],[2]=-12$	$t[1],[3]=-18$	$t[1],[4]=-15$
$t[2],[1]=15$	$t[2],[2]=18$	$t[2],[3]=20$	$t[2],[4]=17$
$t[3],[1]=5$	$t[3],[2]=7$	$t[3],[3]=11$	$t[3],[4]=13$

- 1) Напишите программу, которая должна распечатать показатели температуры на 2-й метеостанции за 4-й день и на 3-й метеостанции за 1-й день.
- 2) Распечатайте показания термометров всех метеостанций за 2-й день.
- 3) Определите среднюю температуру на 3-й метеостанции.
- 4) Распечатайте, в какие дни и на каких метеостанциях температура была в диапазоне 15–20 градусов тепла.

13. В кинотеатре n рядов по m мест в каждом. В двумерном массиве хранится информация о проданных билетах. Число 1 означает, что билет на данное место уже продан, число 0 – что место свободно. Поступил запрос на продажу k билетов на соседние места в одном ряду. Определите, можно ли выполнить такой запрос.

Входные данные

Данные вводятся с клавиатуры и выводятся на экран. Программа получает на вход числа n и m . Далее идет n строк, которые содержат m чисел (0 или 1), разделенных пробелами. Затем дано число k .

Выходные данные

Программа должна вывести номер ряда, в котором есть k подряд идущих свободных мест. Если таких рядов несколько, то выведите номер наименьшего подходящего ряда. Если подходящего ряда нет, выведите число 0.

Примеры

Входные данные

```
3 4
0 1 0 1
1 0 0 1
1 1 1 1
```

2

Выходные данные

2

14. Дано нечетное число n . Создайте двумерный массив из $n \times n$ элементов, заполнив его символами "." (каждый элемент массива является строкой из одного символа). Затем заполните символами "*" среднюю строку массива, средний столбец массива, главную и побочную диагонали. В результате единицы в массиве должны образовывать изображение звездочки. Выведите полученный массив на экран, разделяя элементы массива пробелами.

Примеры

Входные данные

5

Выходные данные

```
* . * . *
. * * * .
* * * * *
. * * * .
* . * . *
```

15. Даны два числа n и m . Создайте массив $n \times m$ и заполните его по следующим правилам:

– числа, стоящие в строке 0 или в столбце 0, равны 1 ($A[0][j]=1$, $A[i][0]=1$);

– для всех остальных элементов массива $A[i][j]=A[i-1][j]+A[i][j-1]$, т. е. каждый элемент равен сумме двух элементов, стоящих слева и сверху от него.

Входные данные

Вводятся два числа n и m .

Выходные данные

Выведите данный массив на экран, отводя на вывод каждого элемента массива ровно 6 символов (см. пример).

Примеры

Входные данные

4 6

Выходные данные

1	1	1	1	1	1
1	2	3	4	5	6
1	3	6	10	15	21
1	4	10	20	35	56



Некоторые правила на языке программирования Python, которых следует обязательно придерживаться.

- 1) Пробелам придают особое значение для удобного чтения кода программы.
- 2) Пробелы имеют синтаксическое значение.
- 3) Используются пробелы для создания отступов, а не символы табуляции.
- 4) Для каждого уровня синтаксически значимых отступов, используйте по четыре пробела.
- 5) Длина строк не должна превышать 79 символов.
- 6) Дополнительные строки, которые являются продолжением длинных выражений, должны выделяться четырьмя дополнительными пробелами.
- 7) Индексы элементов списков не должны быть окружены пробелами.
- 8) До и после оператора присваивания помещайте только по одному пробелу.

4.5. Сортировка

Вы научитесь:

- применять алгоритмы сортировки;
- создавать программы на языке программирования Python с использованием одномерных массивов;
- создавать программы на языке программирования Python с использованием двумерных массивов.

Ключевые слова

Сортировка пузырьком	Көпіршік әдісі бойынша сұрыптау	Bubble sort
Сортировка по возрастанию	Өсуі бойынша сұрыптау	Sort Ascending
Сортировка по убыванию	Кемуі бойынша сұрыптау	Sorting in decreasing order



Дайте общую трактовку ключевых слов и предположите, как они будут фигурировать в последующем тексте по теме.



Сортировка элементов в массиве является неотъемлемой частью анализа данных.



Сортировка – упорядочение элементов в подмножестве данных по какому-либо критерию.



Основная цель сортировки – облегчить поиск элементов в отсортированном массиве при обработке данных. Сортировка бывает двух видов:

1) **сортировка по убыванию**, при котором следующий элемент не больше предыдущего: 7, 6, 5, 4, 3, 2, 1;

2) **сортировка по возрастанию**, при которой каждый последующий элемент не меньше предыдущего: 1, 2, 3, 4, 5, 6, 7.

Существует большое количество различных алгоритмов сортировки, отличающиеся друг от друга временем выполнения, сложностью в понимании и реализации, используемой памятью (рис. 4.10).

Сортировка

- Метод «пузырька»
- Метод простого выбора
- Метод простого включения
- Метод Шелла
- Пирамидальная
- Быстрая сортировка Хоара

Рис. 4.10. Виды сортировок

Разберем один из самых простых алгоритмов сортировки – сортировка методом «пузырька». Этот метод очень прост в понимании и реализации. Подходит для сортировки небольших списков (массивов).

Алгоритм сортировки заключается в следующем:

1. Для сортировки по возрастанию сравниваем $[i]$ элемент списка с $[i+1]$. Если больший элемент имеет наименьший порядковый номер, то они меняются местами.

2. При упорядочивании самый большой элемент сдвигается в конец списка.

3. Процесс повторяется до последнего отсортированного элемента списка.



Дан список $[1, 6, 9, 7, 8, 3]$. Расположите элементы списка в порядке убывания. То есть, если элемент слева меньше элемента-соседа справа, то меняется с ним местами. В итоге должен получиться список $[9, 8, 7, 6, 3, 1]$.



Шаг за шагом

В табл. 4.6 приведен последовательный разбор сортировки методом «пузырька».

Таблица 4.6

Первая итерация							
Сравниваем 1 и 2	6	1	9	7	8	3	Выполним сортировку по убыванию списка $[1, 6, 9, 7, 8, 3]$ вручную. После прохождения первой итерации (прохода) минимальное число 1 переходит на конец списка. Поэтому после первой итерации с числом 1 сравнивать нет смысла.
Сравниваем 2 и 3	6	9	1	7	8	3	
Сравниваем 3 и 4	6	9	7	1	8	3	
Сравниваем 4 и 5	6	9	7	8	1	3	
Сравниваем 5 и 6	6	9	7	8	3	1	
Вторая итерация							
Сравниваем 1 и 2	9	6	7	8	3	1	После прохождения второй итерации в конце списка будут находиться два числа из списка по убыванию: 3 и 1. Что вы заметили? Количество сравнений уменьшилось на единицу.
Сравниваем 2 и 3	9	7	6	8	3	1	
Сравниваем 3 и 4	9	7	8	6	3	1	
Сравниваем 4 и 5	9	7	8	6	3	1	
Третья итерация							
Сравниваем 1 и 2	9	7	8	6	3	1	Продолжаем сравнивать элементы в списке. После третьей итерации в конце списка будут находиться числа 6, 3, 1.
Сравниваем 2 и 3	9	8	7	6	3	1	
Сравниваем 3 и 4	9	8	7	6	3	1	

Четвертая итерация						
Сравниваем 1 и 2	9	8	7	6	3	1
Сравниваем 2 и 3	9	8	7	6	3	1
Пятая итерация						
Сравниваем 1 и 2	9	8	7	6	3	1
[1, 6, 9, 7, 8, 3] → [9, 8, 7, 6, 3, 1]						

После прохождения пятой итерации получим отсортированный список по убыванию.

Если список состоит из шести элементов, то выполняется пять итераций для последовательного сравнения соседних элементов.

Программа сортировки методом «пузырька»

```
# Список из шести чисел
lst = [6, 9, 7, 8, 3, 1]
# Определение длины списка
n = len(lst)
# Внешний цикл для отсчета количества итераций по списку
for j in range(0, n-1):
    # Вложенный цикл для сравнения i-го с i+1-м элементом
    # Перестановка элементов местами при необходимости
    # Количество сравнений уменьшается на величину j с каждой
    итерацией
    for i in range(0, n-j-1):
        if lst[i] < lst[i+1]:
            lst[i], lst[i + 1] = lst[i + 1], lst[i]
    print(j+1, "-ая итерация цикла - ", end=" ")
    print(lst)
print(lst)
```

Результат

```
=== RESTART: C:/Users/Алира/Desktop/Задачи Питон
1 -ая итерация цикла - [6, 9, 7, 8, 3, 1]
2 -ая итерация цикла - [9, 7, 8, 6, 3, 1]
3 -ая итерация цикла - [9, 8, 7, 6, 3, 1]
4 -ая итерация цикла - [9, 8, 7, 6, 3, 1]
5 -ая итерация цикла - [9, 8, 7, 6, 3, 1]
[9, 8, 7, 6, 3, 1]
>>>
```

В языке программирования Python легко выполнять сортировку, применяя один из двух способов вызова сортировки.

Первый способ – это метод `sort([key])`, который сортирует элементы по возрастанию.

Код программы

```
arr = [13, -2, 55, 17, -15, 0, 8, 3, 10, 11]
arr.sort()
print (arr)
```

Результат

```
===== RESTART: C:/Users/Алира/Desktop/Э
[-15, -2, 0, 3, 8, 10, 11, 13, 17, 55]
>>>
```

Приведем еще пример, в котором элементы списка являются вложенными списками (матрица). В такой матрице сортировка происходит по первым элементам или по первому столбцу.

```
arr = [[35, 100], [18, 8], [31, 11]]
arr.sort()
print (arr)
```

Результат
[[18, 8], [31, 11], [35, 100]]

Метод `sort` не возвращает значения, поэтому результат данного метода нельзя использовать в арифметических выражениях или при выводе результата.

Второй способ – это функция `sorted`, наоборот, не модифицирует переданный ей список, а возвращает новый список. Данную функцию можно использовать следующим образом:

```
import random
a = [random.randint(1, 10) for i in range(10)]
b = sorted(a)
print(b)
```

```
===== RESTART: C:/Users/Ал
[3, 3, 5, 6, 6, 7, 8, 8, 8, 10]
```



В чем преимущество сортировки списков с помощью функции `sort()`? Исследуйте функцию `sorted`. Существует ли возможность провести в одной строке сортировку списка, состоящего из чисел, и вывести его результат на экран?



Дана матрица 3×3 , элементы в ней созданы случайным образом. Необходимо матрицу отсортировать по возрастанию элементов первой строки. В приведенной ниже программе, запишите комментарии к каждому оператору. Что получится в результате работы программы? Как отсортирована первая строка? Как перемещены элементы столбцов, упорядоченные по возрастанию?



Шаг за шагом

В матрице меняется последовательность столбцов, т. е. элементы первой строки будут отсортированы по возрастанию.

Код программы

```
from random import random
n1 = 3
n2 = 3
a = []
for i in range(n1):
    b = []
    for j in range(n2):
        n = int(random() * 100)
        b.append(n)
        print("%4d" % n, end="")
    print()
    a.append(b)
print()
z = n2-1
while z > 0:
    index = 0
    for j in range(z+1):
        if a[0][j] > a[0][index]:
            index = j
    for i in range(n1):
        buf = a[i][index]
        a[i][index] = a[i][z]
        a[i][z] = buf
    z -= 1
for i in range(n1):
    for j in range(n2):
        print("%4d" % a[i][j], end="")
    print()
```

Результат

```
===== RESTART:
59 64 55
20 83 72
 4  5 64

55 59 64
72 20 83
64  4  5
```

Для того чтобы быстро расставить все элементы в списке в обратном порядке, применяют функцию `reverse()`. Приведем пример.

```
data = [1, 2, 3, 4, 5, 6, 7, 8, 9]
data.reverse()
print(data)
```

```
===== RESTART: C:/Users/
[9, 8, 7, 6, 5, 4, 3, 2, 1]
>>>
```




Знание и понимание



1. Что такое *сортировка*?
2. Какая сортировка называется сортировкой по убыванию? Возрастающую?
3. Какова основная цель сортировки?
4. Какие виды сортировок существуют?
5. Какой метод сортировки самый распространенный?
6. Как происходит перебор элементов массива методом «пузырька»?
7. С какой целью применяются функции для сортировки списков `sort()` и `sorted`?
8. Дан код программы. Что будет выведено на экран в результате ее выполнения?

```
kol_a = 2
kol_b = 3
elem_arr = 0
arr = []
for i in range(kol_a):
    arr.append([])
    for j in range(kol_b):
        arr[i].append(elem_arr)
        elem_arr += 1
print(arr)
```

9. Дан код программы:

```
list_arr = ['список', 'одномерный массив', 'двумерный массив',
            'строка', 'столбец', 'цикл']
for i in list_arr:
    print(i)
```

- Какой результат будет выведен на экран? Какой размерности список?
10. Составьте и запишите свою версию рассказа, употребив все предложенные ключевые термины по теме.



Применение

Анализ



11. В приведенной программе дана матрица 5×6 . Выполните по шагам каждый оператор, записав комментарии к ним. Каково условие задачи, что должно получиться в результате выполнения программы? Сколько раз в программе применяется генератор? Выполните программу в среде программирования Python.

```
from random import random
N = 5
M = 6
lst = []
```

```

arr = [0] * N*M
for i in range(N):
    arr1 = []
    for j in range(M):
        arr1.append(int(random()*100))
    lst.append(arr1)
for i in range(N):
    for j in range(M):
        print("% 3d" % lst[i][j], end='')
    print()
print()
k = -1
for j in range(M):
    for i in range(N):
        k += 1
        arr[k] = lst[i][j]
        print("% 3d" % arr[k], end='')
print()

```

12. В одномерном массиве, созданном случайным образом из n целых чисел, отсортируйте элементы массива методом выбора по возрастанию. Сортировка выбором заключается в поиске максимального элемента массива в каждой итерации. Максимальный элемент перемещается на последнее место в просматриваемой части массива.

Пример, входного и выходного массива, созданных случайным образом:

Входной массив

[47, 12, 51, 54, 16, 24, 74, 86, 99, 30]

Выходной массив

[12, 16, 24, 30, 47, 51, 54, 74, 86, 99]

13. Создайте программу, состоящую из списка целых чисел. Проведите сортировку по возрастанию суммы цифр чисел.

14. Дан список целых чисел, состоящий из 30 элементов. Найдите сумму пяти самых больших и пяти самых маленьких элементов списка.



Синтез

Оценивание



15. Дан двумерный массив из целых чисел, состоящий из пяти строк и четырех столбцов. Найдите сумму пяти самых больших и пяти самых маленьких элементов списка.

16. Создайте список, сформированный из целых чисел. Замените все нечетные числа списка нулями и выведите их количество на печать.



Интересные задачи

1. «Любимые вещи». Создайте список своих любимых развлечений и сохраните его в переменной `games`. Затем создайте список любимых лакомств, сохранив его в переменной `foods`. Объедините два этих списка, сохранив результат в переменной `favorites`, и напечатайте значение этой переменной.

2. «Подсчет воинов». Есть 3 дома, на крыше каждого из которых прячутся по 25 ниндзя, и есть 2 туннеля, в каждом из которых скрывается по 40 самураев. Сколько всего воинов?

3. «Вариация массива». Кайрат назвал «вариацией массива» операции, выполненные по следующему алгоритму:

- 1) массив сортируется по возрастанию элементов;
- 2) после сортировки находится разность второго и первого элементов, третьего и второго, четвертого и третьего и т. д., до разности последнего и предпоследнего элементов;
- 3) затем все найденные разности суммируются.

Помогите Кайрату, написать программу, которая проделывает указанную операцию над заданным массивом.

Формат входа. В первой строке задано целое число n – количество элементов в массиве ($2 \leq n \leq 1000$). В следующей строке через пробел заданы целые числа, каждое из которых по модулю не превосходит 10^6 .

Формат выхода. Выведите единственное целое число – вариацию заданного набора чисел.

Пример

Вход	Выход
4	5
5 2 6 1	

Примечание. Результат получается следующий: отсортированный массив: 1 2 5 6; вычисляем разности: $2 - 1 = 1$, $5 - 2 = 3$, $6 - 5 = 1$; вычисляем сумму разностей $1 + 3 + 1 = 5$.

4.6. Удаление и вставка элемента

Вы научитесь:

- создавать программы на языке программирования Python с использованием одномерных массивов;
- создавать программы на языке программирования Python с использованием двумерных массивов.

Ключевые слова

Удаление элемента

Элементті жою

Delete an item

Вставка элемента

Элементті кірістіру

Inserting an element



Дайте общую трактовку ключевых слов и предположите, как они будут фигурировать в последующем тексте по теме.



Часто из массива необходимо удалять ненужные элементы. Каким образом происходит удаление элемента из массива?



Удаление элемента из массива означает, что на место удаленного элемента записывается тот элемент массива, который должен остаться. Поэтому элементы массива сдвигаются к какому-нибудь краю массива. Обычно сдвиг делается в сторону начала массива, чтобы индексация снова начиналась с нуля. Рассмотрим удаление на конкретном примере.



Как удалить элементы списка по условию? Например, из списка баллов по экзаменам необходимо удалить элементы, значения которых больше 40 и меньше 60 баллов. При этом удаляемые числа необходимо сохранить в другом списке.



Шаг за шагом

Для удаления элемента списка воспользуемся инструкцией **del**, указав сам список и индекс удаляемого элемента. Алгоритм решения задачи заключается в переборке элементов списка.

- Если элемент списка удовлетворяет условию, то удаляем элемент.
- При удалении элемента на его место становится следующий элемент.

– Переходим к следующему элементу, при этом пропускаем проверку элемента, который занял место удаленного.

Цикл **for** при удалении из списка элементов нельзя использовать, поскольку меняется количество элементов списка.

```
import random
arr = []
for i in range(10):
    n = round(random.random() * 100)
    arr.append(n)
print("Исходный список", arr)
lst = []
i = 0
while i < len(arr):
    if 20 < arr[i] < 40:
        lst.append(arr[i])
        del arr[i]
    else:
        i += 1
print("Список после удаления по условию", arr)
print("Список удаленных", lst)
```

Результат

Исходный список [10, 24, 46, 76, 32, 79, 19, 90, 12, 94]

Список после удаления по условию [10, 46, 76, 79, 19, 90, 12, 94]

Список удаленных [24, 32]

В программе использован цикл **while**, который позволяет измерять длину списка на каждой итерации.



Не менее часто применяется на практике **вставка** элементов в массив. Как можно вставлять элементы в массив?



При добавлении элементов в массив их количество будет возрастать. Данные, удовлетворяющие заданному условию, можно добавлять как перед, так и после элемента массива. Это зависит от условия решаемой задачи.

Для вставки элементов в списки применяется метод **insert(index, x)**. Данный метод добавляет элемент **x** в список по индексу **index**.



В список `colors`, состоящий из трех элементов строкового типа данных, во вторую позицию нужно вставить новый элемент.



Шаг за шагом

Код программы

```
colors = ['blue', 'orange', 'red']
colors.insert(2, 'yellow') # вставляет элемент в указанную позицию
print(colors)
```

Результат

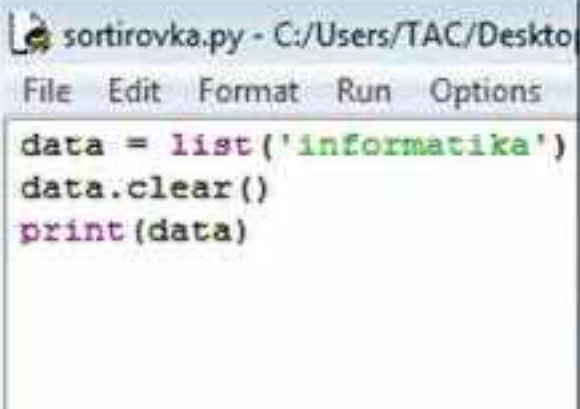

```
----- RESTART: C:/Users/TAC/Desktop
['blue', 'orange', 'yellow', 'red']
>>>
```

Кроме выше рассмотренных методов работы со списками, существует ряд других, которые приведены в табл. 4.7.

Таблица 4.7

Методы в списках

Методы	Назначение	Примеры
<code>remove(x)</code>	Удаляет только первое вхождение элемента <code>x</code> . Если элемент не найден, генерирует исключение <code>ValueError</code> .	<pre>b=[2, 3, 5] print(b) b.remove(3) print(b)</pre> <p>[2, 5] – будет удалено первое вхождение на данный элемент.</p>
<code>count(x)</code>	Вычисляет количество элементов равных <code>x</code>	<pre>a = [17.05, 777, 5, 777, 777, 111.25] print(a.count(777), a.count(111.25), a.count('f')) # считает количество повторений</pre> <p>Код программы</p> <div style="border: 1px solid black; padding: 5px; display: inline-block; margin-top: 10px;">Результат</div>

Методы	Назначение	Примеры
<code>clear()</code>	Удаляет элементы списка	<pre>data = list('informatika') data.clear() print(data)</pre> <div style="display: flex; justify-content: space-around;"> <div style="text-align: center;"> <p>Код программы</p>  </div> <div style="text-align: center;"> <p>Результат</p>  </div> </div>



Знание и понимание

1. Каким образом можно вставлять и удалять элементы массива?
2. Почему использование методов вставки и удаления элементов в списке облегчает работу программиста?
3. Разгадайте кроссворд по разделу IV «Программирование алгоритмов на языке программирования Python», записав слова по горизонтали и вертикали.

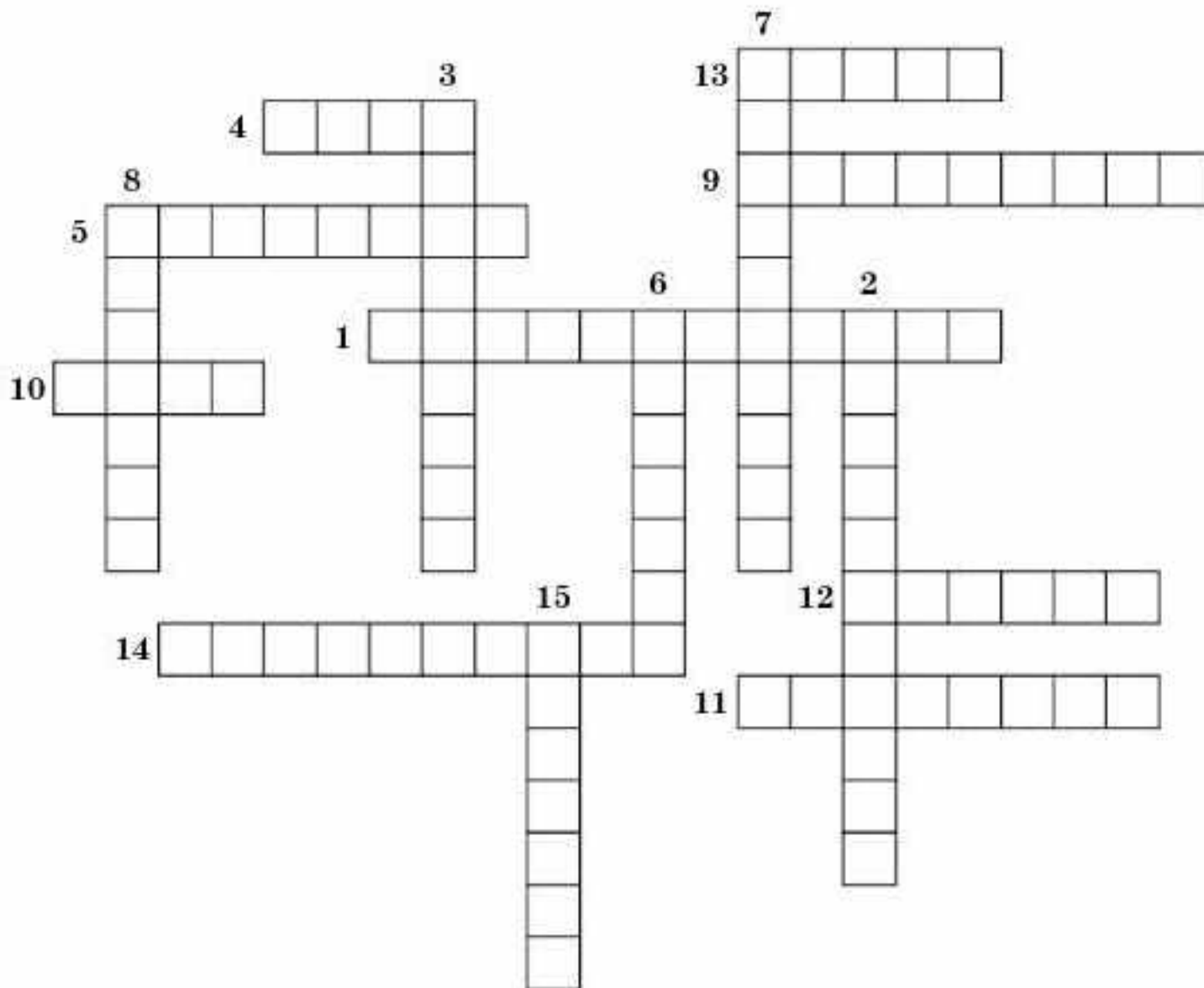
По горизонтали:

1. Чтобы поменять местами элементы массива, используется метод
4. ... данных с клавиатуры.
5. Наибольшее число в последовательности – это
9. Первая ..., которая выводит на экран строку текста, состоит из одной строки `print("Hello, world!")`.
10. ... – это повторение одного и того же участка кода в программе.
11. Какой метод используется, чтобы убрать ненужные элементы ... ?
12. Структура в Python, где элементы находятся в упорядоченной последовательности.
13. Как называется функция для печати на экран?
14. Упорядочение элементов в подмножестве данных по какому-либо критерию называется

По вертикали:

2. Отсортированный список [1, 2, 3, 5, 6, 7, 8] – это
3. Данные, хранящиеся в прямоугольной таблице, называются матрицами, или ... массивами.
6. ... представляется в языке программирования Python в виде списка строк.
7. Команда Run (F5) –

8. Наименьшее число в последовательности – это
 15. ... применяется как перед, так и после элемента массива.



4. Составьте и запишите свою версию рассказа, употребив все предложенные ключевые термины по теме.



Применение

Анализ

5. Что получится в ходе выполнения программы с применением метода `insert()`?

```
1)
a = [1,2,3,4]
a.insert(1,'a')
print(a)
```

```
2)
a = [1,2,3,4]
a.insert(1,[10,12])
print(a)
```

6. В список нужно ввести целые числа, при этом четные числа должны добавляться в начало списка, а нечетные – в конец списка.

1) Для написания программы используйте методы списка: `append()` и `insert()`.

- 2) Метод `append()` применяйте для того, чтобы добавить переданный в качестве аргумента элемент в конец списка.
- 3) Метод `insert()` вставляет элемент по указанному индексу. Элемент будет добавлен в начало, если указать индекс 0.
- 4) Для проверки числа на четность его надо разделить на 2 и найти остаток от деления. Если полученное число равно 0, то число четное, иначе – нечетное.
- 5) Программу проверьте в среде программирования Python.

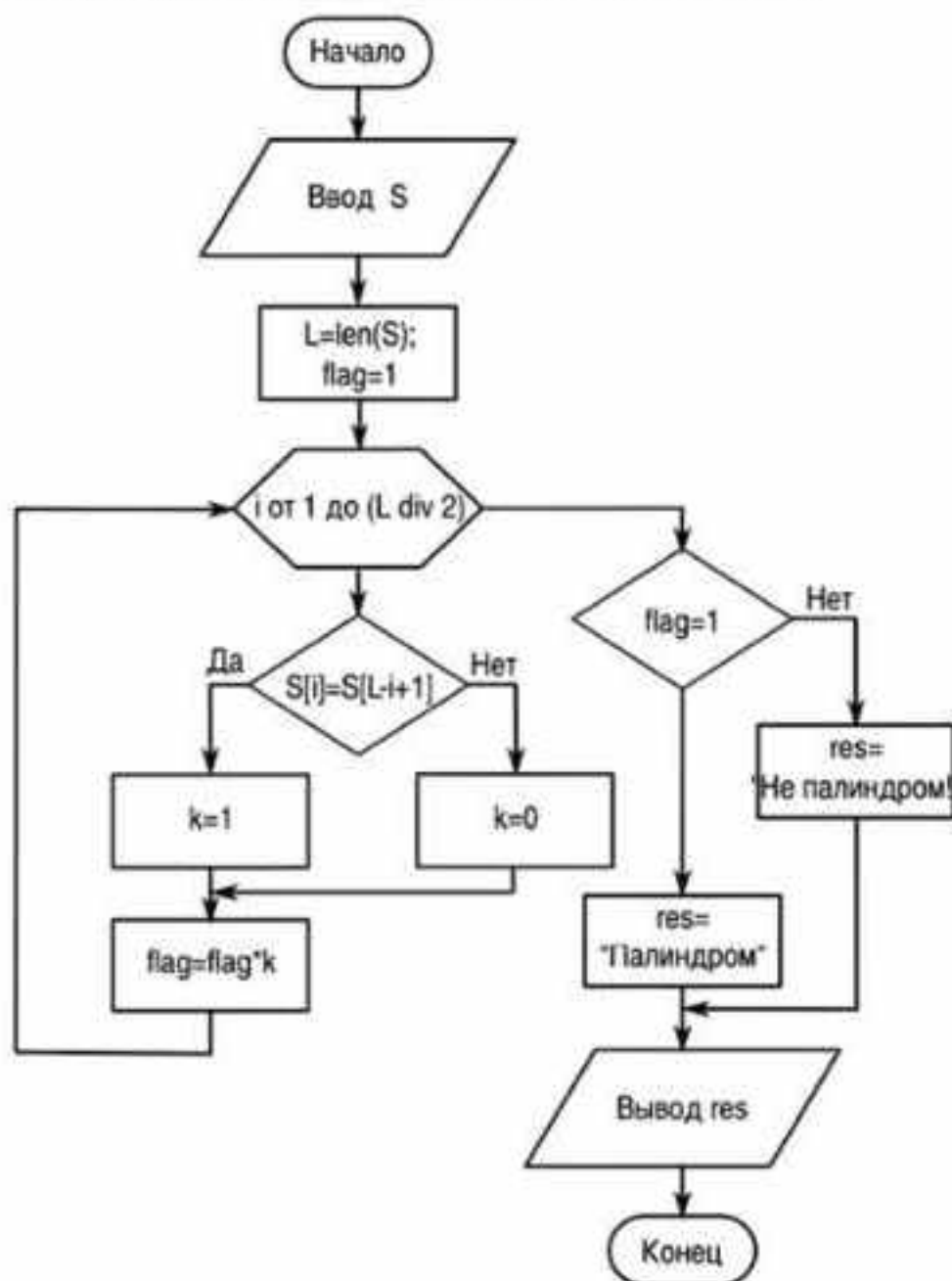


Синтез

Оценивание



7. На приведенном ниже рисунке дана блок-схема. Напишите программу, которая определяет, является ли введенная строка палиндромом («перевертышем»). Палиндромы – это слова, словосочетания и предложения, которые одинаково читаются слева направо и справа налево. Например, *shalash*, *oko*, *ata*, *mat* и т. д.



Блок-схема поиска палиндрома

8. Дан список целых чисел, число k и значение C . Необходимо вставить в список на позицию с индексом k элемент, равный C , сдвинув все элементы, имевшие индекс не менее k , вправо.

Так как количество элементов в списке увеличивается, после считывания списка в его конец нужно будет добавить новый элемент, используя метод `append`.

Вставку необходимо осуществлять уже в считанном списке, не делая этого при выводе и не создавая дополнительного списка.



Дополнительные типы данных в Python

Кроме изученных типов данных, в языке программирования Python существуют дополнительные: множества, кортежи, словари.

Множество (set) – неупорядоченная коллекция неизменяемых уникальных элементов.

Создание множества:

```
mn = {'A', 'C', 4, '5', 'B'}
print(mn)
```

Результат:

```
{4, 'B', '5', 'A', 'C'}
```

Полученное множество отображается в другом порядке, а не таким, как его создавали.

Функция `range` позволяет создавать множества из диапазона:

```
set(range(10))
```

Результат:

```
{0, 1, 2, 3, 4, 5, 6, 7, 8, 9}
```

Словари

Тип данных **словарь (dict)** – неупорядоченная изменяемая коллекция или список с произвольными ключами неизменяемого типа.

Для примера создадим словарь, в котором каждому слову на английском языке будет соответствовать слово на казахском языке:

```
enkz = dict() # пустой словарь
enkz['one'] = 'bir' # добавляем 'bir' для элемента с индексом 'one'
print(enkz)
```

Результат:

```
{'one': 'bir'}
```

```
enkz['two'] = 'eki' # добавляем 'eki' для элемента с индексом 'two'
print(enkz)
```

Результат:

```
{'one': 'bir', 'two': 'eki'}
```

```
enkz['three'] = 'ush' # добавляем 'ush' для элемента с индексом 'three'
print(enkz)
```

Результат:

```
{'one': 'bir', 'two': 'eki', 'three': 'ush'}
```

Срезы

Со списками, так же, как и со строками, можно выполнять срезы:

- $A[i:j]$ срез из $j-i$ элементов $A[i], A[i+1], \dots, A[j-1]$.
- $A[i:j:-1]$ срез из $i-j$ элементов $A[i], A[i-1], \dots, A[j+1]$ (т. е. меняется порядок элементов).
- $A[i:j:k]$ срез с шагом k : $A[i], A[i+k], A[i+2*k], \dots$. Если значение $k < 0$, то элементы идут в противоположном порядке.
- Каждое из чисел i или j может отсутствовать, что означает "начало строки" или "конец строки"

Списки, в отличие от строк, являются изменяемыми объектами: можно отдельному элементу списка присвоить новое значение. Также срезы можно менять целиком.

Итоговые тесты и задания по разделу «Программирование алгоритмов на языке программирования Python (Пайтон)»

1. Определите, что будет напечатано в результате выполнения следующего кода программы:

```
s = 0
for k in range(3,11):
    s = s+k
print(s)
```

Ответ:

2. Определите, что будет напечатано в результате выполнения следующего кода программы:

```
s = 0
for k in range(-5,11):
    s = s+2*k
print(s)
```

Ответ:

3. Определите, что будет напечатано в результате выполнения следующего кода программы:

```
s = 1
for k in range(1,30):
    s = (k-5)*s
print(s)
```

Ответ:

4. В результате выполнения программы, записанной ниже, на экран будет выведено два числа А и В. Укажите такое наибольшее число х, при вводе которого на экран будет выведено сначала 3, а потом 5.

```
x = int(input())
A = 0
B = 0
while x > 0:
    A = A+1
    if B < x%10:
        B = x%10
        x = x//10
print(A)
print(B)
```

Ответ:

5. Какое число нужно написать вместо многоточия, чтобы цикл выполнялся ровно 2 раза?

```
i = 3
while i >= ...:
    print ( "Привет!")
    i -= 1
```

Ответ:

6. Задан массив X из N элементов. Какое условие надо поставить вместо многоточия, чтобы найти количество четных элементов массива в переменной S?

```
S = 0
for j in range(N):
    if ...:
        S += 1
```

- a) $S // 2 == 1$;
- b) $S \% 2 == 0$;
- c) $X[j] \% 2 == 1$;
- d) $X[j] \% 2 == 0$;
- e) $X[j] // 2 == 0$.

7. Требуется заполнить массив именно так:

```
X = [1 3 5 7 9 11]
```

Какой оператор надо поместить в тело цикла вместо многоточия?

```
X = [0] * 6
for k in range(6): ...
```

- a) $X[k] = k$;
- b) $X[k] = 2 * k$;
- c) $X[k] = 2 * k - 1$;
- d) $X[k] = 2 * k + 1$;
- e) $X[k] = 2 * (k + 1)$.

8. Требуется заполнить массив именно так:

```
X = [12 9 6 3 0 - 3]
```

Какой оператор надо поместить в тело цикла вместо многоточия?

```
X = [0] * 6
for k in range(6): ...
```

- a) $X[k] = k$;
- b) $X[k] = 12 - 2 * k$;
- c) $X[k] = 3 * k - 12$;
- d) $X[k] = 3 * (k + 1) + 9$;
- e) $X[k] = 12 - 3 * k$.

9. Требуется заполнить массив именно так:

$X = [0\ 3\ 4\ 7\ 8\ 11]$

Какой оператор надо поместить в тело цикла вместо многоточия?

$X = [0]*6$

for k in range(6): ...

- a) $X[k] = 3 * k - k \% 2;$
- b) $X[k] = 2 * k + k \% 2;$
- c) $X[k] = 2 * k - k \% 2;$
- d) $X[k] = 2 * k + k / 2;$
- e) $X[k] = 2 * (k - 1).$

10. Требуется заполнить массив именно так:

$X = [1\ 2\ 4\ 8\ 16\ 32]$

Какой оператор надо поместить в тело цикла вместо многоточия?

$X = [1]*6$

for k in range(1,6): ...

- a) $X[k] = k;$
- b) $X[k] = 2 * k;$
- c) $X[k] = X[k - 1] + 1;$
- d) $X[k] = 2 * X[k - 1];$
- e) $X[k] = 2 * (X[k - 1] - 1).$

Проекты к разделу IV

1. Проект «Вероятность повторения заданного элемента в списке, созданном случайным числом» (поиск элемента с заданными свойствами).

2. Проект «Сбор повторяющихся элементов массива в нескольких массивах в первый массив» (поиск и перестановка элементов с заданным свойством).

Практические задания

1) Напишите программу, которая получает на входе произвольный список и удаляет из него все повторяющиеся элементы.

2) Напишите программу, которая запрашивает с клавиатуры 10 чисел и добавляет их в список. На экран выводит их сумму, максимальное и минимальное значения. Воспользуйтесь встроенными функциями `sum()`, `max()`, `min()`.

3) Напишите программу, которая генерирует 100 случайных вещественных чисел и заполняет ими список. Выведите отсортированный список на экран. Сортировку проведите с помощью метода `sort()`.

4) Напишите программу, которая в цикле вводит десять чисел по отдельности. Затем добавляет их в список, который затем сортирует и выводит, умножив каждое число на 10.

5) Напишите программу, которая в списке чисел проверяет уникальность каждого элемента (т. е. каждое число встречается только один раз).

6) Дана матрица. Найдите сумму элементов столбцов матриц. Отсортируйте столбцы матрицы по возрастанию их сумм.

7) Даны два списка. Найдите в них одинаковые элементы, присутствующие в обоих списках. Выведите их на печать.

8) Напишите программу, которая вычисляет среднее арифметическое отрицательных элементов списка.

9) Дана матрица:

-1	2	1	-4	-2
-2	1	-3	-2	-6
-3	-1	1	5	2
-4	-2	3	-1	-4

Запишите в последний элемент строки матрицы сумму предыдущих элементов из той же строки.

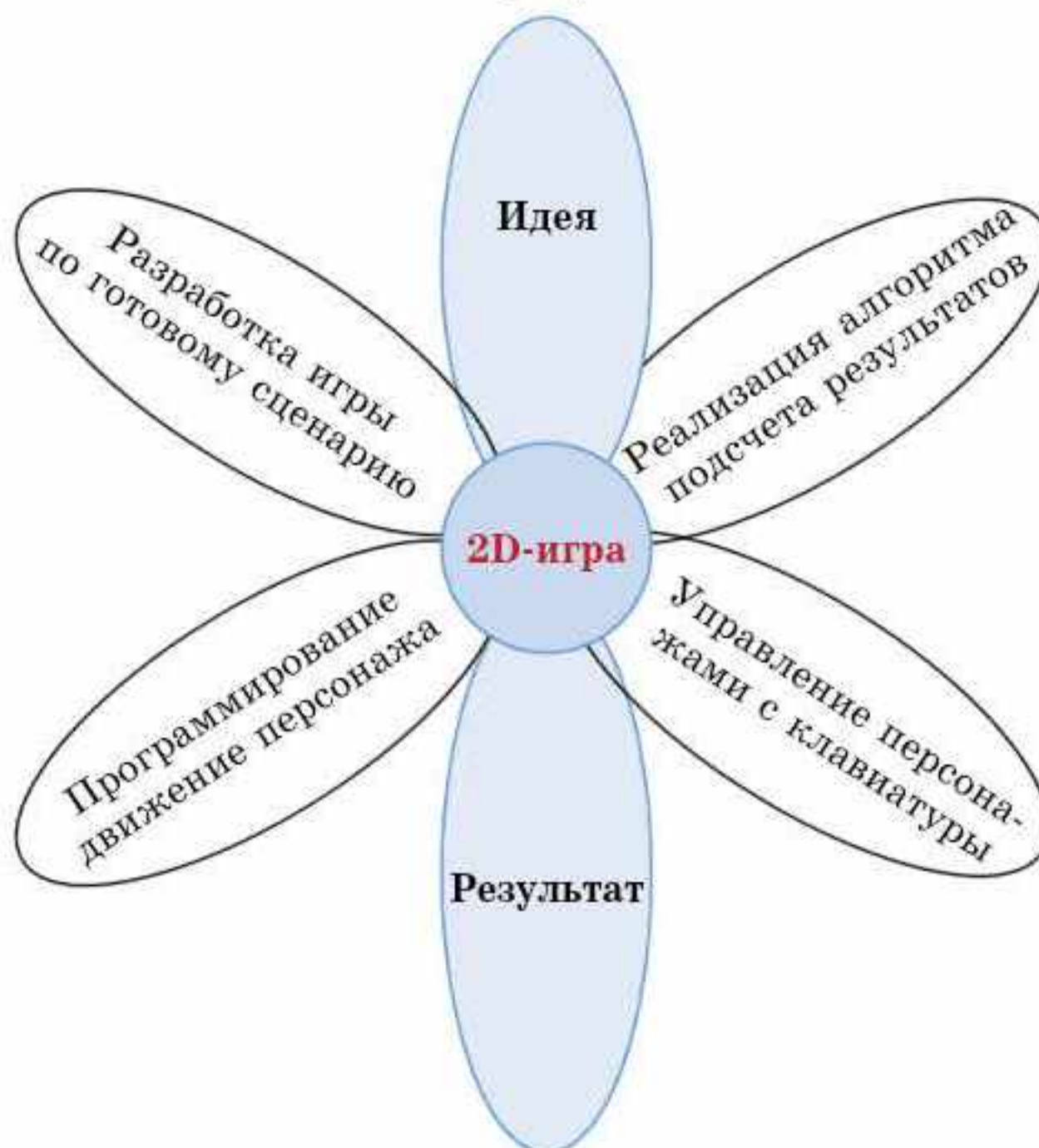
10) Напишите программу, которая удаляет из списка вещественных чисел элементы, значения которых больше 5,7 и меньше 10,8.

11) Дан список целых чисел, число n и значение M . Необходимо вставить в список на позицию с индексом n элемент, равный M , сдвинув все элементы, имевшие индекс не менее n , вправо.

Так как при этом количество элементов в списке увеличивается, после считывания списка в его конец нужно будет добавить новый элемент, используя метод **append**. Вставку необходимо осуществлять уже в считанном списке, не делая этого при выводе и не создавая дополнительного списка.

12) Даны два числа: количество строк k и количество столбцов m . Необходимо создать список размером $k \times m$, заполненный единицами.

СИСТЕМА ОСНОВНЫХ ПОНЯТИЙ РАЗДЕЛА V



Раздел V

СОЗДАНИЕ 2D-ИГРЫ НА ЯЗЫКЕ ПРОГРАММИРОВАНИЕ PYTHON

5.1. Библиотека PyGame

Вы научитесь:

- подключать библиотеку PyGame;
- использовать готовые модули библиотеки PyGame для создания окна для игры.

Ключевые слова

*Библиотека**Персонаж**Модуль PyGame**Элемент**Кітапхана**Кейіпкер**PyGame модулі**Элемент**Library**Character**PyGame module**Element*

Дайте общую трактовку ключевых слов и предположите, как они будут фигурировать в последующем тексте по теме.



Двумерная графика и 2D-игры появились в начале 1980 г. Создание 2D-игры происходит в двумерном пространстве, состоящем из двух осей координат: X и Y. В настоящее время большое предпочтение отдается 3D-графике (X, Y, Z-высота) и 3D-играм. Но инструменты для создания 3D-игр достаточно сложные и привлекают команды, состоящие из сотен разработчиков. Поэтому в качестве начального старта создания игр начнем с 2D-игры.



Подумайте и ответьте.

Какие 2D-игры вам знакомы? Возможно ли создавать игры в среде программирования Python?



При создании игр разрабатываются новые программы, написание которых затрачивается длительное время. Чтобы его не тратить, можно применять готовые фрагменты программ из пакетов библиотеки, делающей программу более мощной. Для создания игр в среде программирования Python применяется библиотека PyGame.



PyGame – это библиотека модулей для языка программирования Python, предназначенная для написания 2D-игр и мультимедиа-приложений. Библиотека содержит звуковые модули, специальную графику, функции, работающие с мультимедийными устройствами компьютера.

Поскольку библиотека PyGame не является стандартной библиотекой языка программирования Python, она устанавливается отдельно. PyGame может работать на многих платформах и операционных системах.



Установите библиотеку PyGame на свой компьютер. Введите код программы, указанный ниже в IDLE.



Перед установкой PyGame необходимо, чтобы на компьютер была установлена среда разработки Python. Чтобы узнать, какая версия Python установлена, выполните в среде IDLE (окно кода) Python: **Сохраните – Запустите модуль – Выведите:**

```
import sys
print (sys.version)
```

После запуска на экране в области вывода появится результат, например:

```
3.4.4 (v3.4.4:737efcadf5a6, Dec 20 2015, 20:20:57) [MSC v.1600 64 bit (AMD64)]
>>>
```

В зависимости от версии Python загружаете и устанавливаете соответствующую версию PyGame с сайта (pygame.org/). На рис. 5.1 изображена установка Pygame-1.9.2a0.win-amd64-py3.4 для Python 3.4.4.



Рис. 5.1. Установка PyGame

Для корректной работы библиотеки PyGame необходимо, чтобы она совпадала с версией среды программирования Python.

Далее необходимо проверить, можно ли импортировать PyGame в качестве модуля Python, набрав команду в IDLE:

```
import pygame
```

Если на экране нет сообщения об ошибке, то вы правильно установили PyGame на свой компьютер.

Библиотека PyGame – это библиотека модулей для языка программирования Python, облегчающая разработку 2D-игр. Большими преимуществами PyGame являются легкое обучение и быстрота разработки игр. Созданные в PyGame 2D-игры выполняются в окнах.

Поэкспериментируйте.

Составьте программу создания окна для игры с применением библиотеки PyGame.

Введите код программы в IDLE – Сохраните – Запустите модуль – Выведите.



Шаг за шагом

Рассмотрим простой способ создания окна с помощью библиотеки PyGame. Созданный объект является поверхностным, поэтому любой рисунок на нем становится видимым на экране (рис. 5.2).



Рис. 5.2. Окно для игры

В табл. 5.1 приведены код программы и пошаговый алгоритм создания окна.

Таблица 5.1

Создание окна для игры

№	Код программы	Алгоритм выполнения
1	<code>import pygame</code>	Подключение библиотеки PyGame
2	<code>FPS = 60</code>	Определение константы частоты составляет 60 кадров в секунду. Это означает, что за секунду в памяти компьютера произойдет 60 обновлений, соответствующих экранному пикселю
3	<code>pygame.init()</code>	Инициализация функцией <code>init()</code> инструментария PyGame
4	<code>pygame.display.set_mode((600, 400), pygame.RESIZABLE)</code>	Создание функцией <code>set_mode()</code> окна размером 600×400 и его изменение. Возвращение объекта типа Surface (поверхность)
5	<code>clock = pygame.time.Clock()</code>	Модуль <code>time</code> возвращает время, которое прошло с момента первого вызова данной функции
6	<code>pygame.display.update()</code>	Отображение объектов на экране до цикла
7	<code>pygame.display.set_caption("Game")</code>	Установка заголовка окна с надписью "Game"
8	<code>while True:</code>	Пока <code>while True</code> , выполняется цикл
9	<code>clock.tick(FPS)</code>	Задержка методом <code>tick()</code> класса <code>Clock</code> означает, что на каждой итерации цикла секунда делится на 60. Затем на эту величину выполняется задержка: – задержка вставляется в начало цикла, если до начала цикла происходит отображение каких-либо объектов на экране; – задержка вставляется в конец цикла, если первое появление объектов на экране происходит внутри цикла
10	<code>for i in pygame.event.get():</code>	Цикл обработки событий отслеживает только те события, у которых значение свойства <code>type</code> совпадает со значением константы <code>QUIT</code> . Если происходит событие при нажатии на "+", то это значение присваивается <code>type</code> . Функция <code>quit()</code> модуля PyGame завершает его работу
11	<code>if i.type == pygame.QUIT:</code>	
12	<code>quit()</code>	
13	<code>pygame.display.update()</code>	Обновление содержимого окна игры означает, что заново устанавливается цвет каждого пикселя

```

# Подключение модуля
import pygame
# Определение константы
FPS = 60
# Инициализация библиотеки
pygame.init()
pygame.display.set_mode((600, 400))
pygame.display.set_caption("Game")
clock = pygame.time.Clock()
# Отображение объектов на экране до цикла
pygame.display.update()
# Цикл
while True:
    # Задержка
    clock.tick(FPS)
    # Цикл обработки событий
    for i in pygame.event.get():
        if i.type == pygame.QUIT:
            quit()
    # Обновление экрана
    pygame.display.update()

```



Рис. 5.3. Код программы, результат, диалоговое окно

При нажатии на "+" происходит событие. На экране появляется диалоговое окно. Нажимаем команду **OK**, и оно закрывается.

В PyGame очень часто применяются события. **Событие** – это объект класса **Event** с атрибутами (свойствами и методами). Например, если не инициализирован модуль отображения **display** библиотеки PyGame и не установлен видеорежим, то не будет работать очередь **событий**.



Событием называется действие, выполняемое пользователем во время игры. Например, при нажатии клавиш клавиатуры или мыши событие создается, размещается в очереди, дожидается обработки.

Библиотека PyGame отслеживает все сообщения о событиях через очередь событий. Процедуры помогают управлять очередью событий (табл. 5.2).

Таблица 5.2

Основные модули Pygame.event для обработки очереди событий

Модуль	Назначение
<code>pygame.event.get</code>	Получает события из очереди
<code>pygame.event.wait</code>	Ждет одиночного события из очереди
<code>pygame.event.clear</code>	Удаляет все события из очереди
<code>pygame.event.set_grab</code>	Проверяет совместное использование устройств ввода
<code>pygame.event.post</code>	Размещает новое событие в очередь
<code>pygame.event.Event</code>	Создает новый объект события

Для эффективной разработки игр рекомендуется применение модулей библиотеки PyGame (табл. 5.3).

Таблица 5.3

Основные модули библиотеки PyGame

<code>pygame.cursors</code>	Загружает изображения курсора
<code>pygame.display</code>	Предоставляет доступ к дисплею
<code>pygame.draw</code>	Рисует фигуры, линии и точки
<code>pygame.event</code>	Управляет внешними событиями
<code>pygame.font</code>	Использует системные шрифты
<code>pygame.image</code>	Загружает и сохраняет изображение
<code>pygame.key</code>	Считывает нажатия клавиш с клавиатуры
<code>pygame.mixer</code>	Загружает и воспроизводит мелодии
<code>pygame.mouse</code>	Управляет мышью
<code>pygame.movie</code>	Воспроизводит видеофайлы
<code>pygame.music</code>	Работает с музыкой и потоковым аудио
<code>pygame.rect</code>	Управляет прямоугольными областями
<code>pygame.sprite</code>	Управляет движущимися изображениями
<code>pygame.surface</code>	Управляет изображениями и экраном
<code>pygame.time</code>	Управляет временем и частотой кадров
<code>pygame.transform</code>	Изменяет размер и перемещает изображения



Знание и понимание



1. Что такое *библиотека PyGame*?
2. С помощью какой команды можно узнать версию среды программирования Python?
3. Каким образом можно установить библиотеку PyGame в среду программирования Python?
4. Почему необходимо, чтобы версии среды программирования Python и библиотеки PyGame соответствовали?
5. В чем преимущества применения PyGame при создании 2D-игр?
6. Что такое *событие*? Какие события существуют в библиотеке PyGame?
7. Какие функции облегчают создание окна для игр?
8. Какие основные модули библиотеки PyGame были использованы в коде программы при создании окна?
9. Составьте и запишите свою версию рассказа, употребив все предложенные ключевые термины по теме.



Применение

Анализ



10. Что появится после запуска кода на экране?
`size=[700,500]`
`screen=pygame.display.set_mode(size)`
11. Дан алгоритм создания окна. Напишите программу.
Введите код программы в IDLE – Сохраните – Запустите модуль – Выведите.
Алгоритм выполнения
 - 1) Импорт библиотеки PyGame.
 - 2) Инициализация PyGame.
 - 3) Установка ширины и высоты окна в пикселях.
 - 4) Создание окна функцией `Set_mode()` модуля `display`, библиотеки PyGame.
12. Дан код программы. Допишите комментарии к ней.

```
import pygame
size=[600,300]
window=pygame.display.set_mode((size))
pygame.display.set_caption("Window")
screen=pygame.Surface((400,400))
gt=True
while gt:
for i in pygame.event.get():
if i.type==pygame.QUIT:
gt=False
window.blit(screen,(0,0))
pygame.display.flip()
```

Ответьте на вопросы.

- 1) Введите код программы в IDLE – Сохраните – Запустите модуль – Выведите.
- 2) Проанализируйте, что будет происходить без цикла.
- 3) Что произойдет, если параметры игрового экрана отличаются от параметров окна?
- 4) Можно ли изменить размер игрового экрана?
- 5) Можно ли написать код программы без применения функции flip()?



Синтез

Оценивание



13. Допишите программу создания окна PyGame. Ниже приведена структура программы, написанной с использованием PyGame, которая начинается с пустого окна.

```
import pygame
pygame.init()
# создание объекта экрана
screen = pygame.display.set_mode()
pygame.display.set_caption("Window")

while True:
# отслеживание событий
    for event in pygame.event.get():
        if event.type == pygame.QUIT:
```

14. Тест по Python. В каждом вопросе только один правильный ответ.

1. Что будет результатом этого кода:

```
x = 23
num = 0 if x > 10 else 11
```

- a) 23;
- b) 10;
- c) 11;
- d) 0?

2. Сколько библиотек можно импортировать в один проект:

- a) не более 3;
- b) не более 10;
- c) не более 5;
- d) не более 23;
- e) неограниченное количество?

3. Что покажет этот код:

```
for i in range(5):  
    if i % 2 == 0:  
        continue  
    print(i)
```

- a) ошибку, так как *i* не присвоена;
- b) ошибку из-за неверного вывода;
- c) числа: 1, 3 и 5;
- d) числа: 1, 3;
- e) числа: 0, 2 и 4?

4. Где правильно создана переменная:

- a) `int num = 2;`
- b) `num = 2;`
- c) `var num = 2;`
- d) `$num = 2;`
- e) `num = float(2)?`

5. Как получить данные от пользователя:

- a) использовать метод `get ()`;
- b) использовать метод `cin ()`;
- c) использовать метод `read ()`;
- d) использовать метод `readLine ()`;
- e) использовать метод `input ()`?

6. Какая библиотека отвечает за время:

- a) `localtime;`
- b) `clock;`
- c) `Time;`
- d) `time;`
- e) `Clock?`

7. Что будет показано в результате?

- a) "Hi, name"
- b) "Hi, "
- c) Ошибка
- d) "Hi, John"
- e) "Hi, name John"

8. Какая функция выводит что-либо:

- a) `write ()`;
- b) `log ()`;
- c) `out ()`;
- d) `print ()`?
- e) `input ()`

9. Что покажет этот код:

```
for j in 'Hi! I\'m mister Robert':
    if j == '\\':
        print ("Найдено")
        break
```

else:

```
    print ("Готово")
```

- a) ошибку в коде;
- b) "Найдено" и "Готово";
- c) "Готово";
- d) "Найдено";
- e) "По вашему запросу ничего не найдено"?

10. Что выведет следующий код:

```
list(range(3))[2]
```

- a) 2;
- b) 3;
- c) 5;
- d) 1?
- e) 4.



Создание окна с помощью модуля Tkinter из стандартной библиотеки Python.

С помощью **Tkinter** кроссплатформенной библиотеки для разработки графического интерфейса на языке Python можно быстро создавать окна. *Tkinter* расшифровывается как «Tk interface». Рассмотрим самый простой способ создания окна с использованием модуля **Tkinter** из стандартной библиотеки Python:

```
from Tkinter import*           # импорт Tkinter из библиотеки Python
window=Tk()                    # Tk создание окна
```



Создание окна модулем Tkinter

5.2. Задний фон и персонаж игры

Вы научитесь:

– создавать задний фон игры и загружать готовые персонажи для игры.

Ключевые слова

Задний фон
Персонаж игры
Окно

Артқы фон
Ойын кейіпкері
Терезе

The background
Game character
Window



Дайте общую трактовку ключевых слов и предположите, как они будут фигурировать в последующем тексте по теме.



С помощью Python и библиотеки PyGame можно создавать собственные игры. В библиотеке PyGame много функций и инструментов, облегчающих создание игр. При создании даже самой простой игры необходимо продумать ее идею. Для этого надо ответить на следующие вопросы:



О чем будет эта игра? Каким будет ее дизайн? Каков жанр игры? Кто будет главным персонажем игры? Каким будет фон игры?



Построение игры: создание окна, заливка заднего фона цветом, загрузка персонажа игры.



Шаг за шагом

Для представления внешнего вида поверхности или фона нужно залить его любым цветом, выбрав значение из табл. 5.4. Цвета в PyGame задаются в схеме **RGB**: **R**ed – красный, **G**reen – зеленый и **B**lue – синий (рис. 5.4). Если смешивать эти цвета, то можно получить любой цвет в диапазоне значений от 0 до 255.

Таблица 5.4

Цвета в PyGame

Цвет		Диапазон
aqua	морская волна	(0, 255, 255)
black	черный	(0, 0, 0)
blue	синий	(0, 0, 255)
fuchsia	фуксия	(255, 0, 255)
gray	серый	(128, 128, 128)
green	зеленый	(0, 128, 0)
lime	лайм	(0, 255, 0)
maroon	темно-бордовый	(128, 0, 0)
navy_blue	темно-синий	(0, 0, 128)
olive	оливковый	(128, 128, 0)
purple	фиолетовый	(128, 0, 128)
red	красный	(255, 0, 0)
silver	серебряный	(192, 192, 192)
teal	зелено-голубой	(0, 128, 128)
white	белый	(255, 255, 255)
yellow	желтый	(255, 255, 0)



Рис. 5.4. Схема RGB

Для заполнения фона цветом в PyGame применяется метод `screen.fill()`. Он получает всего один аргумент – цвет фона. Например, при создании голубого фона запись команды будет выглядеть: `screen.fill(blue)` или `screen.fill(0,0,255)`. По умолчанию в PyGame устанавливается черный цвет.

Объект `screen` называется **поверхностью** (`surface`). Поверхность в PyGame представляет собой часть экрана, на которой отображается игровой объект или персонаж. Поэтому каждый объект в игре представляется также в виде поверхности. Например, геометрическая фигура, пришелец, человек, робот и др.

Рассмотрим программу, в которой создано окно, задний фон которого в последующем заливается красным цветом (рис. 5.5).

Код программы

```
import pygame
pygame.init()
width = 300
height = 200
screen=pygame.display.set_mode((width, height))
pygame.display.set_caption('My Game')
red = (255, 0, 0)
screen.fill(red)
pygame.display.flip()
running = true
while running:
    for event in pygame.event.get():
        if event.type == pygame.QUIT:
            running = false
pygame.quit()
```

Результат

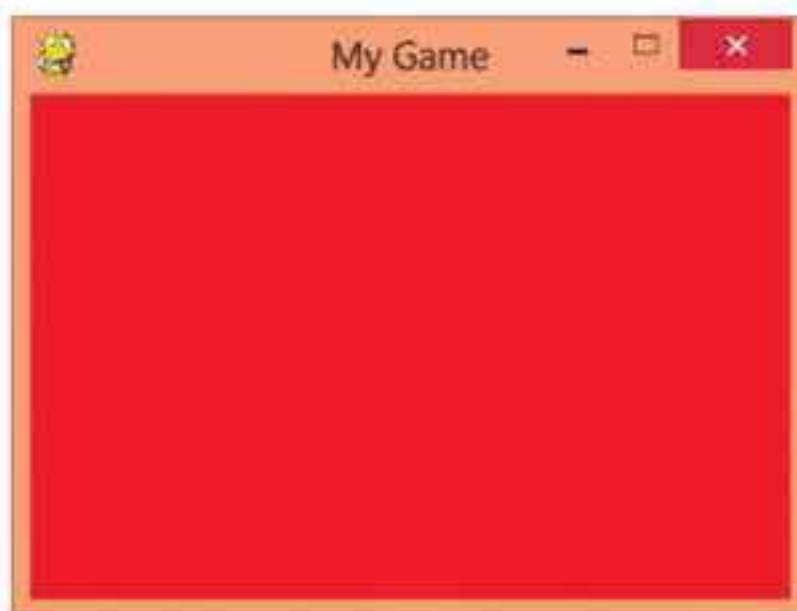


Рис. 5.5. Окно с фоном красного цвета



Как вы думаете, интересен ли однотонный фон игры? Можно ли загрузить свои изображения для фона?



В PyGame фон (поверхности) также можно создавать вызовом конструктора класса **Surface**. При создании необходимо указывать ширину и высоту: `surf = pygame.Surface((150, 150))`.

Метод **blit()** применяется к родительской Surface поверхности, а дочерняя передается в качестве аргумента. Например: `sc.blit(surf, (50, 20))`, где `sc` – основная поверхность с координатами (50×20).

Более подробно рассмотрим, как можно загрузить изображения в виде фона. Это возможно выполнить функцией **load()** модуля `pygame.image`. В **load()** передается имя файла форматом **.bmp**. Если функция `pygame.image.get_extended()` возвращает истину, то можно загрузить ряд других форматов: **png, gif, jpg** и т. д.

Попробуйте самостоятельно выполнить загрузку изображения в код программы и получить результат (рис. 5.6). При загрузке изображения необходимо, чтобы размер рисунка совпадал с размером окна.

Код программы

```
import pygame
pygame.init()
sc=pygame.display.set
_mode((400,416))
sc.fill((100, 150, 200))
img_surf=pygame.image.load
('bater.bmp')
img_rect=img_surf.get_rect
(bottomright=(400,416))
sc.blit(img_surf, img_rect)
pygame.display.update()
while 1:
    for i in pygame.event.get():
        if i.type==pygame.QUIT:
            exit()
    pygame.time.delay(20)
```

Результат

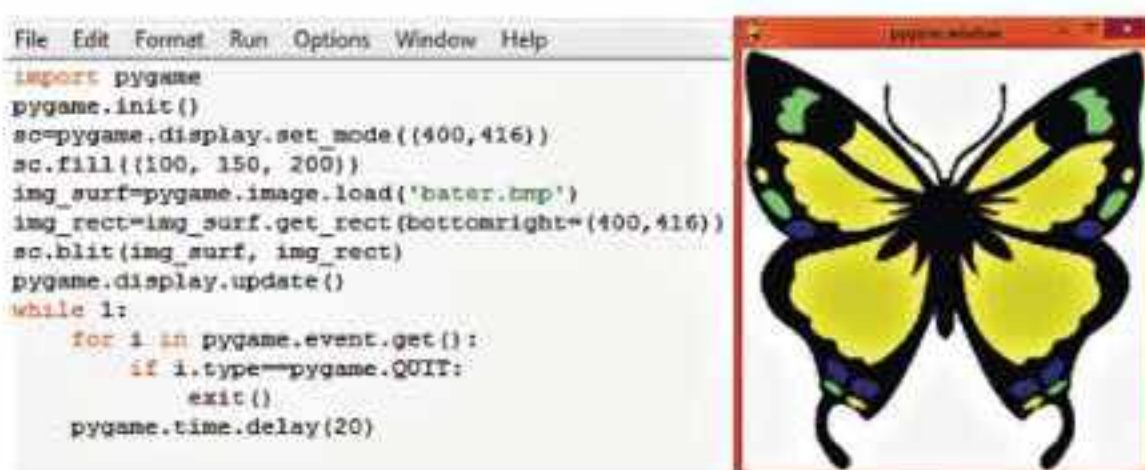


Рис. 5.6. Фон окна – изображение

Перед основным циклом с помощью функции `pygame.image.load()` загружается фоновое изображение. Затем выполняется `sc.blit` блиттинг (копирование битов) изображения на экран в позицию (0,0). В результате на экране появляется изображение.

Для любой игры необходимы **персонажи**. Они могут быть в виде нарисованного объекта или готового изображения. Рассмотрите, каким образом можно загрузить готового персонажа на поверхность окна (рис. 5.7). Проанализируйте, какие библиотеки, модули, функции, методы были использованы в программе.

Код программы

```

import pygame
pygame.init()
sc = pygame.display.set_mode((300, 300))
sc.fill((255, 255, 255))
pygame.display.set_caption('My Game')
a_surf = pygame.image.load('auto.bmp').convert()
a_surf.set_colorkey((255, 255, 255))
a_rect = a_surf.get_rect(center=(130, 130))
sc.blit(a_surf, a_rect)
pygame.display.update()
while 1:
    for i in pygame.event.get():
        if i.type == pygame.QUIT:
            exit()
        elif i.type == pygame.KEYUP and
i.key == pygame.K_f:
            flip = pygame.transform.flip(a_surf, 1, 0)
            sc.fill((100, 150, 200))
            sc.blit(flip, a_rect)
            pygame.display.update(a_rect)
pygame.time.delay(20)

```

Результат**Рис. 5.7.** Готовый персонаж**Знание и понимание**

1. Для чего нужно создавать задний фон?
2. Как можно использовать готовые изображения для заднего фона?
3. Какие две команды должны выполняться, прежде чем программа сможет использовать функцию `pygame.display.set_mode()`?
4. Что делает функция `pygame.display.set_mode()`?
5. Для чего используется `pygame.time.Clock`?
6. Что делает цикл `for event in pygame.event.get()`?
7. Какие функции создают геометрические объекты на экране?
8. Каким образом можно загрузить персонаж для игры?
9. Можно ли нарисовать свой персонаж для игры?
10. Почему возникает иллюзия анимации персонажей?

11. Дан код программы. Прокомментируйте программу. Что будет выведено на экран в результате ее выполнения?

```
import pygame
# Цвета
black = (0, 0, 0)
white = (255, 255, 255)
green = (0, 255, 0)
red = (255, 0, 0)
pygame.init()
# Размер окна
size = (700, 500)
screen = pygame.display.set_mode(size)
pygame.display.set_caption("My Game")
done = False
clock = pygame.time.Clock()
while not done:
    # События
    for event in pygame.event.get():
        if event.type == pygame.QUIT:
            done = True
    screen.fill(WHITE)
    pygame.display.flip()
    clock.tick(60)
pygame.quit()
```

12. Составьте и запишите свою версию рассказа, употребив все предложенные ключевые термины по теме.



Применение

Анализ



13. Напишите программу, которая выводит окно на экран (рисунок «Окно с поверхностью»). В программе примените методы и функции создания фона на поверхности Surface.

14. Создайте окно, в котором цикл программы постоянно заполняет экран серым цветом с помощью метода `screen.fill((192, 192, 192))` и вызывает метод таймера `tick()` с частотой кадров. Для пояснения кода программы напишите комментарии.

15. Дан код программы. Что будет выведено на экран в результате ее выполнения? С какими новыми командами программы вы познакомились?



Окно с поверхностью



Синтез

Оценивание



```
import pygame, sys
from pygame.locals import *
pygame.init()
screen = pygame.display.set_mode((300,200),0,32)
pygame.display.set_caption("My first PyGame");
helloText = "Hello, World!"
x,y,fontSize = (70,40,40)
myFont = pygame.font.SysFont("None", fontSize)
fontColor = (128,0,0)
bgColor = (0,255,255)
fontImage = myFont.render(helloText, 0, (fontColor))
mainLoop = True
while mainLoop:
for event in pygame.event.get():
    if event.type == QUIT:
        mainLoop = False
screen.fill(bgColor)
screen.blit(fontImage,(x,y))
pygame.display.update()
pygame.quit()
```

16. Создайте проект, применив темы из многонационального творческого колорита народа Казахстана. Продумайте идею, дизайн, фон окна, главного персонажа проекта.

17. Напишите код анимации квадрата, который перемещается от левой границы к правой, касается ее, но не исчезает за ней. После этого возвращается назад – от правой границы к левой, касается ее, опять двигается вправо. Циклы движения квадрата повторяются до завершения программы.



Основные принципы программирования игр.

1. Главное в играх – перемещение пикселей на экране.
2. Обновление экрана через фиксированные интервалы времени (частота кадров) с помощью цикла. Внутри цикла выполняются три основные операции: обработка событий, обновление состояния игры и отрисовка текущего состояния на экране.
3. События в игре – это все, что происходит за пределами управления кода игры, относится к выполнению игры.

4. Игре нужно отображать свое состояние на экране, в том числе отрисовку геометрических фигур, изображения и текст.
5. В большинстве игр симулируется физическое окружение. В более сложных играх могут использоваться реалистичные физические системы (особенно в 3D-играх).
6. Подключение во многих играх искусственного интеллекта.
7. Воспроизведение звука – важный аспект игр. Существует два типа звука: фоновая музыка и звуковые эффекты.
8. В играх жизнь, очки и уровни дают мотивацию и самосовершенствование.

5.3. Анимирование персонажей

Вы научитесь:

- программировать движение персонажа;
- управлять персонажами с клавиатуры.

Ключевые слова

Анимация

Движение

Событие

Кадр

Частота кадров

Анимация

Қозғалыс

Оқиға

Кадр

Кадр жиілігі

Animation

Motion

Event

Frame

Frame frequency



Дайте общую трактовку ключевых слов и предположите, как они будут фигурировать в последующем тексте по теме.



В предыдущем параграфе мы рассмотрели, как перемещается персонаж по окну, создавая иллюзию анимации. Что такое *анимация*?



Анимация (фр. *animation* – «оживление, одушевление») – это быстрая смена изображений на экране, создающая эффект движения.

Алгоритм движения заключается в следующем. Рисуем объект на поверхности экрана. Обновляем экран и видим картину. Стираем объект и рисуем его с небольшим смещением от первоначальной позиции. Снова

обновляем окно и т. д. Если не сотрем старое изображение, то на экране будут видны оба изображения.

Поэтому стираем изображение, рисуем его на новом месте, обновляем экран, и в итоге достигаем иллюзии движения объектов. Эти этапы повторяются многократно.

Чтобы предыдущее положение объекта удалялось, добавляем функцию `fill()`. Функция заполняет окно цветом фона и предыдущий рисунок как бы стирается.

<code>screen.fill((код цвета))</code>	Оператор заполнения окна цветом, цвет указываем в кодировке RGB.
---------------------------------------	------------------------------------------------------------------

При обновлении экрана обновляется все то, что программа отслеживает (позиция игрока и скорость движения, жизни, очки и т. д.). Для обновления поверхности экрана используем функцию `flip()` и `update()`

<code>pygame.display.flip()</code>	Обновляет всю поверхность отображения.
<code>pygame.display.update()</code>	Обновляет только часть изображения, что экономит память.

Обновление экрана происходит с определенной частотой.



Частота кадров (частота обновления экрана) – это количество изображений, которые программа рисует в секунду, измеряется в **FPS (Frames Per Second)** кадрах в секунду.

Цикл крутится со скоростью, зависящей от мощности компьютера. Человек дает команды и воспринимает изменения медленнее, чем компьютер. В небольших играх для обновления экрана в главном цикле выполняют задержку. Делают это двумя способами.

Первый способ – вызов функции `delay()` модуля `time` библиотеки `pygame`.

<code>pygame.time.delay()</code>	Частота обновления кадров. В скобках пишется количество миллисекунд (1000 мс = 1 с). Чем меньше значение в скобках, тем быстрее будет обновляться игра, так как цикл вызывается большее количество раз.
----------------------------------	---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

Второй способ – создают объект **Clock** и устанавливают ему частоту кадров. Методу **tick** класса **clock** передается желаемое количество кадров. Задержку он вычисляет сам.

clock.tick(количество кадров) | В скобках указывается количество кадров.



Напишите программу движения геометрической фигуры по оси X. Фигура появляется с левой стороны, доходит до правой и исчезает. После этого снова появляется слева.



Шаг за шагом

```

Движение фигуры по оси x.py - C:\Python34\Программки\Да...
File Edit Format Run Options Window Help
import pygame

pygame.init()

clock = pygame.time.Clock()

sc=pygame.display.set_mode((800,200))

pygame.display.set_caption("Движение фигуры по оси X")

r = 50
x = 0 - r
y = 200 // 2

run=True
while run:
    sc.fill((0,0,255))
    for event in pygame.event.get():
        if event.type==pygame.QUIT:
            run=False

    pygame.draw.circle(sc, (225,225,0), (x, y), r)
    pygame.display.update()

    if x >= 800 + r:
        x = 0 - r
    else:
        x += 2

    clock.tick(40)

pygame.quit()
Ln:1 Col:0

```

До начала цикла задаем радиус и координаты центра круга:
 1) r – радиус, X и Y координаты центра круга;
 2) 0 – левая граница окна, круг появляется из-за левой границы окна;
 3) выравнивание круга посередине (по вертикали).

Заполняем окно цветом фона, чтобы предыдущий рисунок как бы стирался.

(if) Если круг полностью скрывается за правой границей, то появляется из-за левой границы.
 (else) Новая позиция центра круга по оси X, новый круг рисуется со смещением на 2 пикселя.

Рис. 5.8



Поэкспериментируйте.

Составьте программу движения геометрической фигуры по оси Y со смещением по оси X при каждом круге.



Как можно управлять движением персонажа в игре с помощью клавиатуры? Движениями персонажей можно управлять с клавиатуры с помощью стандартных событий.

<code>pygame.keydown</code>	Тип события (нажатие на кнопку)
<code>pygame.K_DOWN, pygame.K_UP</code>	«Ключи» события (<code>event.key</code>)

Пример записи кода программы одиночного нажатия на клавишу, при этом объект передвинется только один раз:

```
if event.type == pygame.KEYDOWN:
    if event.key == pygame.K_UP:
        vel_y=vel_y-10
    if event.key == pygame.K_DOWN:
        vel_y=vel_y+10
```

Чтобы перемещать игрока, нужно создать список, нажимая и удерживая клавишу.

<code>keys = pygame.key.get_pressed()</code>	Создание списка. Помещаем все кнопки, на которые будем нажимать, заставляя игрока двигаться.
----------------------------------------------	----------------------------------------------------------------------------------------------

Список всех клавиш, которые будем использовать.

<pre>if keys [pygame.K_LEFT] x- = speed if keys [pygame.K_RIGHT] x+ = speed if keys [pygame.K_UP] y- = speed if keys [pygame.K_DOWN] y+ = speed</pre>	<p>«Стрелка влево» – из координаты X вычитаем скорость (движение по оси X влево);</p> <p>«стрелка вправо» – к координате X прибавляем скорость (движение по оси X вправо);</p> <p>«стрелка вверх» – из координаты Y вычитаем скорость (движение по оси Y вверх);</p> <p>«стрелка вниз» – к координате Y прибавляем скорость (движение по оси Y вниз).</p>
-----------------------------------------------------------------------------------------------------------------------------------------------------------------------	-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------



Напишите программу, в которой объект передвигают вверх, вниз, влево, вправо с помощью клавиш.



Шаг за шагом

Мы уже знаем простейшую структуру, в которой импортируем и инициализируем `Pygame`, определяем размеры окна и даем ему название.

Нам необходимы пять переменных, которые будут отвечать за нашего игрока: его начальные координаты, откуда он начнет движение, размеры объекта, скорость движения.

<pre>x=20 y=20 width=50 height=70 speed=10</pre>	<p>начальная координата X начальная координата Y ширина игрока высота игрока скорость движения</p>
--------------------------------------------------	--------------------------------------------------------------------------------------------------------------------

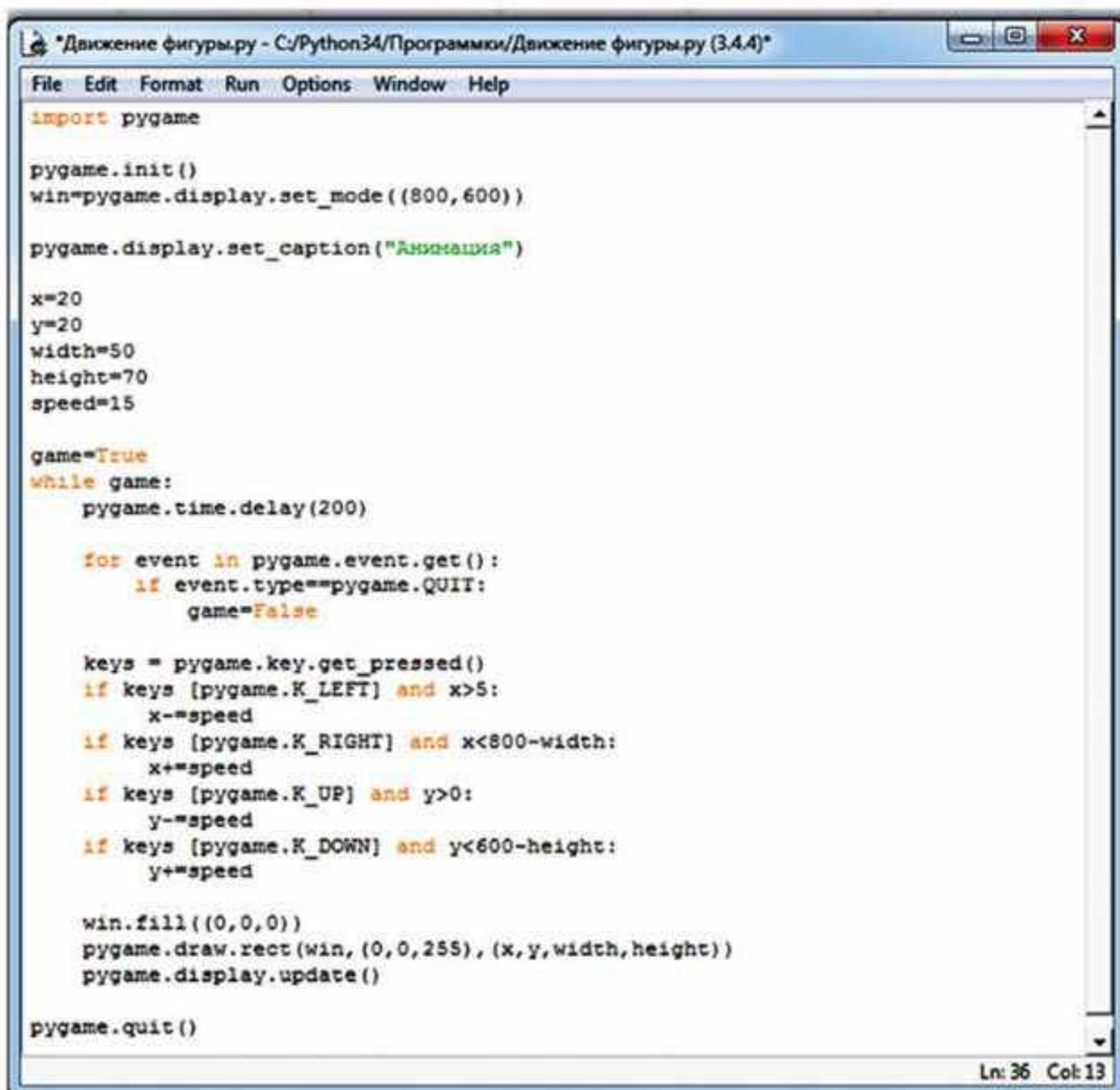
Создаем бесконечный цикл, так как часть кода, отвечающего за движение игрока, обрабатывается только в цикле. Выполним задержку в цикле, чтобы он не выполнялся слишком быстро.

<pre>game=True while game: pygame.time.delay (200)</pre>	<p>Пока переменная <истина>, цикл будет выполняться бесконечное число раз. Цикл будет повторяться каждые 0,2 секунды.</p>
--------------------------------------------------------------	-------------------------------------------------------------------------------------------------------------------------------------

Создаем список, помещаем туда все клавиши, с помощью которых будем передвигать наш объект. Чтобы объект не уходил за границы окна, добавляем проверку границ.

<pre>keys = pygame.key.get_pressed() if keys [pygame.K_LEFT] and x>0 x-=speed if keys [pygame.K_RIGHT] and x<800-width x+=speed if keys [pygame.K_UP] and y>0 y-=speed if keys [pygame.K_DOWN] and y<600-height y+=speed</pre>	
------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	--

Готовый программный код выглядит так (рис. 5.9):



```

File Edit Format Run Options Window Help
import pygame

pygame.init()
win=pygame.display.set_mode((800,600))

pygame.display.set_caption("Анимация")

x=20
y=20
width=50
height=70
speed=15

game=True
while game:
    pygame.time.delay(200)

    for event in pygame.event.get():
        if event.type==pygame.QUIT:
            game=False

    keys = pygame.key.get_pressed()
    if keys [pygame.K_LEFT] and x>5:
        x-=speed
    if keys [pygame.K_RIGHT] and x<800-width:
        x+=speed
    if keys [pygame.K_UP] and y>0:
        y-=speed
    if keys [pygame.K_DOWN] and y<600-height:
        y+=speed

    win.fill((0,0,0))
    pygame.draw.rect(win, (0,0,255), (x,y,width,height))
    pygame.display.update()

pygame.quit()
Ln: 36 Col: 13

```

Рис. 5.9. Программный код

Добавим нашему объекту возможность прыгать. Для этого добавим еще две переменные. В первой указываем, прыгает объект или нет, во второй – высоту прыжка.

```

Is_Jump = False
Jump_Count = 10

```

Определяем клавишу при нажатии, на которую объект прыгает.

```

if not(Is_Jump):

```

Пусть он прыгает, когда мы нажмем клавишу «пробел».

```
if keys[pygame.K_SPACE]:
    Is_Jump = True
```

Когда персонаж прыгает, то сначала он поднимается вверх, затем, достигнув определенной точки, начинает падать вниз.

```
else:
    if Jump_Count >= -10:
        if Jump_Count < 0:
            y += (Jump_Count ** 2)/2
        else:
            y -= (Jump_Count ** 2)/2
            Jump_Count -= 1
        else:
            Is_Jump = False
            Jump_Count = 10
```

С каждым разом переменная `Jump_Count` будет уменьшаться, и, соответственно, квадрат числа тоже будет становиться все меньше. Объект будет все медленнее и медленнее подниматься вверх. В какой-то момент квадрат переменной станет равен нулю, объект перестанет двигаться. Он начнет медленно, а затем быстрее и быстрее приземляться. Отсчет пойдет в обратную сторону, и объект начнет с той же скоростью падать обратно.



Поэкспериментируйте.

Измените программу, в которой круг движется по оси X так, чтобы он прыгал при нажатии на какую-либо клавишу.



Для изменения и поворота объектов применяется модуль `pygame.transform`, который содержит функции для изменения поверхностей. Чтобы изменения (например, изменение размера) не приводили к ухудшению изображения из-за потери части пикселей, трансформацию выполняют с дополнительной поверхностью, которую создают на основной. Исходная поверхность сохраняется.

Функции модуля `transform`, которые изменяют поверхности, возвращают новые. Первым аргументом им передается исходный `Surface`.

<pre> sc = pygame.display.set_ mode((ширина, высота)) sc.fill((100, 150, 200)) surf = pygame.image.load('имя. bmp').convert() surf.set_colorkey((255, 255, 255)) rect = surf.get_rect(center=(200, 150)) sc.blit(surf, rect) </pre>	<p>Основная поверхность.</p> <p>Дополнительная поверхность в виде изображения для изменения.</p> <p>Прозрачный фон изображения.</p> <p>Создается класс <code>rect</code> для дополнительной поверхности.</p> <p>Метод <code>blit</code> применяется к основной поверхности.</p>
-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

Приведем наиболее востребованные функции модуля `pygame.transform`. Функция `flip()` переворачивает объект по горизонтали и вертикали без потери качества. Указывается поверхность и булевы значения оси переворота.

Пример программы поворота объекта слева направо:

```

flip = pygame.transform.flip(surf, 1, 0)
sc.fill((100, 150, 200))
sc.blit(flip, rect)
pygame.display.update(rect)

```

Пример программы уменьшения изображения в два раза:

```

scale = pygame.transform.scale(surf, (surf.get_width()//2,
surf.get_height()//2))
scale_rect = scale.get_rect(center=(200, 150))
sc.blit(scale, scale_rect)

```

Пример программы поворота на определенное количество градусов:

```

rot = pygame.transform.rotate(dog_surf, «указываем угол в градусах»)
rot_rect = rot.get_rect(center=(200, 150))
sc.blit(rot, rot_rect)

```



Поэкспериментируйте.

Создайте несколько геометрических фигур и напишите программу для их перемещения.





Знание и понимание



1. Что такое *анимация изображения*?
2. Как происходит анимация в `pygame`?
3. Какое назначение модуля `pygame.transform`?
4. Заполните таблицу для модулей, которые используются для перемещения, движения объекта.

Модуль <code>pygame</code>	Функции модуля	Назначение

5. Составьте и запишите свою версию рассказа, употребив все предложенные ключевые термины по теме.



Применение

Анализ



6. Напишите код анимации фигуры, которая перемещается от левой границы к правой, касается ее, но не исчезает за ней, а возвращается назад – от правой границы к левой, касается ее, опять двигается вправо. Циклы движения квадрата повторяются до завершения программы.
7. Измените программу, в которой объект передвигают вверх, вниз, влево, вправо так, чтобы он мог передвигался по диагонали.
8. Нарисуйте машинку. Напишите программу управления ее движением с помощью стрелок клавиатуры. Машинка всегда двигается только вперед.



Синтез

Оценивание



9. Дана программа. Что будет происходить с изображением собачки, когда мы запустим программу на выполнение?

```
import pygame
pygame.init()
sc = pygame.display.set_mode((400, 300))
sc.fill((100, 150, 200))
dog_surf = pygame.image.load('dog.bmp').convert()
dog_surf.set_colorkey((255, 255, 255))
dog_rect = dog_surf.get_rect(center=(200, 150))
sc.blit(dog_surf, dog_rect)
pygame.display.update()
while 1:
```

```

for i in pygame.event.get():
    if i.type == pygame.QUIT:
        exit()
    elif i.type == pygame.KEYUP and i.key == pygame.K_f:
        flip = pygame.transform.flip(dog_surf, 1, 0)
        sc.fill((100, 150, 200))
        sc.blit(flip, dog_rect)
        pygame.display.update(dog_rect)
        pygame.time.delay(20)

```

10. Дана программа. Измените ее так, чтобы, когда вы отпускали клавишу, круг возвращался в начальную точку.

```

import pygame
FPS = 60
W = 700 # ширина экрана
H = 300 # высота экрана
WHITE = (255, 255, 255)
BLUE = (0, 70, 225)
RIGHT = "to the right"
LEFT = "to the left"
STOP = "stop"
pygame.init()
sc = pygame.display.set_mode((W, H))
clock = pygame.time.Clock()
# координаты и радиус круга
x = W // 2
y = H // 2
r = 50
motion = STOP
while 1:
    sc.fill(WHITE)
    pygame.draw.circle(sc, BLUE, (x, y), r)
    pygame.display.update()
    for i in pygame.event.get():
        if i.type == pygame.QUIT:
            exit()
        elif i.type == pygame.KEYDOWN:
            if i.key == pygame.K_LEFT:
                motion = LEFT
            elif i.key == pygame.K_RIGHT:
                motion = RIGHT
            elif i.type == pygame.KEYUP:

```

```

if i.key in [pygame.K_LEFT, pygame.K_RIGHT]:
    motion = STOP
    if motion == LEFT:
        x -= 3
    elif motion == RIGHT:
        x += 3
    clock.tick(FPS)

```



Добавление фоновой музыки и звуков.

В Pygame для добавления фоновой музыки используется модуль `pygame.mixer.music`.

Функция `pygame.mixer.music.load()` загружает потоковое аудио, т. е. не грузит файл целиком, а делает это отдельными порциями. В результате можно проигрывать только один файл за раз. Однако можно ставить файлы в очередь функцией `queue()`. Поддерживает в том числе формат mp3.

С помощью функции `music.play()` файл начинает проигрываться. Если требуется зациклить композицию, то в `play()` передается число `-1`. Положительный аргумент указывает на количество повторов + одно дополнительное. То есть, если надо проиграть композицию два раза, то в функцию передается число `1`.

В программе при нажатии на клавишу «1» музыка ставится на паузу: `music.pause()`. Клавиша «2» уменьшает громкость в два раза: `music.set_volum(0.5)`. Нажатие на клавишу «3» возвращает громкость на прежний уровень. Функция `unpause()` вызывается на случай, если до этого музыка была выключена (клавишей «1»).

В примере создаются два объекта типа `Sound`. У них есть свой метод `play()`. В данном случае файлы проигрываются при клике левой и правой кнопками мыши. Объекты `Sound` могут проигрываться одновременно, так как обычно принадлежат разным каналам. Если требуется более тонкое управление звуками, дополнительно используют класс `Channel`.

В данном примере программы подгружается фоновая музыка: `pygame.mixer.music.load()`.

```

import pygame
pygame.init()
sc = pygame.display.set_mode((400, 300))
pygame.mixer.music.load('Test.mp3')
pygame.mixer.music.play()
sound1 = pygame.mixer.Sound('boom.wav')
sound2 = pygame.mixer.Sound('one.ogg')
while 1:

```

```

for i in pygame.event.get():
if i.type == pygame.QUIT:
exit()
elif i.type == pygame.KEYUP:
if i.key == pygame.K_1:
pygame.mixer.music.pause()
# pygame.mixer.music.stop()
elif i.key == pygame.K_2:
pygame.mixer.music.unpause()
# pygame.mixer.music.play()
pygame.mixer.music.set_volume(0.5)
elif i.key == pygame.K_3:
pygame.mixer.music.unpause()
# pygame.mixer.music.play()
pygame.mixer.music.set_volume(1)
elif i.type == pygame.MOUSEBUTTONDOWN:
if i.button == 1:
sound1.play()
elif i.button == 3:
sound2.play()
pygame.time.delay(20)

```

5.4. Программирование условий

Вы научитесь:

- разрабатывать игру по готовому сценарию;
- реализовывать алгоритм подсчета результатов игры.

Ключевые слова

Компьютерная игра

Сценарий игры

Сюжет

Игровая среда

Компьютерлік ойын

Ойын сценарийі

Сюжет

Ойын ортасы

Computer game

Scenario of game

Plot

Playing environment



Дайте общую трактовку ключевых слов и предположите, как они будут фигурировать в последующем тексте по теме.





Подумайте и ответьте.

Что такое *идея игры*? Что означает взаимодействие в игровой среде? Назовите этапы подготовки компьютерной игры.



Первое, что понадобится, это определиться с целью, идеей игры. Что в итоге мы хотим получить? Даже у самой простой игры есть свой сценарий.



Сценарий – это форма, в которой выражается сюжет.

Сюжет – это последовательность событий, которые складываются в историю.

В сценарии определяются и описываются персонажи, фон, игровая среда, планируются события и взаимодействия персонажей.

Игровая среда – это совокупность всех объектов и связей в игре и законов их изменения. Например, в шахматах игровой средой будет совокупность, в которую входят: доска, два набора фигур, правила перемещения фигур на доске, а также правила взятия (или превращения) фигур. Это важнейший компонент. Если игровая среда удачна, то, изменяя другие компоненты, можно создавать вариант игры с теми или иными свойствами (темпом, сложностью).

Метод организации кода компьютерной игры состоит из шести разделов.

1) **Загрузка модулей**, которые требуются в игре. Стандартные вещи, импорт локальных имен `pygame`, а также сам модуль `pygame`.

2) **Загрузка ресурсов**. Определите некоторые классы для обработки ваших основных ресурсов, которые будут загружать изображения и звуки, а также подключаться и отключаться от сетей и из сети, загружать файлы сохраненных игр и любые другие ресурсы, которые у вас могут быть.

3) **Определение классов игровых объектов**. Например, в игре пинг-понг это будет один класс для ракетки игрока (которую можно инициализировать несколько раз, по одному разу для каждого игрока) и один – для мяча (который может снова иметь несколько экземпляров). Если у вас будет хорошее игровое меню, также неплохо создать меню для класса.

4) **Определение других необходимых функций**, таких как табло, обработка меню и т. д. Любой код, который вы могли бы вставить в основную игровую логику, но который затруднил бы понимание указанной логики,

должен быть помещен в его собственную функцию. Так как построение табло не является игровой логикой, его следует перенести в функцию.

5) **Инициализация игры**, включая сами объекты `pygame`, фон, игровые объекты (инициализирующие экземпляры классов) и любые другие небольшие кусочки кода, которые вы, возможно, захотите добавить.

6) **Основной (игровой) цикл**, в который помещают обработку событий, обновление состояния игры (объектов) и обновление экрана для отображения текущего состояния игры (рис. 5.10).



Рис. 5.10. Схема взаимодействия в игровом цикле

Разделение игрового цикла на три составляющие позволяет легко вносить изменения в код, если возникают новые идеи.

Примерный шаблон программного кода для создания игры в Pygame

```

# здесь подключаются модули
import pygame
# здесь определяются константы, классы и функции
FPS = 60
# здесь происходит инициация, создание объектов и др.
pygame.init()
pygame.display.set_mode((600, 400))
clock = pygame.time.Clock()
  
```

```

# если надо до цикла отобразить объекты на экране
pygame.display.update()
# главный цикл
while True:
# задержка
clock.tick(FPS)
# цикл обработки событий
for i in pygame.event.get():
if i.type == pygame.QUIT:
exit()
# -----
# изменение объектов и многое другое
# -----
# обновление экрана
pygame.display.update()

```



Поэкспериментируйте.

Используя шаблон программного кода, создайте окно Pygame с синим фоном и движущимся на фоне солнцем.



Алгоритм подсчета результатов игры

Как нам проверить, что игрок побил свое рекордное количество очков, что у него больше нет жизненных очков? Как узнать, что у него есть ключ для того, чтобы открыть двери?

Оператор **if** – то, что нам нужно. Также он известен как *условный оператор*. **If** позволяет компьютеру принять решение. Тепло ли снаружи? Достиг ли космический корабль предела экрана? Было ли выведено со счета слишком много денег? Программа может проверять эти условия с помощью оператора **if**. Для этого используем атрибут **score**.

Пример программы:

```

if score % 1000 == 0:
if counter > 100:
counter -= 50 #
pygame.time.set_timer(MoveFigure_down, counter)

```


Пример программы:

```
#переменная, которая отвечает за общее количество баллов
topScore = 0

while True:
# Настройка начала игры.
-----
# задаем начальное значение
score = 0
-----
reverseCheat = slowCheat = False
baddieAddCounter = 0 (добавление счетчика)
pygame.mixer.music.play(-1, 0.0)

while True: # запуск игрового цикла во время игры.
    score += 1 # Увеличение счета.
```

Вывод счета

```
#Показать счет и лучший результат.
drawText('Score: %s' % (score), font, windowSurface, 10, 0)
drawText('Top Score: %s' % (topScore), font, windowSurface, 10, 40)
```



Знание и понимание



1. Что такое *игровая среда*?
2. Как вы понимаете, что такое *сценарий игры*?
3. Придумайте сценарий компьютерной игры по казахской народной сказке.
4. Из каких разделов состоит метод организации кода для компьютерной игры?
5. Что включает в себя основной игровой цикл?
6. Дан код игры «Угадай число». Проанализируйте синтаксис игры. Распишите построчно, что этот код делает.

Программный код	Описание
import random	
NumberToGuess=random.randint(1,100)	
userGuess=-1	

while userGuess!=NumberToGuess:	
userGuess=int(input(«Угадай число от 1 до 100»))	
if userGuess > NumberToGuess:	
print(«Число должно быть меньше!»)	
elif userGuess < NumberToGuess:	
print(«Число должно быть больше!»)	
else:	
print(«Вы угадали, это число = » + str(NumberToGuess))	
#Конец игры – выйти из цикла while	
break	

7. Составьте и запишите свою версию рассказа, употребив все предложенные ключевые термины по теме.



Применение

Анализ



8. Напишите по памяти шаблон программного кода для компьютерных игр в Pygame.

9. Создайте изображение игрового персонажа, который вам нравится, в формате .bmp (или преобразуйте существующее изображение). Создайте класс, который рисует персонаж в центре экрана, и приведите цвет фона изображения в соответствие с цветом фона экрана (или наоборот).

10. Создайте игру с движущимися метеоритами и ракетой. Игрок может перемещать ракету вверх, вниз, вправо и влево четырьмя клавишами со стрелками. Перемещать надо так, чтобы метеориты не попали в ракету.



Синтез

Оценивание



11. Дан код игры «Угадай число». Измените код игры, чтобы можно было подсчитать, сколько попыток сделал игрок, прежде чем угадал правильное число.

```
import random
NumberToGuess=random.randint(1,100)
userGuess=-1
while userGuess!=NumberToGuess:
```

```

userGuess=int(input(«Угадай число от 1 до 100»))
if userGuess > NumberToGuess:
print(«Число должно быть меньше!»)
elif userGuess < NumberToGuess:
print(«Число должно быть больше!»)
else:
print(«Вы угадали, это число = » + str(NumberToGuess))
#Конец игры – выйти из цикла while
break

```

12. Попробуйте адаптировать игру «Угадай число», изменив код игры.



Создание классов `Sprite` и `Group`.

В программировании игр спрайтом называют объект, который предстает перед пользователем в виде анимированного изображения и в большинстве случаев предполагает взаимодействие с ним. Другими словами, все, что в игре не является фоном, а интерактивным объектом-картинкой, – это спрайт.

Хотя каждый спрайт может быть уникальным, у всех есть нечто общее, что в ругаме вынесено в отдельный класс `Sprite`, находящийся в модуле `pygame.sprite`.

На базе этого класса следует создавать собственные классы спрайтов и уже от них объекты. Таким образом, класс `pygame.sprite.Sprite` играет роль своего рода абстрактного класса, хотя таковым не является. Можно создавать объекты непосредственно от `Sprite`.

В модуле `pygame.sprite`, кроме класса `Sprite`, есть класс `Group` и родственные ему, которые предназначены для объединения спрайтов в группы. Это позволяет вызывать один метод группы, который, например, обновит состояние всех спрайтов, входящих в эту группу.

Почти все предопределенные методы класса `pygame.sprite.Sprite` касаются добавления экземпляра в группу, удаления из нее, проверки вхождения. Только метод `update()` затрагивает поведение самого спрайта. Этот метод следует переопределить в производном от `Sprite` классе.

В конструкторе производного от `Sprite` класса необходимо вызвать конструктор родительского класса, а также обзавестись экземплярами `Surface` и `Rect`, имена которых должны быть соответственно `self.image` и `self.rect`. Так надо, чтобы с экземплярами класса могли работать методы группы. В остальном вы можете добавлять любые атрибуты.

Как создается поверхность (а также прямоугольная область), неважно. Можно делать это с помощью функции `load()`. В конструктор может передаваться уже подготовленный экземпляр `Surface`.

Класс Group решает проблему, если у вас много однотипных объектов. Добавлять спрайты в группу можно методом **add()** группы (по одному или все вместе).

У групп есть методы **update()** и **draw()**. Метод **update()** группы вызывает методы **update()** всех входящих в нее объектов. А метод **draw()** выполняет метод **blit()**. При этом в **draw()** надо передать поверхность, на которой будет происходить отрисовка.

Метод **kill()** спрайта удаляет его из всех групп, в которых он содержится. Есть метод **remove()**, который удаляет только из указанных в качестве аргумента групп. У спрайтов, так же как у групп, есть метод **add()**. Только в данном случае ему передается не объект, а группа.

Итоговые задания (проекты)

Предлагаем выбрать любое одно из трех заданий в соответствии с вашим уровнем знаний и умений, полученных при изучении разделов IV и V. Для реализации заданий выполняйте работу поэтапно. Затем соедините все части программ. В итоге получите игру, в которую сможете играть с одноклассниками или поделиться с друзьями.

Задание 1. Найдите в Интернете готовую идею игры и напишите к ней программу, применяя язык программирования Python и библиотеку PyGame.

Задание 2. Придумайте идею игры и напишите для нее программу, применяя язык программирования Python и библиотеку PyGame.

Задание 3. Напишите программу игры «Белое и черное», применяя язык программирования Python и библиотеку PyGame, а также представленные ниже правила и план игры:

- Игровое поле – плоскость, состоящая из клеток, у каждой из которых восемь соседей.

- Клетки могут находиться в двух состояниях (цветах): белом или черном. Совокупность белых клеток называется поколением.

- В процессе игры каждая клетка может неограниченное количество раз переходить из одного состояния в другое.

Правила перехода:

- Каждое следующее поколение рассчитывается на основе предыдущего.

– Черная клетка оживает, если рядом с ней находится ровно 3 белые клетки.

– Белая клетка продолжает жить, если рядом с ней находится 2 или 3 белые клетки.

План создания игры «Белое и черное» выглядит примерно так:

1. Создается двумерный массив из булевых значений, где True означает наличие белой клетки, а False – черной.

2. Генерируется новый массив.

3. Для каждой клетки определяется число белых соседей и на основании этого определяется, будет ли она жить в новом поколении.

4. Новый массив записывается на место старого.

5. Происходит возврат к пункту 2.

Чтобы не выйти за пределы массива, используйте следующее:

– Отбрасывать в функции все обращения к $i, j < 0$ и $i \geq n, j \geq m$ (где m и n – размер строки и столбца соответственно).

– Замкнуть поле на само себя – все обращения к клеткам из предыдущего пункта считать обращением к клетке с другой стороны поля. То есть $i < 0$ будет обращением к $m - i$, $i > m$ будет обращением к клетке $m - i$. Для j формулы аналогичны.

– Оградить игровое поле со всех сторон элементами, обозначающими его границу. Такие элементы физически не являются ни черными, ни белыми, но при подсчете числа белых соседей считаются черными.

Глоссарий

Актуальность – свойство информации, отражающее ее своевременность, важность и существенность для настоящего времени.

Анимация (фр. *animation* – «оживление, одушевление») – быстрая смена изображений на экране, создающая эффект движения.

База данных (БД) – организованная совокупность данных, предназначенная для хранения, изменения и обработки взаимосвязанной информации, преимущественно больших размеров.

Базовая конфигурация компьютера – минимальный комплект аппаратных и программных средств, достаточный для начала работы на компьютере.

Библиотека PyGame – библиотека модулей для языка программирования Python, облегчающая разработку 2D-игр.

Возникновение «системного эффекта» – главное свойство любой системы. При объединении элементов у системы появляются новые свойства, которыми не обладал ни один ее элемент.

Достоверность – свойство информации, отражающее ее истинность, соответствие действительности, фактам.

Загрузка операционной системы (ОС) – самая важная функция, выполняемая BIOS.

Записи базы данных в электронной таблице Excel – строки.

Запись – строки таблицы, в которых записывается и хранится информация. В таблице может быть любое количество записей (ограничение – объем носителя) или ни одной записи (пустая таблица).

Игровая среда – совокупность всех объектов и связей в игре и законов их изменения.

Информационная модель – модель, в табличной форме детально описывающая объекты, выявленные при формализации задачи, их параметры, действия.

«Как сделать, чтобы...» – тип задач, в которых требуется определить, какое надо произвести воздействие на объект, чтобы его параметры удовлетворяли заданному условию.

Конфигурация персонального компьютера – набор устройств и программного обеспечения, характеристики которых определяют основное направление использования персонального компьютера.

Массив – структура данных, представленная в виде группы ячеек одного типа, объединенных под общим именем.

Матрицы, или двумерные массивы (two-dimensional array) – данные, хранящиеся в прямоугольной таблице.

Облачные технологии (вычисления) (англ. *cloud computing*) – вычислительный ресурс для электронного хранения, обработки и совместной работы с данными в сети Интернет.

Одномерный массив – массив с одним параметром, характеризующим количество его элементов.

Перестановки – выборки элементов, отличающиеся только порядком расположения элементов, но не самими элементами.

Поиск – обработка некоторого множества данных с целью выявления подмножества данных, соответствующего критериям поиска.

Поля – столбцы таблицы, образующие структуру базы данных. Для каждого поля задается имя поля, тип данных, перечень свойств, описание.

Поля базы данных в электронной таблице Excel – столбцы.

Сбалансированная конфигурация компьютера – конфигурация, при которой все возможности его процессора, памяти, видеокарты, шин для обмена информацией дают возможность обрабатывать данные максимально быстро, минуя малейшие простои, для каждого типа решаемых задач.

Сетевой этикет – (англ. *net* – «сеть» и фр. *etiquette* – «этикет») – правила поведения, общения в сети, традиции и культуры интернет-сообщества, которых придерживается большинство пользователей.

Система – (греч. – «целое, составленное из частей; соединение») – совокупность элементов, находящихся в отношениях и связях друг с другом, образующая определенную целостность, единство.

Систематизация (классификация) – процесс превращения множества объектов в систему.

Событие – действие, выполняемое пользователем во время игры. Например, при нажатии клавиш клавиатуры или мыши событие создается, размещается в очереди, дожидается обработки.

Сортировка – упорядочение (перестановка) элементов в подмножестве данных по какому-либо критерию. Чаще всего в качестве критерия используется некоторое числовое поле, называемое ключевым.

Столбцы электронной таблицы Excel – поля базы данных.

Строки электронной таблицы Excel – записи базы данных.

Список (list) – один из типов последовательностей в языке Python. В последовательностях хранятся упорядоченные наборы объектов, объединенные под общим именем.

Структура системы – определенный порядок объединения элементов системы.

Сценарий – форма, в которой выражается сюжет.

Сюжет – последовательность событий, которые складываются в историю.

Таблица базы данных – обычная двухмерная таблица Excel, созданная с соблюдением определенных правил.

Точность – свойство информации, определяющееся степенью ее ясности, близости к реальному состоянию объекта, процесса, явления.

Упорядоченность – наличие отсортированного ключевого поля.

Ценность – свойство, зависящее от полезности конкретных объектов и задач, которые можно решать с помощью полученной информации.

Частота кадров (частота обновления экрана) – количество изображений, которые программа рисует в секунду, измеряется в FPS (Frames Per Second) кадрах в секунду.

«Что будет, если...» – тип задач, в которых требуется исследовать, как изменятся характеристики объекта при некотором воздействии на него.

Элементы массива – переменные, входящие в состав массива.

BIOS (Basic Input/Output System) – набор микропрограмм, которые отвечают за управление всеми компонентами, установленными на материнской плате и подключенными к компьютеру устройствами.

POST (power-on self-test) – тестирование всего установленного на материнской плате оборудования (за исключением дополнительных плат расширения), проводимое после каждого включения компьютера.

Рекомендуемая литература

1. Закон Республики Казахстан об информатизации (новая редакция). Постановление Правительства Республики Казахстан от 31 мая 2014 года № 594.
2. Закон Республики Казахстан от 16 ноября 2015 года № 401-у «О доступе к информации» (с изменениями и дополнениями по состоянию на 28.12.2016 г.).
3. *Златопольский Д. М.* 1700 заданий по Microsoft Excel. СПб: БХВ-Петербург, 2003.
4. *Молодцов В. А., Рыжикова Н. Б., Головки Т. Г.* Репетитор по информатике. Ростов-на-Дону, 2004.
5. *Мухамбетжанова С. Т., Тен А. С., Даутова Т. К., Сагымбаева А. Е.* Информатика: учебник для 7 класса общеобразовательных школ. Алматы: Атамұра, 2015.
6. Основы компьютерных сетей. Учебное пособие. Алматы: Microsoft Corporation, 2008.
7. Персональный компьютер: настройка и техническая поддержка. Дополнительные материалы к методическому пособию. Microsoft Corporation. Партнерство в образовании. 2008.
8. *Попов А.* Excel: Практическое руководство. М., 2000.
9. *Сафронов И. К.* Задачник-практикум по информатике. СПб: БХВ-Петербург, 2002.
10. Сборник «Инструментарий международного исследования PISA-2018/ компьютерный формат». Астана: АО «Информационно-аналитический центр», 2016. 126 с.
11. *Семакин И. Г., Вараскин Г. С.* Структурированный конспект базового курса. Приложение к учебнику: Информатика. Базовый курс 7–9 классы. М.: Лаборатория базовых знаний, 2001.
12. *Федоров Д. Ю.* Основы программирования на примере языка Python. Учебное пособие. СПб, 2019.
13. *Буйначев С. К., Боклаг Н. Ю.* Основы программирования на языке Python. Учебное пособие. Екатеринбург, 2014.
14. *Бриггс Дж.* Python для детей. Самоучитель по программированию. М., 2017.
15. *Лутц М.* Изучаем Python. СПб – М., 2009.
16. *Вордерман К.* Компьютерное программирование. Алматы, 2015.

Рекомендуемые интернет-ресурсы

enet.kz.
<http://assembly.kz>.
<http://cppstudio.com/cat/285/>.
<http://daryn.kz/>.
<http://elar.uspu.ru/bitstream/uspu/8249/2/22Oganyan.pdf>
<http://fb.ru/article/187288/po-kakim-kriteriyam-vyibrat-po-programmnoe-obespechenie-pk>
<http://on-line-teaching.com/excel/index.html>.
<http://programm.ws/kat.php?id=14>.
<http://purecodecpp.com/archives/1>
<http://school-collection.edu.ru>.
<http://schools.keldysh.ru/sch444/MUSEUM/>.
http://umk-model.narod.ru/newfile_41.html
<http://www.c-cpp.ru/>.
<http://www.computer-museum.ru/index.php>.
<http://www.edusite.ru>.
<http://www.elektramoto.ru/1/15.html>
<http://www.informatik.kz/>.
http://www.lessons-tva.info/edu/e-inf2/m1t1_4.html.
http://www.lessons-tva.info/edu/e-inf2/m2t2_1.html.
<http://www.openclass.ru/>.
<http://www.pulser.kz>
<http://www.sc109.ru/content/distant/inform/9/modelirovanie/7.htm>
<http://www.videoyroki.info/>.
<http://ypk.yspu.yar.ru/tolerance/3.htm>.
<https://bilimland.kz/ru>.
<https://habr.com/post/329758/>
<https://idaten.ru/technology/primenenie-baz-dannih-v-sovremennom-mire>
<https://natworld.info/vopros-otvet/kakaja-chast-sveta-naibolshaja-ploshhadi-i-chislennosti-naselenija> © Природа Мира|NatWorld.info
<https://nsportal.ru/npo-spo/informatika-i-vychislitelnaya-tehnika/library/2015/05/03/vybor-konfiguratsii-pk>
<https://ppt-online.org/111263>
<https://rcdo.kz/>.
<https://tproger.ru/digest/competitive-programming-practice/>.
<https://www.facebook.com/redmarkfilm/videos/235341703688976/>

[https://www.osp.ru/school/archive.](https://www.osp.ru/school/archive)

[it-n.ru.](http://it-n.ru)

[metod-kopilka.ru.](http://metod-kopilka.ru)

physics-cell.xls

www.itmathrepetitor.ru

<https://bilimland.kz/ru/courses/informatika-ru/9-klass/>

https://online.zakon.kz/Document/?doc_id=39797352#pos=2;-274

Закон Республики Казахстан «О защите детей от информации, причиняющей вред их здоровью и развитию».

Дистрибутивы для установки:

<https://www.python.org/ftp/python/3.7.0/python-3.7.0a1.exe> – Windows, 32 бита.

<https://www.python.org/ftp/python/3.7.0/python-3.7.0a1-amd64.exe> – Windows, 64 бита.

<https://www.python.org/downloads/release/python-362/> – Mac OS X.

Установка интегрированной среды разработки WING IDE 101:

<http://wingware.com/pub/wingide-101/5.0.9/wingide-101-5.0.9-1.exe> – дистрибутив для Windows.

<https://www.youtube.com/watch?v=OrpavIGbkSw> – видео установки Python и Wing IDE 101 на Windows 7 (на англ.).

<https://pythondigest.ru/>

<https://checkio.org> – игра для обучения программированию.

<https://www.hackerrank.com> – задания по Python на основе рейтинга.

Приложение

Дополнительный материал к разделу III «Базы данных»

Базы данных классифицируются по трем основным признакам: *по характеру хранимой информации* (рис. 1), *по способу хранения*, *по структуре организации*.



Рис. 1. По характеру хранимой информации

Классификация по способу хранения данных делит базы данных на *централизованные* и *распределенные* (рис. 2).



Рис. 2. По способу хранения информации



Рис. 3. Типы данных

Многие годы реляционная модель считалась единственным способом создания баз данных при организации их хранения. Сам по себе реляционный подход с применением языка запросов SQL (на английском языке *Structured Query Language* – «язык структурированных запросов») оказался простым и удобным.

С появлением Big Data (Большие Данные), облачных технологий, Интернета вещей (концепция вычислительной сети физических предметов «вещей»), всеобщей мобилизации начала проявляться ограниченность такого подхода.

В настоящее время развитие технологий баз данных связано с появлением принципиально новых прикладных задач, объединенных с аналитикой, социальными сетями и т. д.

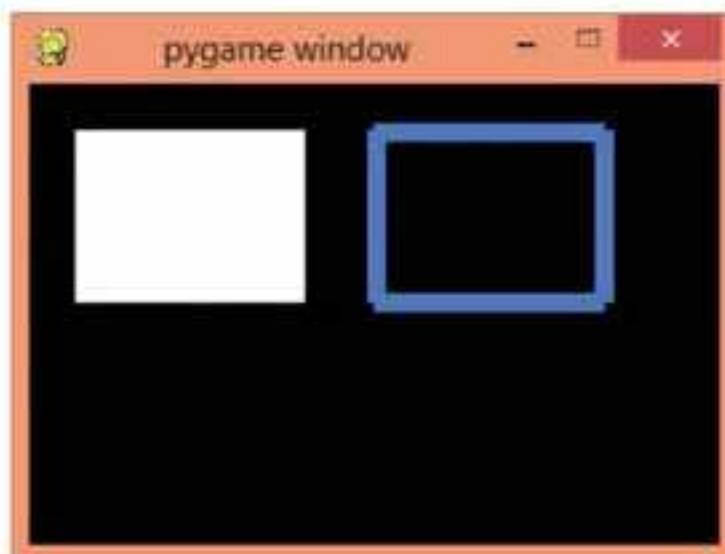
Появление большого количества различных систем класса NoSQL (на английском языке *not only SQL* – «не только SQL») обозначает ряд подходов, направленных на реализацию хранилищ баз данных, которые существенно отличаются от моделей, используемых в традиционных реляционных СУБД.

NewSQL (на английском языке *NewSQL* – «новый SQL») сочетает в себе достоинства реляционного и нереляционного подходов.

Дополнительный материал к разделу V

Для создания своего персонажа в PyGame можно использовать модуль `pygame.draw`, содержащий различные функции. С помощью этих функций можно рисовать персонажи непосредственно на поверхности. Например, на рисунке изображены два объекта – прямоугольники.

```
import pygame
pygame.init()
sc = pygame.display.set_mode((300, 200))
# Прямоугольники
pygame.draw.rect(sc, (255, 255, 255),
(20, 20, 100, 75))
pygame.draw.rect(sc, (64, 128, 255),
(150, 20, 100, 75), 8)
pygame.display.update()
while 1:
    pygame.time.delay(1000)
    for i in pygame.event.get():
        if i.type == pygame.QUIT: exit()
```



Прямоугольники

Если указывается толщина контура `pygame.draw.rect (sc, (64, 128, 255), (150, 20, 100, 75), 8)` – последний аргумент, то прямоугольник будет незаполненным. Кортеж из четырех чисел определяет координаты верхнего левого угла прямоугольника, а следующие два числа – ширину и высоту.

Основные функции Draw для рисования фигур

Функция	Назначение	Примеры
<code>draw.rect</code>	1. Прямоугольная форма 2. Закрашенная форма прямоугольника	1. <code>pygame.draw.rect(sc, (white), (30, 30, 100, 70))</code> 2. <code>pygame.draw.rect(sc, (green), (150, 20, 100, 75), 8)</code>
<code>draw.polygon</code>	Фигура с любым количеством сторон	<code>pygame.draw.polygon(sc, WHITE, [[150, 10], [180, 50], [90, 90], [30, 30]])</code>
<code>draw.line</code>	Отрезок прямой линии	<code>pygame.draw.line(sc, WHITE, [10, 30], [290, 15], 3)</code>
<code>draw.circle</code>	1. Круг вокруг точки 2. Закрашенная форма круга	1. <code>pygame.draw.circle(sc, YELLOW, (100, 100), 50)</code> 2. <code>pygame.draw.circle(sc, PINK, (200, 100), 50, 10)</code>

```

# Импорт PyGame
import pygame
# Определение констант
FPS = 60
WIDTH = 500
HEIGHT = 100
WHITE = (255, 255, 255)
ORANGE = (255, 150, 100)
# Инициализация PyGame
pygame.init()
clock = pygame.time.Clock()
# Создание отображаемой области для
прорисовки графических объектов игры
sc = pygame.display.set_mode((WIDTH,
HEIGHT)) # Размеры игрового поля
# Установка радиуса круга
r = 30
# Координаты круга x, y
x = 0-r # Исчезает круг за левой
границей окна
y = HEIGHT // 2 # Выравнивание круга по
центру и вертикали
# Цикл управляет всем процессом игры: до
обнаружения и обработки событий (event)
while 1:
    sc.fill(WHITE)
    for i in pygame.event.get():
        if i.type == pygame.QUIT: exit() #
Обнаружение события QUIT – выход из программы
        pygame.draw.circle(sc, ORANGE, (x, y), r)
        pygame.display.update()
        # Проверка условия: если полностью
скрылся за правой границей
        if x >= WIDTH + r:
            x = 0-r
        else: # Иначе в остальных случаях
            x += 2 # Круг сместится в следующем кадре
    clock.tick(FPS) # Скорость движения зависит от значения FPS

```



Перемещение круга

При перемещении объекта на экране вызов `pygame.display.update()` будет постоянно обновлять экран, отображая игровые объекты в новых позициях и скрывая старые изображения.

Таким образом, создается иллюзия плавного движения или анимации. Приведенный пример показывает именно такую анимацию объекта, т. е. он двигается с левой стороны, достигает правой, а затем исчезает. После этого снова появляется слева.

Учебное издание

**Мухамбетжанова Сауле Талапеденовна
Тен Алира Сунтаковна
Голикова Нелля Генриховна**

ИНФОРМАТИКА

Учебник для 9 класса общеобразовательной школы

Зав. редакцией *Н. Жиенгалиев*

Редактор *А. Раймкулова*

Художественные редакторы *З. Аульбекова, А. Лукманов*

Технический редактор *О. Рысалиева*

Корректор *Г. Туленова*

Компьютерная верстка *А. Бекбергеновой, Е. Козловой*

ИБ № 120

Сдано в набор 26.02.2019. Подписано в печать 05.07.2019. Формат 70x90^{1/16}.
Бумага офсетная. Гарнитура «Школьная». Печать офсетная. Усл.печ.л. 11,7.
Уч.-изд.л. 6,64. Тираж 12 000 экз. Заказ №4367.

ТОО «Корпорация «Атамұра», 050000, г. Алматы, пр. Абылай хана, 75.

Полиграфкомбинат ТОО «Корпорация «Атамұра», Республика Казахстан,
050002, г. Алматы, ул. М. Макатаева, 41



