

Н.С. Кольева, Е.В. Шевчук, Э.Б. Ержанов

ИНФОРМАТИКА

Жалпы білім беретін мектептің
9-сыныбына арналған оқулық

*Қазақстан Республикасының
Білім және ғылым министрлігі бекіткен*



Алматы "Мектеп" 2019

ӘОЖ 373.167.1
КБЖ 32.973я72
К51

Кольева Н.С. т.б.
К51 **Информатика.** Жалпы білім беретін мектептің 9-сыныбына арналған оқулық. — Алматы: Мектеп, 2019 — 104 б.

ISBN 978—601—07—1262—1

К $\frac{4306020500-087}{404(05)-19}$ 12(1)—19

ӘОЖ 373.167.1
КБЖ 32.973я72

© Кольева Н.С., Шевчук Е.В.,
Ержанов Э.Б., 2019
© “Мектеп” баспасы,
көркем безендірілуі, 2019
Барлық құқықтары қорғалған
Басылымның мүлкітік құқықтары
“Мектеп” баспасына тиесілі

ISBN 978—601—07—1262—1

КІРІСПЕ

Қазір қоғамда ақпараттың шамадан тыс көбейген тұсы. Компьютерлік техника автоматты немесе автоматтандырылған құрылғыларды қолданып, аса күрделі амалдарды адамдардың ақпаратты өңдеу жылдамдығынан миллион есе жылдам орындауға мүмкіндік береді.

Бүгінгі маман басқаруда бағытты қадам жасайтын жоғары ақпараттанған қоғамның жаңа шарттарымен жұмыс істеуге дайын болуы керек. Ол информатика саласын, компьютерлік техниканы қолданудың практикалық дағдысын, қазіргі ақпараттық технологияларды, ақпараттық жүйелермен жұмыс істеудің негізгі принциптерін, ақпараттың толықтығын және дәлдігін бағалай білуі керек. Бұған информатика оқулығы өз көмегін тигізеді.

Оқулықта берілген шартты белгілер қажетті ақпаратты меңгеруге көмектеседі.

Авторлар

Шартты белгілер:



— мұны білулерің қажет



— өзін-өзі тексеруге арналған сұрақтар



— бақылау сұрақтары мен тапсырмалары



— компьютерде орындайтын практикалық жұмыстар

А деңгейі

— оқу-танымдық тапсырмалар

В деңгейі

— танымдық-зерттеу тапсырмалары

С деңгейі

— шығармашылық тапсырмалар

1- ТАРАУ

Ақпаратпен жұмыс

Бұл тараудан білесіңдер:

- ▶ ақпараттың қасиеттерін;
- ▶ Google-дің бұлттық технологиясын, бірлескен электрондық құжаттармен жұмыс істеуге арналған дискіні;
- ▶ аккаунттың не екенін.

Мынаны үйренесіңдер:

- ▶ ақпараттың қасиеттерін анықтауды;
- ▶ Google-де аккаунт құруды;
- ▶ Google сервисін пайдаланып, ортақ ұжымдық презентацияларда слайд жасауды;
- ▶ бірлескен электрондық құжаттармен жұмыс жасауды;
- ▶ Google-де бірлескен презентацияларға қатысу параметрлерін басқаруды;
- ▶ желідегі этикалық және құқықтық нормалардың бұзылу салдарын талдауды.

§1 Ақпараттың сипаттары

Бүгінгі сабақта:

- ▶ ақпараттың қасиеттерін анықтауды (өзектілігін, дәлдігін, нақтылығын, құндылығын);
- ▶ компьютерде ұзақ уақыт жұмыс істеуді сыни түрде бағалауды үйренесіңдер.

Тірек ұғымдар:

- ▶ ақпарат;
- ▶ ақпараттың жинақтылығы;
- ▶ ақпараттың құндылығы;
- ▶ ақпараттың өзектілігі;
- ▶ ақпараттың дәлдігі;
- ▶ ақпараттың нақтылығы;
- ▶ ақпараттың түсініктілігі;
- ▶ репрезентативтік;
- ▶ қолжетімділік.

Қоғамның экономикалық дамуы, адамдардың өмірі мен денсаулығы қазіргі ақпараттың қасиеттеріне тәуелді. Кез келген жағдайда алынған ақпарат қаншалықты түсінікті, өзекті, пайдалы және мәліметтер шынайы екенін бағалау үшін ақпараттың қасиеттерін талдайық.



Сендерге кез келген ақпарат көмектесе ала ма? Неге кейбір адамдар қандай да бір ақпаратқа бірден көңіл аударады, ал кейбіреулері бұл ақпаратқа немқұрайлы қарайды?

Ақпарат қасиеттерінің маңыздылығы нақты бір жағдайда анықталады. Адамдар бір ақпаратты әртүрлі қабылдайды. Мысалы, теледидардан шетел валютасының бағамын көрсеткен кезде көп адамдар бұл ақпаратқа көңіл аудармайды, ал кейбір адамдар валюта айырбастау үшін асығып банкке барады. Демек, ақпарат әртүрлі қасиетке ие болғандықтан, кез келген адам оны түрліше қабылдайды.

Кейбір жағдайларда ақпараттың өзектілігі, анықтығы, қолжетімділігі және түсініктілігі сияқты қасиеттері маңызды (мысалы, шет тілін білмеген балаға сол тілде жазылған кітаптар түсініксіз болады).

Ақпаратты қолданудың тиімділігі өзектілік, қолжетімділік, нақтылық, репрезентативтік, толықтық, жеткіліктілік сияқты қасиеттермен байланысты.

Ақпарат бір адам үшін құнды болса, ал басқа адам үшін ешқандай мағынасы болмауы мүмкін. Мысалы, басқа қаладағы ауа райы туралы ақпарат көп адамдарға қажет емес, бірақ сол қалаға іссапармен баратын адамдар үшін бұл ақпарат құнды болып табылады. **Құндылық** (пайдалылық) — *ақпарат қасиеттерінің бірі.*

Ақпараттың тағы маңызды қасиеттерінің бірі — **өзектілігі**. Ақпараттың өзектілігі адам немесе қоғам үшін бұл мәліметтер қаншалықты маңызды екендігімен және мәселені шешуге қажетті нақты жағдайда қолданылуымен анықталады. Бұл қасиет көбінесе ақпараттың пайда болған кезінен бастап өткен уақыт аралығында, сонымен қатар жағдай қаншалықты тез өзгеретініне тәуелді.

Тек уақытында алынған өзекті ақпарат қана адамға маңызды болып табылады. Мәселен, ауа райы болжамын да бір күн бұрын хабарлайды.

Ақпараттың **қолжетімділігі** оның түсінікті және нақты түрлендірілуін қамтамасыз етеді. Қабылдаушыға бір ақпаратты әртүрлі түрде ұсынуға болады.

Қабылдаушыға ақпаратты түсінікті түрде қажет және түсінікті тілде көрсету .

Мысалы, 10-сыныпқа арналған физика оқулығындағы ақпарат 8-сынып оқушысына мүлдем түсініксіз, себебі, оқулықта таныс емес ұғымдар мен формулалар бар. Керісінше 8-сыныпқа арналған физика оқулығы сол сынып оқушыларына түсінікті ақпаратты қамтиды, бірақ жоғары сынып оқушысы осы оқулықтан ешқандай жаңалық таба алмайды.

Кітап дүкендеріндегі балалар әдебиеті бөлімінде әрбір кітапта қай жас аралығындағы оқырманға арналғаны көрсетілген. Бұл ақпарат сол көрсетілген жастағы оқырманға қолжетімді және түсінікті екенін білдіреді.

Қазіргі уақытта кітапханаларға енгізілген *кітапханалық каталог* ақпаратты іздеу жүйесін жеңілдетеді. Ол сұралған тақырып бойынша оқулық туралы ақпаратты оқырманға қолжетімді және ыңғайлы түрде беретіндей етіп жобаланған. Ғалымдар түрлі табиғи апаттар кезінде алдын ала ескертудің жылдам және сенімді тәсілдерін осы ережеге сәйкес іздестіруде.

Ақпараттың анықтылығы нақты нысанның, процесстің немесе оқиғаның жағдайын қамтып көрсететін қасиетімен анықталады . Бұрыс ақпарат жағдайды теріс түсінуге әкелуі мүмкін, оның салдарынан дұрыс шешім қабылданбайды.

Ақпараттың көптігі дұрыс шешім қабылдау үшін мәліметтің жеткілікті деректер жинағын қамтитынын білдіреді . Нысан туралы қандай да бір қосымша ақпарат көп болса, онда ақпараттың қолжетімділігі туралы айтуға болады.

Ақпараттың көптігі оның мағыналық мазмұнымен байланысты. Ақпарат жеткілікті болмаса, ол шешім қабылдаудың тиімділігін төмендетеді.

Сонымен, ақпарат *өзекті* , *қолжетімді* , *нақты* және *көп болуы* қажет.

Ақпараттың жеткіліктілігі — ол алынған ақпараттың көмегімен жасалынған бейненің нақты нысанға, процеске немесе оқиғаға сәйкестігі . Нақты өмірде ақпараттың толық жеткілікті болуы екіталай. Нысанның нақты күйіне ақпараттың сәйкес болу дәрежесі адамның қабылдайтын шешіміне әсер етуі мүмкін.

Мысалы, сендер мектепті табысты аяқтап, білімдерінді экономикалық бағытта жалғастыруды қалайсындар. Жолдастарыңмен ақылдасып, мұндай дайындықты әртүрлі ЖОО-да алуға болатынын білесіндер. Бір-біріне қайшы келген әртүрлі мәліметтер, алынған ақпараттар, істің нақты ақпаратына сәйкес келмеуі мүмкін. Себебі

накты ақпарат алу үшін ЖОО-ның ресми сайты немесе экономикалық мамандықтардың рейтингісін көрсететін түрлі сайттарда берілген нақты ақпаратқа сүйенулерің қажет.

БҮЛ ҚИЗЫҚ!

Википедияда берілген ақпарат қаншалықты шынайы? Егер Google-ге “Википедиядағы ақпарат қаншалықты дұрыс?” деген сұрақ қойсақ, онда осы тақырып бойынша Википедияның келесі мақаласына сілтеме ашылады. (“Википедияның сенімділігі”). 2005 жылы ғалымдар Википедиядағы ғылыми мақалалардың сапасын тексерді. Нәтижесінде, мамандар сарапшылардың талдау жасаған 42 мақалалары арасынан көптеген кәсіптіктер анықтайды.

Википедия ақпараттың нақтылығын тексеретін басқа да арнайы энциклопедиялармен салыстырғанда жүргізілген зерттеулердің нәтижесі қанағаттанарлық болды.

Дегенмен де әрбір адам ақпаратқа түзету енгізе алатын болғандықтан, берілген деректер басқа бұрыс мәліметпен өзгеруі де мүмкін.



Ақпараттың репрезентативтігі ақпаратты таңдау дұрыстығымен және нысанның қасиеттерін дәлме-дәл бейнелеу арқылы ол ақпаратты қалыптастыруға байланысты. Ақпараттың репрезентативтік қасиетін анықтаудың міндетті шарты — әртүрлі дереккөздерден ұқсас ақпаратты іздеу. Барлық дереккөздерден келіп түскен ақпарат бірдей болмайтыны түсінікті. Бірақ егер барлығы дұрыс болса, онда алынған ақпарат нысанның маңызды сипаттамаларын көрсетеді.

Мұнда ең бастысы — берілген әдістемелер мен ақпаратты таңдау мен бағалау әдістері.

Мәселен, қаланың әлеуметтік қызметкерлері апта сайын әрбір отбасы орташа азық-түлікке қанша ақша жұмсайтынын анықтауы қажет делік. Осы мәселені шешу үшін әлеуметтік қызметкерлер әрбір қаланың тұрғындарынан сұрақ-жауап алу арқылы оны анықтайды деу мүмкін емес. Демек, сұрастырылатын адамдар тобын таңдау қажет. Сұраудың нәтижесінде *іріктеу* деп аталатын ақпарат массиві қалыптасады. Сонымен қатар, сұрастыру әдісін, жиналған деректерді өңдеу, бағалау және талдау әдістерін анықтау керек. Алынған нәтижелер қаланың көп тұрғындарына

тән жағдайды көрсететін болса, онда таңдаған адамдар тобын сұрастыру нәтижесінде алынған ақпараттың репрезентативтігі туралы айтады. Ақпараттың жеткіліктілігі мен репрезентативтігі туралы қорытындылар *статистика және математикалық статистика* деп аталатын ғылым саласы айналысатын арнайы әдістердің негізінде жасалады.

Сонымен қатар ақпарат *объективті* немесе *субъективті* болуы мүмкін (біреудің ой-пікіріне тәуелді немесе тәуелсіз). “Еуропада қыс мезгілі суық” хабарламасы субъективті, себебі еуропалық қыс мезгілі біреулер үшін суық (мысалы, Оңтүстік Қазақстан тұрғынына), ал кейбіреулер үшін (мысалы, Солтүстік Қазақстан тұрғындарына) жылы болып көрінуі мүмкін. “Ауа температурасы -20°C ” деген хабарламаны әрбір адам өзіне нақты қорытынды жасайтын объективті ақпарат ретінде қабылдайды.



Бірдей хабарланған ақпарат түрлі адамдар үшін әртүрлі қасиеттерге ие болуы мүмкін.

Ең маңызды қасиеттері нақты жағдай үшін ерекшеленген бірнеше мысал қарастырайық.

1-мысал. Танертен мектепке жиналған кезде міндетті түрде сағатқа қарайсындар: ол объективті ақпарат көзі. Қандай киім кию керек екенін анықтау үшін терезеге немесе термометрге қарайсындар. Мұнда ақпараттың өзектілігі маңызды. Содан кейін сабақ кестесіне сәйкес сабақ болатын кабинетті іздеп табасындар. Бұл толық және нақты ақпарат, ол болмаса қажетті кабинетті таба алмай қалуларың мүмкін.

2-мысал. Сендерге сапардың бағдарын анықтау, жаңа елмен танысу, тарихи оқиғаларды зерттеу үшін географиялық картаны қолдану керек. Картаны адамдар әрқашан бағдар жасау үшін ақпарат көзі ретінде пайдаланады. Нақты аймақпен сәйкестендіру және құрылыс жұмыстарын үйлестіру сияқты міндеттер картаның көмегімен шешіледі. Сондықтан мұнда нақты аймақтың картасы қамтитын ақпараттың жеткіліктілігі маңызды.

3-мысал. Қазіргі уақытта компьютер немесе мобильді құрылғылардағы карталар — геоақпараттық жүйелер көпшілікке мәлім. Оларда жолсеріктерден келетін ақпараттар өңделеді және талданады. Мұндай жүйелер дәстүрлі емес міндеттерді шешуге көмектеседі:

— сату көлемі мен нарық әлеуетін жобалайды, сондықтан демографиялық деректер дүкендер мен тауар сұрыпталымы туралы ақпаратты көрсете алады;

— экологиялық апаттардың салдарын жою үшін тиімді шешімдерді талдайды;

— гидрографиялық желі модельдерін құру және су тасқыны болуы мүмкін аймақтарды анықтайды;

— Жер бедерінің моделін құрастырады.

Барлық карталар тек мамандар түсінетін арнайы тілмен “жазылған”. Бұл ақпаратты барлық адамдар қабылдап, пайдалана алмауы мүмкін. Әрбір таңба сол маман үшін нақты, объективті және түсінікті ақпаратты қамтиды.

4-мысал. Заманауи “ғарыштық технологияларда” әртүрлі құралдардан алынған ақпарат шешуші рөл атқарады. Мысалы, ғарыш станциясының Күнге қатысты орналасуы оның Күн батареясының жұмысына тәуелділігінде. Кішігірім қателік болса, ғарыш кемесінде қуат көзі азайып, апатқа алып келуі мүмкін. Мұндай ақпарат өзекті, нақты және толық болуы керек.

Бақылау сұрақтары



1. Әрбір адам үшін ақпарат дегеніміз не?
2. Ақпараттың қасиеттерін атаңдар. Оған сипаттама беріңдер.
3. Көп және аз, анық және анықталмаған ақпаратқа мысал келтіріңдер.
4. Бір ақпараттың адамдар үшін әртүрлі қасиетке ие екендігіне және оны көпшілік бірдей қабылдамайтынына мысал келтіріңдер.

Практикалық жұмыс

А деңгейі

Келесі мысалдарда кездесетін ақпараттардың қасиеттерін анықтандар:

— Келесі күні мұғалім барлық есептердің жауаптарын іліп қойды.

— Математика пәнінен бақылау жұмысы жүруде. Сен көршіңнен есептің шешімін сұрадың. Ол берген қағазда толық және дұрыс шешім жазылған, бірақ жазуы түсініксіз.

— Көрші елді жаулап алардың алдында парсы патшасы бақсыға келіп: “Мен өзімнің әскерлеріммен шегаралық өзеннен өтсем не болады?” — деген сұрақ қойды. Бақсы: “Тақсыр, сен ұлы патшалықты жоясың”, — деді. Мұндай болжамнан кейін

патша өзінің әскерімен өзен арқылы жорыққа шықты да жаудың әскерінен жеңіліс тапты. Қатты ашуланған патша бақсыға өтірік айттың деп қайта барды. Сонда бақсы: “Тақсыр, сенің патшалығың ұлы болмады ма?” — деп жауап берді.

В деңгейі

Мысал келтіріңдер:

- 1) анық, бірақ өзекті емес ақпарат;
- 2) өзекті, бірақ анық емес ақпарат;
- 3) толық, анық, бірақ пайдалы емес ақпарат;
- 4) өзекті, бірақ түсінікті емес ақпарат;
- 5) өзекті, бірақ репрезентативті емес ақпарат.

С деңгейі

Халықаралық ғылыми конференция материалдары жинағынан мақаланы оқыңдар. Мәтіннің жазылу стилін сақтап, оның мазмұнын өзгертіндер.

“Көп ядролы процессорлардың дамуы жабдықтың бірдей өлшемімен өнімділікті жоғарылатуына, құрал-жабдықтың өзіндік құнын азайтуға және операциялық шығындардың нәтижесінде бұлтты жүйені қолданудың төмендеуіне әкеліп соқтырды. Бұл көптеген Деректерді өңдеу орталықтары үшін (ДӨО) қуаттылықты арттыру кезінде үлкен қиындық тудырды. Ақпаратты тасымалдау көлемінің ұлғаюы 1 Мб ақпарат сақтау құнының төмендеуіне және сақталатын ақпарат көлемінің шектеусіз ұлғаюына алып келді. Ақпараттық жүйеде сақталған деректер көлемінің көбеюі басқа да ақпараттарды сақтауды қамтамасыз етуін төмендетті. Көп ағымды бағдарламалау технологиясының дамуы көппроцессорлық жүйелердің есептік ресурстарын — “бұлтты” есептеулердің қуатын біркелкі тиімді пайдалануға әкелді. Виртуалды технологияның дамуы интернет желісі арқылы виртуалды инфрақұрылымның болуын желілік өткізу қабілеті деректер алмасу жылдамдығын, интернет-трафиктің құнын төмендетуге және бұлтты технологияларға қол жеткізуге мүмкіндік берді. Осы факторлардың барлығы Ақпараттық технологиялар саласында бұлтты технологиялардың бәсекеге қабілеттілігін арттыруға алып келді”.

Широкова Е.А. Бұлтты технология // Техника ғылымдарының қазіргі заманғы үрдістері.

Рефлексия:

Менің білгенім... . Енді мен ... қолдана аламын.

§2-3 Құжаттармен бірігіп жұмыс істеу

Бүгінгі сабақта:

- ▶ бұлттық технологияларды қолдана отырып, құжаттармен бірлесе жұмыс істеуді ұйымдастыруды үйренесіңдер.

Тірек ұғымдар:

- ▶ бұлттық технологиялар;
- ▶ құжаттармен бірлескен жұмыс.

? *Бүкіл сыныптан бірдей уақытта бір құжатпен жұмыс істеуге бола ма? Мысалы, бірі мәтінді өңдейді, екіншісі кестелер қосады, ал үшіншісі бұл уақытта суреттермен жұмыс істеуі мүмкін. Нәтижесінде азғана уақыттың ішінде суреттері және сызбалары дайын құжатты алу мүмкін бе?*

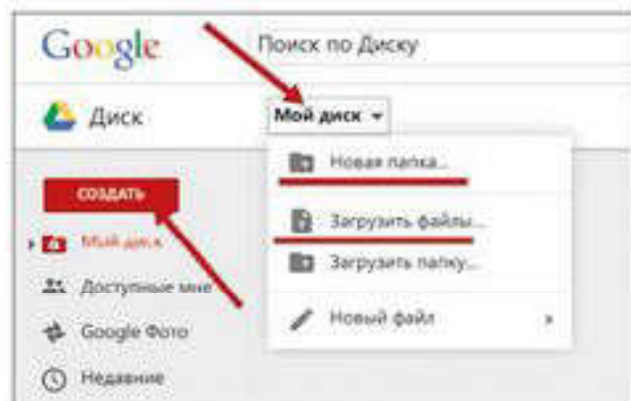
Бұл параграфта Google-де құжаттармен жұмыс істеу принциптерін, мысалы, мәтіндерді, кестелерді біріктіріп қалай өңдеуге болатынын, сондай-ақ жалпы файлдарды оқу мүмкіндігін қалай қамтамасыз ету керектігін қарастырамыз.

Google-дің бұлтты сақтау қызметі дискісі интернетте 15 Гб-қа дейін ақпаратты сақтауға мүмкіндік береді (2.1-сурет). Google дискісімен жұмыс істеу үшін Google аккаунтын құру керек. Егер

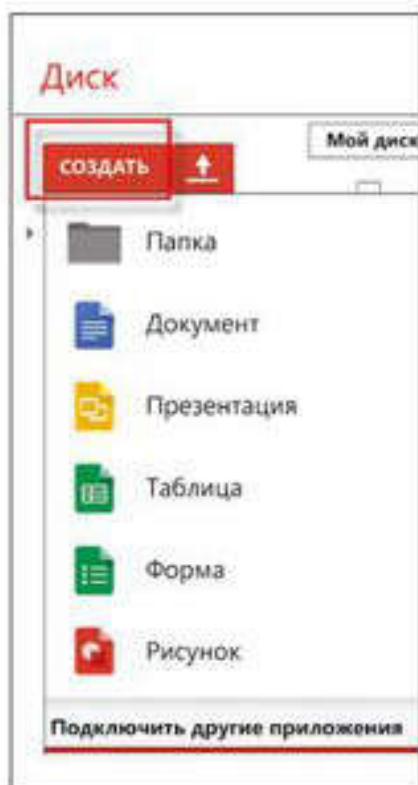


2.1-сурет

gmail.com пошталық жәшікті құрсандар, аккаунт автоматты түрде орнатылады. Кез келген пошталық серверде аккаунт құруға болады. Google дискісінен файлды жүктеу үшін аккаунтка кіру қажет. Файлдарды жүктеу үшін *Менің дискім* (Мой диск) → *Құру* (Создать) тармағын таңдау керек (2.2-сурет).



2.2-сурет



2.3-сурет

Ашылмалы тізімнен қатынасу деңгейін таңдап, оған қолдануға рұқсат берілетін пайдаланушылардың электрондық пошта мекенжайларының тізімін енгізуге болады.

Сілтемеге қол жеткізу (Доступ по ссылке) батырмасы жоғарыда орналасқан. Одан кейін, осы құжатқа сілтеме барлық пайдаланушыларға қолжетімді болатын әрекеттерді таңдау арқылы параметрді (опцияны) конфигурациялауға болады.

Осы әдістің артықшылығы — әрбір пайдаланушыны жеке қосу қажеттілігінің болмауы. Алайда оның кемшілігі: кез келген кездейсоқ адамға осы парақша қолжетімді болады. Бірақ кейде құжатты жалпыға қолжетімді етіп, сілтемеге тек оқуға арналған қатынасу мүмкіндігін беру ыңғайлы, содан кейін файлдарды өңдеуге қатысатындар үшін электрондық пошта мекенжайларының тізімі қосымша көрсетіледі. Ол үшін *Кеңейтілген* (Расширенный) сілтемесін басып, құжатқа қатынау мүмкіндігі бар пайдаланушылар тізімін және олар қандай құқықтарға ие екенін қараңдар.

Құжатқа қатысты кез келген пайдаланушының құқығын өзгертуге, сонымен қатар оны қосып немесе жоя аласындар (2.4-сурет).

Жаңа пайдаланушы файлға қатынау мүмкіндігіне рұқсат бергенде, өздерінің электрондық пошталарына сәйкес хабарлама алады және жаңа файлға сілтемені пайдаланып байланыса алады.

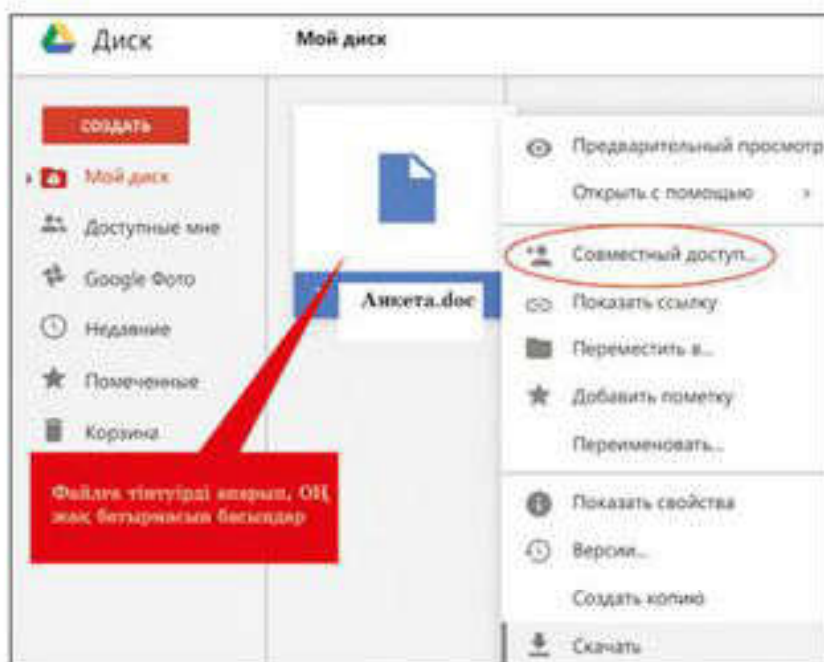
Құжаттармен *online* түрінде бірігіп жұмыс істеген кезде бір уақытта осы файлмен қанша пайдаланушы жұмыс істеп отырғанын

Сондай-ақ Google дискі электрондық кестелермен және презентацияларға арналған өзінің қосымшаларымен, құжаттармен жұмыс істеуге мүмкіндік береді (2.3-сурет).

Аталған қызметте кеңсе құжаттарымен жеке және басқа пайдаланушылармен бірге де жұмыс істей аласындар.

Мұны орындау үшін, жаңа құжат құрғаннан кейін, ол мәтін, кесте немесе презентация болсын, *Кіру параметрлері* (Настройка доступа) батырмасын басу арқылы басқа пайдаланушыларға қолдану рұқсатын қосу қажет.

Содан кейін файлдармен қандай әрекеттерді орындау керектігін анықтаңдар. Мәселен, белгілі бір адамдарға мазмұнды көруге, оқу үшін, түсініктемелер қалдыру немесе редакциялау мүмкіндігін беруге болады (2.3-сурет).



2.4-сурет

белгішелер арқылы көруге болады. Сонымен қатар чат батырмасын басу арқылы құжатты талқылап, пікірлерінді қалдыра аласындар.

Құжатпен жұмыс істеу барысында нақты уақыттағы белгілі әрекеттерді көруге болады, мысалы; бір пайдаланушының қай жерде орналасқанын, жасаған әрекеттерін, жою, өңдеу және т.б. (2.5-сурет).

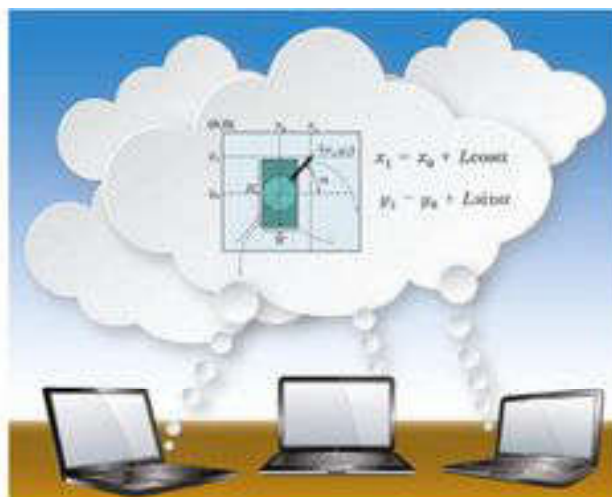
Осылайша кез келген құжаттармен бір уақытта бірігіп жұмыс істей аласындар. Бұл тез және ыңғайлы, ең бастысы құжаттармен кез келген құралда жұмысты жалғастыруға болады (2.6-сурет).

Сондай-ақ заманауи компьютерлерді, ноутбуктерді және ұялы телефондарды ұзақ уақыт пайдаланудың мынадай зиянды жақтары бар:

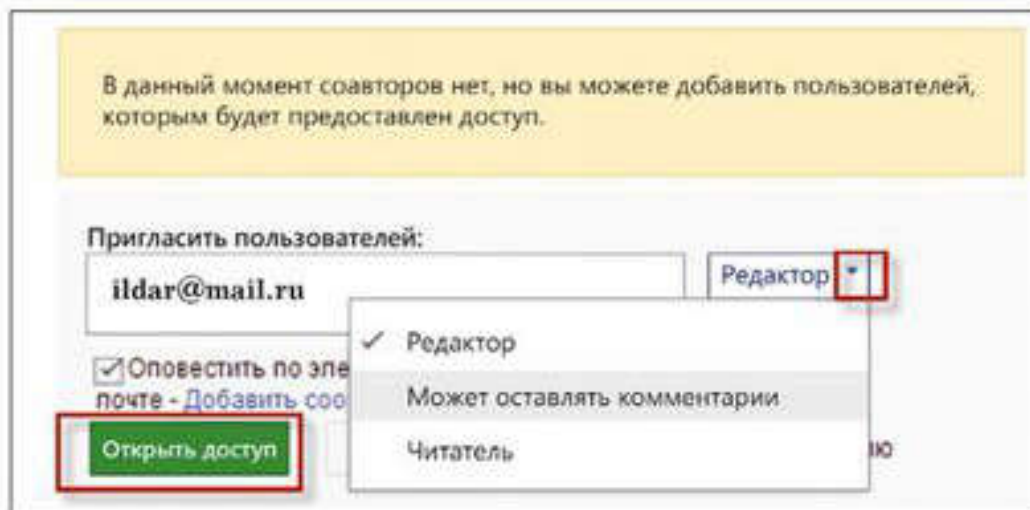
- компьютер мониторуна жақын отырсаңдар, көзге көп жүктеме түсіп, көздің көру қабілеті нашарлайды;

- егер ұзақ уақыт бойы қозғалмай компьютерде отырса, омыртқа жотасына, бұлшық еттерге салмақ түседі және қан айналымының бұзылуына әкеліп соғады;

- компьютер мен принтер жұмыс істегенде қосымша шудың деңгейі өсіп, миға қосымша жүктеме түседі;



2.5-сурет



2.6-сурет

- егер ұзақ уақыт ойын ойнайтын болсаңдар, ол сендердің эмоционалдық жағдайларына әсер етуі мүмкін;
- компьютердегі үлкен ақпаратты өңдеу ұқыптылықты және тиянақтылықты талап етеді, ал ол жүйке жүйесіне әрі эмоционалдық күйзеліске алып келуі мүмкін.

БҰЛ ҚЫЗЫҚ!

Microsoft компаниясының негізін қалаушы Билл Гейтс балаларына компьютерлер мен мобильді құрылғылармен күнделікті 45 мин және демалыс күндері бір сағат ойын ойнауға болады деген шектеулер енгізді. Ол компьютер тек үй тапсырмаларын орындау үшін қолдану керек деп есептеген.



! **Бұлттық технологиялар** (есептеулер) — бұл компьютерлік ресурстар мен қуаттарды пайдаланушыға интернет-қызметінің көмегімен деректерді өңдеуге мүмкіндік береді.

Бақылау сұрақтары



1. Бұлтты технологиялар дегеніміз не?
2. Бұлтты технологиялар қандай мүмкіндіктер ұсынады?
3. Бұлтты технологиялардың артықшылықтары мен кемшіліктерін атаңдар.

Практикалық жұмыс

А деңгейі

1. Google akkaунтын құрындар немесе қолданыстағы akkaунттарына кіріңдер.
2. Google Drive (диск) қызметін пайдаланып, бірнеше бума ашындар, деректерді жүктеп, түрлі бумаларға орналастырындар, файлды жүктеңдер, кейін оны жойындар.

В деңгейі

“Бірлесіп презентация жасау”.

Әрбір оқушы әртүрлі нысандарды (фотосуреттер, бейнелер, кестелер және т.б.) пайдаланып, “Өзім жайлы” деген бет дайындандар. Презентация слайдтарының дизайнын, мазмұнын бірге талқыландар. Google бұлттық қызметінің көмегімен сыныптарың туралы презентация әзірлендер.

С деңгейі

“Google Форм арқылы интернет-сауалнама жүргізу”.

1. Google-де akkaунт жазбаларыңды құрындар немесе оған кіріңдер.
2. Кез келген сегменттегі (өнім, қызмет, бренд) тұтынушылық преференцияларды зерттеу үшін сауалнама формасын жасандар.
3. Аталуына өздеріңнің тектеріңді енгізіңдер.
4. Әртүрлі сұрақтарды (3-тен кем емес) дайындандар.
5. Мазмұн тақырыбын қолданындар.
6. Мазмұнның сипаттамасына жауап беруші респонденттің жіберген уақытына алғыс айтындар. Оның бұл сауалнамаға қатысуы өте маңызды екенін ескертіндер.
7. Тексеру мәтінінің соңына “Сауалнамаға қатысқаның үшін рақмет” сөйлемін енгізіңдер.

Рефлексия:

Менің білгенім... . Енді мен ... қолдана аламын.

§4 Желілік этикет

Бүгінгі сабақта:

- ▶ желіде жұмыс істеу барысында этикалық және құқықтық нормалардың бұзылу салдары туралы ой қорытып айтуды үйренесіңдер.

Тірек ұғымдар:

- ▶ желілік этикет;
- ▶ компьютерлік этика;
- ▶ нетикет (сетикет);

Интернетте де, әдеттегідей қарым-қатынастың белгілі бір ережелері бар. Егер кәдімгі өмірде бұл қарым-қатынас этикасы болса, онда информатикада бұл — *компьютер этикасы*.



Компьютер этикасы аппаратты қолданушылардың мінез-құлқын зерттейді.

Мысалы, интернеттен материалдарды көшіру кезінде авторлық құқықтар туралы естен шығармау қажет. Интернеттегі кез келген ақпарат (мәтін, суреттер, фотосуреттер, бағдарламалар, ойындар) авторлық құқықпен қорғалған. Бұл ақпаратты өз мақсаттарың үшін автордың келісімімен ғана пайдалана аласыңдар. Мысалы, электрондық пошта арқылы автордың жазбаша рұқсатын алуға болады. Кейде веб-парақтарда материалдарды пайдалану шарттары: материалды сақтап, көшіруге болатыны көрсетіліп тұрады.

Егер қандай да бір тақырып бойынша реферат дайындаған кезде интернеттегі ақпаратты қолдансаңдар, авторды (егер белгілі болса) және ақпарат алынған деректер көзін, веб-сайттың мекенжайын көрсетіңдер.

“Авторлық және сабақтас құқықтар туралы” Қазақстан Республикасының Заңдарында қандай жағдайларда автордың келісімінсіз ақпарат пайдалану мүмкін екендігі жазылған.



Нетикет дегеніміз не?

Бұл ұғым XX ғасырдың 80-жылдарында пайда болды.



Нетикет (ағылшынша *net* — “желі”) — мінез-құлық ережелері, интернет желісіндегі қарым-қатынас, интернет-қоғамдастықтың көпшілігі қолданатын ережелер.

Виртуалды тілдесу — шынайы өмірдегі тілдесуден өзгеше. Көп адамдарға болашақта мүмкін кездеспейтін бейтаныс адам-

дармен, жолсеріктермен, кездейсоқ әңгімелесушімен және т.с.с. сөйлесу оңай. Мұндай қарым-қатынаста тәуелділік жойылып, байланыс “таза беттен” басталады. Шамамен, желідегі қарым-қатынас кезінде де осылай: тілдесушілер бір-бірі туралы ештеңе білмейді, адамгершілігі туралы да, кемшілігін де; ешкімді ештеңе қызықтырмайды. Мысалы, оның мектептегі бағалары т.с.с. Барлығы бастапқыда бірдей деңгейде болады. Осының барлығы адамға өзін еркін сезінуге көмектеседі, бірақ оның да белгілі бір шегі бар. Кейде виртуалды әңгіменің бір қатысушысы әріптестеріне дөрекі болуы, ашуландыруы, көңілін қалдыруы немесе жалықтыруы мүмкін. Желіде жұмыс істеген кезде жалпыға таныс ережелерді, қатынасу этикасын басшылыққа алу қажет. Нетикет ережелері жалпыға бірдей болмайды. Әрбір желілік қауымдастық өздерінің ережелерін қалыптастырады және олардың желілік тілдесу жолдары үшін өзінің ережелері бар. Соның кейбіреулерін қарастырайық.

Чаттарда, форумдардағы қарым-қатынас — жастар арасында желілік тілдесудің ең кең тараған демократиялық жолы. Чаттарда тілдесу кезінде келесі ережелерді ұстану қажет:

- қатысушылардың көбі тілдесетін тілді қолданындар. Ол шет тілдерге және сленгтерге де қатысты;

- “тым ақылды” болмандар. Ол арқылы әңгімелесушіні өздеріне қарсы қоюларың мүмкін. Қарапайым және түсінікті хабарлама жіберіндер;

- егер тілдесуші қандай да бір себептермен ренжітсе немесе тақырып қызықты болмаса, онда чаттан шығындар;

- егер белгілі бір тақырыптағы мамандандырылған чатта болсаңдар, әңгіме тақырыбынан ауытқымандар. Жай тілдесуге арналған арнайы серверлер бар;

- егер хаттарына жауап қайтарылмаса, онда бірнеше рет жаза бермендер, беймаза немесе сабырсыз адам болып көрінбеу үшін бірнеше рет қайталамандар. Мүмкін, сендердің хаттарына жауап кейін жазылады;

- дөрекі сөздерді қолданбаңдар. Көптеген чаттарда тәртіп бұзшыларға серверге қатынауды шектеу түрінде жазалау қолданады.

Электрондық пошта бойынша тілдесу — интернеттің қолданып, көпшілікке мәлім және қолжетімді хабарласудың жолы. Хабарлар іскерлік (мысалы, “мұғалім — оқушы”, “мұғалім — оқушының анасы”) немесе жеке (“анасы — баласы” немесе жолдастар арасында) болуы мүмкін.

БҰЛ ҚЫЗЫҚ!

Тақырыптық чаттардың, форумдардың тұрақты қатысушылары хабарламаларда артық сөздерді жазбай, жауапты қысқа беруге көмектесетін қысқарған сөздер қолданады.

Сөздер	Қысқартылған (бас әріптермен жазылады)
Менің ойым бойынша	IMHO (In My Humble Opinion)
Кейін сөйлесеміз	TTYL (Talk To You Later)
Сендердің назарларыңа	FYI (For Your Information)
Айтпақшы	BTW (By The Way)
Менің білуімше	AFAIK (As Far As I Know)
Кездескенше	SUL (See You Later)



Егер хат жеке болса, оған этиканың катал талаптары қойылмайды. Жақындарына, достарына жазылған күнделікті немесе электронды хат кез келген пішінде жазылуы мүмкін. Жеке хаттың стилі және мазмұндау мәнері хаттың несі мен араласу деңгейіне байланысты.

Әдетте, іскерлік хат жазудың жалпыға ортақ қабылданған “алтын” ережелері бар, оларды білу және хат жазу кезінде оны басшылыққа алу керек:

- хаттың негізін ғана білдіріп, қысқа жазыңдар;
- сауатты жазыңдар;
- әдептілікті (адалдықты, мейірбандықты) талап етсеңдер, ол да сыпайы жауап береді;
- жауап жазған кезде жалпы сөздерді қолданбаңдар. Егер тілдесуді аяқтағылары келсе, әдепті түрде хабарлап, тілдесуді тоқтатыңдар;
- тақырып жолын толтырыңдар. Компьютерлік вирустың хатпен бірге тарау қаупіне байланысты көп қолданушылар тақырыбы жазылмаған, күмәнді хабарламаларды оқымай, өшіріп тастайды;
- хабарланған тақырыпты ұстаныңдар;
- жіберілген хабарламаға қол қойыңдар;
- үлкен файлдарды мұрағаттап жіберіңдер;
- хатта көпшілікке арналған маңызды және пайдалы ақпарат болмаса, онда оны пошта жіберу (рассылканың) көмегімен жібермеңдер. Мекенжайларың “кара тізімге” кіруі мүмкін;

• хат алушыға қызықты емес хаттарды жіберу (мысалы, әртүрлі жарнамалар) желілік этикеттің ережелерін бұзу болып саналады және электрондық *спам* деп аталады. Ережелерді бұзу қателігіне байланысты спамерді жазалауы мүмкін. Желіге келіп түскен хабарламадан интернет қызметтерін жеткізуші серверді оңай табу арқылы тәртіп бұзушыны анықтауға болады.

Электрондық пошта, ең алдымен, таныс адамдардың бір-бірімен байланысуларына арналған және онымен жұмыс істеу тәртібі әдеттегі телефон арқылы сөйлесумен бірдей.

Сонымен қатар электрондық пошта хаттарына жауап берудің де өзіндік ережелері бар. Жауаптың мағынасы кіріс хат белгілі немесе бейтаныс жіберушіден келгеніне байланысты:

• хатқа жауап жазуды созбандар. Таныс адамдарға кәдімгі телефонмен жауап бергендей жауап жазындар. Егер тап осы кезде келіп түскен өтінішті орындау мүмкіндігі болмаса, онда бір тәуліктен асырмай жауап беріндер. Хат бейтаныс адамнан келсе, онда үш күн аралығында жауап беру қажет;

• егер хат қызықты емес (мысалы, жарнама, бейтаныс файл) ақпаратты қамтыса, онда оған жауап берудің қажеті жоқ. Қатал, жағымсыз жауап болса да, кез келген жауап спамер үшін пошталық адрестің дұрыстығын растайды және басқа спамерлерден де хаттардың жаңа ағынын шақыруы мүмкін. Нақты электрондық мекенжайлар сатушылардың арасында сауда заты болып табылады, әсіресе, мекенжай несі туралы қосымша мәлімет белгілі болса (жынысы, жасы, білімі, жеке қызығушылықтары). Осы белгілерге байланысты электрондық поштадан келген сауалнамаларды толтыру керек емес, тіпті лотереяда үлкен ұтыс уәде етілсе де толтырмандар;

БҮЛ ҚЫЗЫҚ!

Электрондық хат алмасуда эмоцияларды білдіретін сөздердің орнында таңбалардың тіркесі қолданылады (смайлдар – ☺, ағылшынша "smiley", "күлкі"), *Смайлик* – әртүрлі таңбалардан құрастырылған бет әлпетті: қос нүкте – көз, сызықша – мұрын, жақша – ауыз. Егер бұл тіркестерге бүйірінен қараса, онда смайлдың көңіл күйі айқын болады – қуаныш, қайғы, мысалы:

:-) – күлімсіреу;

:(– тұнжырап қарау;

:-))) – күлкі;

:-(– мұңаю.



• электрондық пошта хаттарының ішінде файлдар тіркеліп жіберілуі мүмкін (мысалы, фотосуреттер, мәтіндік файлдар). Мұндай файлдарға бейтаныс адамнан жіберілген зат сияқты қарау керек. Жіберілген зат жарылғыш жабдыктан тұруы мүмкін (мысалы, компьютерлеріндегі барлық ақпаратты жоятын вирус). Файл таныс адамнан келсе, онда қандай файл және хаттың келу мақсаты міндетті түрде көрсетілуі керек. Хат бейтаныс адамнан келсе, онда қосып жіберілген файлды ашпай және оқымай жою керек. Мұндай әдіспен таратылған бағдарламалық, құжаттық файлдарда вирустар болады.

Желілік этикет догма (дәлелденбеген қағида) емес. Интернет желісінде қарым-қатынас ережелерін білмесең де, “басқалардан сыпайылықты талап етсең, өзің де сыпайы бол” қағидасын ұстанып, виртуалды тілдесушілермен жақсы араласуға болады.

Бақылау сұрақтары



1. Неге желілік тілдесудің этикасын сақтау керек?
2. Жалпы желілік тілдесудің ережелері қандай?
3. Электрондық хат алмасу кезінде қандай негізгі ережелерді басшылыққа алу керек?
4. Электрондық хатқа қосылған файлдарға қандай шектеулер қойылады?
5. Чаттағы ережелер қандай?
6. Желіде этикалық және құқықтық нормаларды бұзудың салдары қандай болуы мүмкін?

Практикалық жұмыс

А деңгейі

Интернет сайттардан, интернеттегі қарым-қатынас ережелерін іздеп, онымен танысындар.

В деңгейі

ҚР-ның “Авторлық және сабақтас құқықтар” Заңы бойынша шығармаларға авторлық құқық не үшін берілгені туралы айтылған. Сонымен қатар, электронды немесе интернет сайтындағы фотосуреттер, бейне, музыка, әдеби шығармалар авторлық құқық нысандары болып табылады. Ол автордың рұқсатынсыз

оның шығармаларын қолдануға: көрсетуге, таратуға, сатуға, өңдеуге және т.б. жасауға болмайтынын білдіреді. Зерттеу жұмысын жүргізейік.

Интернет желісінде ҚР-ның “Авторлық және сабақтас құқықтар туралы” Заңының көмегімен мына сұраққа жауап дайындаңдар: “Интернеттен суреттерді, бейнефайлдарды, музыкалық файлдарды компьютерге қандай жағдайларда сақтауға болады?”

С деңгейі

ҚР-ның “Авторлық және сабақтас құқықтар” Заңы бойынша шығармаларға авторлық құқық не үшін берілгені туралы айтылған. Сонымен қатар, электронды немесе интернет сайтындағы фотосуреттер, бейне, музыка, әдеби шығармалар авторлық құқық нысандары болып табылады. Ол автордың рұқсатынсыз оның шығармаларын қолдануға: көрсетуге, таратуға, сатуға, өңдеуге және т.б. жасауға болмайтынын білдіреді. Зерттеу жұмысын жүргізейік.

Интернет желісінде ҚР-ның “Авторлық құқық және сабақтас құқықтар” Заңын тауып, келесі сұраққа жауап дайындаңдар: “Веб-сайт құрып жатқан кезде оған интернеттен алынған музыкалық файлдар, бейнефайлдар, суреттер орналастыруға бола ма және қандай талаптар бар?”

Рефлексия:

Менің білгенім... . Енді мен ... қолдана аламын.

2- ТАРАУ

Компьютерді таңдау

Бұл тараудан білесіңдер:

- ▶ ДК конфигурациясын;
- ▶ қолданушының қажеттілігіне байланысты бағдарламалық жасақтаманы таңдауды;
- ▶ қолданбалы бағдарламаны;
- ▶ компьютерді таңдау ережесін.

Мынаны үйренесіңдер:

- ▶ компьютердің конфигурациясын анықтауды;
- ▶ қолданбалы бағдарламаның типін ажыратуды;
- ▶ компьютердің құнын анықтауды.

\$5

Компьютердің конфигурациясы

Бүгінгі сабақта:

- ▶ компьютердің міндетіне қарай конфигурациясын таңдауды үйренесіңдер.

Тірек ұғымдар:

- ▶ компьютер;
- ▶ монитор;
- ▶ процессор;
- ▶ аналық тақта;
- ▶ бейнекарта;
- ▶ дыбыстық карта;
- ▶ жүйелік карта;
- ▶ ішкі жады;
- ▶ сыртқы жады.



Сендер жақындарыңмен компьютер сатып алуға бірге бардыңдар ма? Бұл жағдайда неге көңіл бөлдіңдер? Компьютерді белгілі бір конфигурацияға сәйкес сатып алу керек деген ой келді ме? Компьютер конфигурациясы дегеніміз не?

Компьютер сатып алатын кезде екі негізгі мәселені ескергендерін жөн. Біріншіден , компьютер қандай мақсатта қолданылатынын анықтап алыңдар. Функционалдық тағайындалуына байланысты компьютерлерді кеңселік, білім алу, ойын, үйге, мультимедиа және т.б. түрлерге бөлуге болады.

Компьютерге арналған компоненттерді таңдау ережелерін қарастырайық:

— “кеңсеге арналған компьютер” негізінен, текст және сандық ақпаратты өңдеу, құжаттарды сканерлеу және интернетке қол жеткізуге арналған;

— “дизайнерге арналған компьютер” физикалық ақпарат, бейне-материалдар, веб-дизайн, сканерлеу құжаттары интернетке қол жеткізу үшін арналған. Экранның ажыратымдылығын, бейнекартаны, қатты дискіні бақылау маңызды;

— “компьютерлік клуб компьютерінің” процессор қуаты, бейнекарта және дыбыс картасының сапасы, желі деректерін беру жылдамдығы маңызды;

— “оқушы компьютері” үй тапсырмасына және бос уақытты өткізуге, мәтіндік және сандық мәліметтермен жұмыс жасауға, музыка тыңдауға, интернетті қолдануға арналған. Маңыздысы, бағасы қымбат болмауы керек.

Екіншіден, кейбір құрылғылардың аналық тақтамен үйлесімділігі. Бұл көбінесе қосылым интерфейсінің үйлесімділігіне қатысты айтылған. Себебі, әртүрлі процессор интерфейстеріне сәйкесінше аналық тақталардың өзіндік үлгілері шығарылады.

Мысалы, Intel фирмасының процессорлары үшін Socket1155, LGA 2011-3 және басқа интерфейстер, ал AMD фирмасының процессорлары үшін SocketAM3, SocketFM2, SocketS-AM2 интерфейстерін пайдаланады. Аналық тақтаны таңдау барысында оның процессорлық интерфейсіне назар аудару қажет.

Сондай-ақ компьютерді жабдықтау кезінде кейбір компоненттер, мысалы, бейнекарта, дыбыстық карта, желілік карталар аналық тақтада есепке алынуы қажет.

Демек, *компьютердің конфигурациясы дегеніміз — жүйелік блок бөлігі болып табылатын компьютерлік компоненттер сипаттамаларының жиынтығы*.

Компьютер құрамы туралы ең маңызды ақпарат — бұл процессор (GPU) түрі мен жұмыс істеу жиілігі, жедел жады көлемі (мөлшері) (ОЗУ, RAM), қатты диск (HDD) көлемі (мөлшері), бейнекартаның түрі мен үлгісі, дискінің болуы (CDROM, DVDRW, BlueRay).



Компьютердің конфигурациясына жүйелік блок бөлігі болып табылатын компьютер компоненттері сипаттамаларының жиынтығы кіреді.

Компьютердің конфигурациясын өздерің анықтауларыңа болады. Ол үшін:

1. *Бастау* (Пуск) батырмасын басып, *Бағдарламалар* (Программы) → *Стандартты* (Стандартный) → *Қызметтік* (Служебные) тармақтарын таңдаңдар.

2. *Бастау* (Пуск) → *Орындау* (Выполнить) батырмасын басып, іздеу жолына *msinfo32* теріңдер.

Содан кейін қажетті ақпаратты алып, мәтіндік құжатта орналастырыңдар.



Бақылау сұрақтары

1. Компьютердің конфигурациясы дегеніміз не?
2. Компьютердің конфигурациясын қалай анықтайды?

Практикалық жұмыс

А деңгейі

1. Үйдегі компьютер (ноутбуктің) конфигурациясының сипаттамасын анықтаңдар 5.1-кестені MS Word мәтіндік процессорде жасап, толықтырыңдар.

5.1- кесте

Компьютер компоненттері	Сипаттамалары
Процессор	Тактілік жиілік —
Жедел жад	Көлемі —
Бейнекарта	
Қатқыл диск	Көлемі —
Монитор	Диагональ

2. 5.1-кестеде берілмеген компоненттерді толықтырып жазыңдар.

В деңгейі

Интернет-дүкендердің веб-сайттарын және компьютерлік дүкендердің бағалар тізімдерін пайдаланып, MS Word мәтіндік процессорда 5.2-кестені толтырыңдар.

5.2- кесте

Компьютер компоненттері	Оқушыларға арналған		Кеңсеге арналған		Ойын компьютері	
	сипаттамасы	бағасы	сипаттамасы	бағасы	сипаттамасы	бағасы
1	2	3	4	5	6	7
Процессор						

Жалғасы

1	2	3	4	5	6	7
Жедел жады						
Бейнекарта						
Қатқыл диск						

С деңгей

Компьютерлік фирмаға компьютер сатып алуға тұтынушы келді. Тұтынушы ойын-сауыққа арналған компьютерді: фильмдерді көру үшін және теледидарға, интернетке қосу мүмкіндігі болуын қалайды. Компьютердің қосымша компоненттерінде сканер, сия бүріккіш принтер және WEB-камера болуы керек. Тұтынушыға компьютердің қандай компоненттерін ұсынасындар? Оны “Компоненттердің атауы — сипаттамалары — бағасы” кесте түрінде жазындар.

Рефлексия:

Менің білгенім Енді мен ... қолдана аламын.

§6

Бағдарламалық жасақтаманы таңдау

Бүгінгі сабақта:

- қолданушының қажеттілігіне байланысты бағдарламалық жасақтаманы таңдауды үйренесіңдер.

Тірек ұғымдар:

- Қолданбалы бағдарламалық жасақтама (ҚБЖ);
- Жүйелік бағдарламалық жасақтама (ЖБЖ);
- бағдарламалау жүйелері.

8-сыныптан бастап сендер компьютердің негізін оның өзгеруіне, толықтырылуына, дамуына мүмкіндік беретін *аппараттық жасақтама* (ағылшынша *hardware* — қатты өнім) және *бағдарламалық жасақтама* (ағылшынша *software* — жұмсақ өнім) құрайтынын білесіңдер (6.1-сурет). Кез келген қиыншылықты шешуде компьютерді пайдалану үшін оған деректерді енгізу, өңдеу және шығаруды жүзеге асыратын бағдарламалық жасақтама орнатылады.



6.1-сурет. Компьютердің негізі



Бағдарламалық жасақтама (БЖ) — компьютердің ұзақ мерзімді барлық құрылғыларында есте сақталатын бағдарламалар (тіркесімі) жиынтығы.

Компьютерлік жүйенің негізгі бөлігі ретінде бағдарламалық жасақтама компьютердің қолдану саласын анықтайды. Қолданыстағы бағдарламалық жасақтаманы пайдалану және компьютерде жана бағдарламаларды орнату арқылы:

— оны есепші немесе конструктор, мектеп директоры немесе агрономның жұмыс компьютеріне айналдыруға болады;

— әртүрлі тапсырмаларды орындайды: реферат дайындау, құжатты өңдеу, электрондық поштаға хабарлама жіберу және т.б.

Әрбір бағдарлама белгілі бір мәселені шешуге бағытталған. Бұл ретте олар компьютердің аппараттық құралдарымен тығыз өзара әрекеттеседі: орталық процессормен, енгізу/шығару, есте сақтау құрылғысымен т.б.

Әдетте, бағдарламалық жасақтама үш түрге бөледі: қолданбалы, жүйелік бағдарламалар және бағдарламалау жүйелері (6.2-сурет).



6.2-сурет. Бағдарламалық жасақтама классификациясы

БҰЛ ҚЫЗЫҚ!



“Microsoft” фирмасы бағдарламалық жасақтаманың жетекші өндірушісі және негізін қалаушы – Билл Гейтс.



Қолданбалы бағдарламалық жасақтама (ҚБЖ) нақты мәселелерді шешуге арналған. Бүгінгі күні көптеген қолданбалы бағдарламалар бар. ҚБЖ жатты және арнайы мақсаттағы бағдарламаларға бөлінеді. Жалпы мақсаттағы бағдарламалар мыналарды қамтиды (6.3-сурет):



6.3-сурет

- калькуляторлар;
 - графикалық редакторлар: Paint, Photoshop, Gimp, Inkscape;
 - мәтіндік редакторлар: Блокнот, WordPad Windows операциялық жүйесінде; nano, gedit, KWrite және Kate Linux операциялық жүйесінде;
 - мәтіндік процессорлар : OpenOffice.org Writer, Microsoft Word;
 - электрондық кестелер: Excel, OpenOffice.org Calc;
 - деректер базасы : Access, OpenOffice.org Base;
 - презентацияларды дайындау бағдарламалары : PowerPoint, OpenOffice.org Impress;
 - ойындар (басқатырғыштар, стратегия және т.б.).
- Арнайы бағдарламалар мыналарды қамтиды:
- компьютерлік сөздіктер мен энциклопедиялар;
 - бухгалтерлік бағдарламалар.

Жүйелік бағдарламалық жасақтама(ЖБЖ) компьютер мен басқа да бағдарламалар арасындағы байланыстырушы ретінде жұмыс істеп, оның құрылғыларын басқарады. Жаңа компьютерде жүйелік бағдарламалық жабдықтың қажетті бөлігі болады, онсыз компьютерде жұмыс істеу мүмкін емес. Бұл топқа операциялық жүйелер, драйверлер (сыртқы құрылғыларды басқару бағдарламалары) және утилиттер (қызметтік бағдарламалар) кіреді (6.4-сурет).

Бағдарламалау жүйелері программистерге түрлі бағдарламаларды жасауға мүмкіндік береді (Visual Basic, Delphi, C++ және т.б.) .



6.4-сурет. Арнайы мақсаттағы бағдарламалар

БҰЛ ҚЫЗЫҚ!

Қолданбалы бағдарламалар адам өміріндегі әртүрлі қызық бағыттарды қамтиды. Мысалы, Такара жапондық ойыншықтың өндірушілері мысық дыбысының (мияуының) аудармашысын жасап шығарды.

Бұрын бағдарламалық жасақтама нақты операциялық жүйеге “байланған” болатын, мәселен, кейбір бағдарламалар Windows-тің, ал басқалары Linux-тің басқаруымен ғана жұмыс істейді.



Кроссплатформалық бағдарлама — әртүрлі операциялық жүйелер үшін нұсқалары бар бағдарлама.

Қазіргі уақытта түрлі операциялық жүйелер үшін нұсқалары бар көптеген кроссплатформалық бағдарламалар әзірленді.



Тағайындалуы бойынша бағдарламалық жасақтаманы қолданбалы, жүйелік және бағдарламалау жүйелеріне бөлуге болады:

1. *Қолданбалы бағдарламалық жасақтамаға* : мәтін, кесте процессорлары, оқыту және тестілеу арнайы бағдарламалары және т.б.

2. *Жүйелік бағдарламалық жасақтамаға* : операциялық жүйелер, антивирустық бағдарламалар және т.б.

3. *Бағдарламалау жүйелері* (программалар тілдері, программалау жүйесі).

Бақылау сұрақтары



1. Компьютерлік бағдарламалық жасақтама дегеніміз не?
2. Бағдарламалық жасақтаманың түрлері қандай және олар немен ерекшеленеді?
3. Қандай қолданбалы бағдарламаларды білесіңдер? Олардың мақсаты қандай?
4. Hardware және Software терминдерін түсіндіріңдер.
5. Кроссплатформалық бағдарламалар дегеніміз не?

Практикалық жұмыс

А деңгейі

Зерттеу жұмысын жүргізіп, компьютерде қандай бағдарлама орнатылғанын анықтаңдар. Бағдарламаның қандай бағдарламалық жасақтаманың түріне жататынын тауып, 6.1-кестеге жазыңдар.

6.1- кесте

Бағдарлама	Бағдарламалық жасақтама түрі

В деңгейі

Зерттеу жұмысын жүргізіп, интернет жүйесінде кроссплатформалық бағдарламалардың мысалдарын табыңдар. Олар БЖ-ның қандай түріне жататынын анықтап, 6.2-кестені толтырыңдар.

6.2- кесте

Бағдарлама	Бағдарламалық жасақтаманың түрі

С деңгейі

“Тәжірибелі пайдаланушылар мен программистерге арналған БЖ” тақырыбына эссе жазыңдар.

Рефлексия:

Менің білгенім Енді мен ... қолдана аламын.

§7 Компьютер ді таңдау

Бүгінгі сабақта:

- компьютердің міндетіне қарай конфигурациясын таңдауды үйренесіңдер.

Тірек ұғымдар:

- кеңсеге арналған компьютер;
- дизайнерге арналған компьютер;
- әмбебап компьютер;
- ойын компьютері.




Компьютерді қалай таңдауға болады? Үйге немесе кеңсеге арналған компьютер таңдағанда қандай параметрлерге назар аудару керек?


Алдымен үйге қандай үлгідегі компьютерді таңдау қажеттігін қарастырайық. Олар: *стационарлық* (десктоп) немесе *шағын* (ноутбук, моноблок, неттоп) болады және көлемі жағынан ғана ажыратылып тұрған жоқ. Ноутбук, моноблок немесе неттоптар ықшам болғандықтан өзіңмен бірге алып жүруге қолайлы. Жұмыс үстелі компьютерлері ноутбуктарға қарағанда өнімдірек және стационарлық компьютерде кез келген жабдықтың жинағын алмастыра аласыңдар, ал ноутбукте алмастыру қиындық тудыруы мүмкін. Жедел жады және қатты диск, сонымен бірге сыртқы бейнекартасын да ауыстыруға болады. Бірақ олар кез келген ноутбук үлгісіне сәйкес келе бермейді.

Сондықтан компьютер сатып алмас бұрын маңызды шешім қабылдау керек. Таңдау кезінде маманға хабарласып немесе үлкен ұқыптылықпен таңдау қажет. Компьютерді таңдаған кезде мыналарды ескерген жөн:

Тапсырыс беру кезінде.

1. Көптеген мамандар жақсы беделге не дүкендерге тапсырыс беруге кеңес береді. Компьютерге тапсырыс бергенде менеджерге немесе жинақтаушыға жаңа компьютерден не күтетіндерінді және қандай бағаға алуға жоспарлап отырғандарыңды айту керек:

 компьютер сипаттамалары мен дизайнын қажеттіліктеріне және таңдауларына сай етіп жасау мүмкіндігі;

 компьютер құраушыларын дұрыс таңдау, жинақтау сапасы жинақтаушының жұмысына тікелей байланысты болады.

2. Тәуелсіз жинақтау кезінде:

Егер базалық білімдерің мен дағдыларыңа сенімді болсаңдар, өз қажеттіліктерің бойынша компьютерді дербес жинақтауға болады:

− бұл жағдайда тапсырыс беруге карағанда төмен бағаға және талаптарына сай жинақталады;

+ қажетті білім мен дағдылар болмаған жағдайда тенгерімсіз жүйе немесе қателіктер жіберу ықтималдығы жоғары болады.

Компьютерді бағасына байланысты 4 топқа бөлуге болады:

1. 150 мың тг дейін — бастапқы деңгей (**бюджеттік компьютер**). Осы топтағы компьютерлер қарапайым тапсырмаларға арналған, мысалы, мәтінмен жұмыс, интернет желісі, бейнероликтерді көру және ең қарапайым ойындар. Көбінесе кеңселерде қолданылады.

Бюджеттік дыбыссыз компьютер бағасының мысалы 7.1-кестеде көрсетілген.

7.1- кесте

Аталуы	Бағасы, тг
Morex CASO-25 60W корпусы (немесе ұқсас дыбыссыз mini-ITX, сыртқы қуат көзі — 60—90 Вт)	17 000
Kingston SSDNow UV400 120 Gb (SUV400S37/120G) жүйесі үшін SSD-диск	9 500
Аналық тақтамен бірге Asus N3150I-C 4 ядролы процессоры	34 000
Kingston ValueRAM DDR3 4Gb 1600GHz (KVR16N11S8/4) құраушысы, жедел жады	11 000
Барлығы	72 000

Азғантай ақшаға дыбыссыз, шағын (22×19×6 см), сенімді, 4 ядролы кеңсеге және интернет қолдануға арналған компьютер жинауға болады. Осы компьютердің құраушысына жылдам SSD-диск кіретіндіктен операциялық жүйе жылдам жүктеледі, офистік бағдарламалар, интернет желісі тез жұмыс істейді.

2. 150—400 мың тг аралығында — орташа деңгей (**қымбат емес ойынға арналған және әмбебап компьютерлер**). Осы топтағы компьютерлер ең көп таралған, өйткені бұл процессорлардың есептеуіш күші және осы топтағы компьютерлердің бейнекарталарының мүмкіндіктері қарапайым тапсырмалар үшін ғана емес, сондай-ақ күшті талап ететін бағдарламалар үшін жеткілікті болып табылады. Ол көптеген жағдайда орташа және максималды сапада қазіргі заманғы ойындарға мүмкіндік береді. Бұл көбінесе үйде қолданушылар үшін ыңғайлы таңдау (7.2-кесте).

7.2- кесте

Әмбебап компьютерлердің құнын есептеу

Аталуы	Бағасы, тг
Zalman Z1 Black корпусы + қуат блогы Chieftec APS-550SB (немесе басқа өндірушінің корпусы +400 Вт-тан бастап қуат блогы + кулер)	36 000
WD Caviar 1000GB WD10EZRZ жүйесі үшін HDD-диск	17 000
Kingston HyperX DDR4 2x4Gb 2666GHz (HX426C15FBK2/8) құраушысы, жедел жады	28 000
AMD үшін	
AMD Ryzen 3 2200G BOX процессоры	36 000
Asus A320M-K аналық тақта	24 000
Intel үшін	
Intel i3-8100 BOX процессоры	49 000
Asus PRIME H310M-K аналық тақта	24 000
AMD қорытындысы	142 000
Intel қорытындысы	155 000

Дыбыссыз, жылд ам және қымбат емес компьютер, жақсы процесорлар, 8 Гб жедел жады және өнімді кіріктірілген бейне, заманауи бағдарламаларды еркін, қолайлы қолдануға болады.

3. 400 мың тг-ден жоғары — жоғары деңгей (орта және қуатты компьютерлер). Жеке компьютер, ерекше өндірушілер, процесордың қуаты қазіргі заманауи кемшіліктерді ең жаңа және ең күрделі тапсырмалармен максималды сапа параметрлерінде шешуге мүмкіндік береді. *Кемшілігі* — құны жоғары бағада болуы, сондай-ақ салқындату жүйесінің шуы. Мұндай компьютерлер компьютерлік клубтарда қолдануға ыңғайлы (7.3-кесте).

7.3- кесте

Орташа ойын компьютері бағасының құны

Аталуы	Бағасы, тг
1	2
Zalman Z1 Black корпусы + Chieftec APS-550SB қуат блогы (немесе басқа өндірушінің корпусы +500 Вт-тан бастап қуат блогы + кулер)	36 000

Жалғасы

1	2
Samsung 970 EVO 500 Gb M.2 NVMe (MZ-V7E500BW) жүйесіне арналған SSD-диск	53 000
WD Caviar 2000GB WD20EZRZ арналған HDD-диск	24 000
Asus PH-GTX1060-6G бейнекарта	117 000
Kingston HyperX DDR4 3200GHz 2x8Gb (HX432C16PB3K2/16) құраушысы, жедел жады	64 000
AMD үшін	
AMD Ryzen 5 2600 BOX процессоры	72 000
Asus PRIME B350-PLUS аналық тақта	41 000
Intel үшін	
Intel Core i5-8400 BOX процессоры	87 000
Asus Prime H370-PLUS аналық тақта	51 000
AMD қорытындысы	410 000
Intel қорытындысы	485 000

Орташадан жоғары деңгейдегі процессорлармен, жақсы бейнекартасымен және өте жылдам SSD M.2 NVMe жүйесімен тез жұмыс істеуге және ультра сапалы параметрлерге барынша және заманауи ойындармен ойнауға өте ыңғайлы (7.4-кесте).

7.4- кесте

Қуатты ойын компьютерінің құнын есептеу

Аталуы	Бағасы, тг
1	2
Zalman Z3 Plus корпусы + Chieftec APS-600SB қуат блогы (немесе 600 Вт-тан басталған қуат блогы + кулер)	43 000
Samsung 970 EVO 500 Gb M.2 NVMe (MZ-V7E500BW) жүйесі үшін SSD-диск	53 000
WD Caviar 2000GB WD20EZRZ жүйесі үшін HDD-диск	24 000
Asus GeForce RTX 2070 8GB (DUAL-RTX2070-8G) бейнекарта	228 000
Kingston HyperX DDR4 3200GHz 2x8Gb (HX432C16PB3K2/16) құраушысы, жедел жады	64 000

Жалғасы

1	2
AMD үшін	
AMD Ryzen 7 2700 процессоры + кулер	106 000
Asus PRIME X370-PRO аналық тақта	53 000
Intel үшін	
Intel Core i7-8700 процессоры + кулер	136 000
Asus TUF Z390-PRO GAMING аналық тақта	76 000
AMD қорытындысы	573 000
Intel қорытындысы	513 000

Өте қуатты компьютермен кез келген үлкен бағдарламаларды орындауға, қуатты ойындар ойнауға болады. Бұл компьютерлер интернетте ондаған немесе жүздеген бетбелгілерді аша алады, сапасы ультра-максималды бапталған ойындарды ойнауға және бейнежазбаларды өндей алады.

4. Арнайы компьютерлер — инженерлерге, бейнемонтаждаушыға арналған жаңа бейнекартамен жабдықталған ең соңғы үлгідегі дербес компьютерлер.

Ең жаңа процессорды және видеокартаны сатып алу кезінде осы кезеңдерді ескерсеңдер, компьютерді таңдау қиындық тудырмайды.

Бақылау сұрақтары



1. Дербес компьютердің компоненттерін өздерің жинақтаған кездегі артықшылықтар мен кемшіліктерді атаңдар.
2. Кеңсе компьютерлері мен арнайы компьютер арасындағы айырмашылықтар қандай?

Практикалық жұмыс

А деңгейі

Интернет ресурстарының көмегімен 7.5-кестедегі өте қуатты ойын компьютерінің бағасын есептендер.

7.5- кесте

Аталуы	Бағасы, тг
Zalman Z3 Plus корпусы + Chieftec APS-650SB қуат блогы (немесе басқа өндірушінің корпусы + 650 Вт-тан басталатын коректендіру блогы + кулер)	
Samsung 970 EVO 1000 Gb M.2 NVMe (MZ-V7E1T0BW) жүйесі үшін SSD-диск	
WD Caviar 4000GB WD40EZRZ жүйесі үшін HDD-диск	
Asus GeForce RTX 2080 8GB (DUAL-RTX2080-8G) бейнекарта	
Kingston HyperX DDR4 3200MHz 4x8Gb (HX432C16PB3K4/32) құраушысы, жедел жапы	
AMD үшін	
AMD Ryzen 7 2700X процессоры + кулер	
Asus PRIME X370-PRO аналық тақта	
Intel үшін	
Intel Core i9-9900K процессоры + кулер	
Asus TUF Z390-PRO GAMING аналық тақта	
AMD қорытындысы	
Intel қорытындысы	

В деңгейі

1. Қазақстандық интернет ресурстарын пайдаланып html бетін жүктендер.
2. “ДК компоненттері”, “ДК жинақтау процесі” тарауы бойынша теориялық ақпаратты оқып, танысындар
3. 1, 2, 3-тапсырмаларын орындандар.
4. Тест сұрақтарына жауап беріндер.

С деңгейі

В деңгейіндегі тест тапсырмаларын орындағаннан кейін ДК жүйелік блогының компоненттерін жинақтандар.

Рефлексия:

Менің білгенім Енді мен ... қолдана аламын.

3-
ТАРАУ

Мәліметтер қоры

Бұл тараудан білесіңдер:

- ▮ мәліметтер қоры дегеніміз не;
- ▮ электрондық кестеде мәліметтер қорын қалай құруға болады;
- ▮ мәліметтерді іздеу, сұрыптау және сүзгіден өткізуді қалай жүзеге асырады.

Мынаны үйренесіңдер:

- ▮ электрондық кестеде мәліметтер қорын құруды;
- ▮ мәліметтерді іздеу, сұрыптау және сүзгіден өткізуді.

§8

Мәліметтер қоры

Бүгінгі сабақта:

- ▮ мәліметтер қоры, жазба, өріс терминдерінің түсіндірмесімен танысасыңдар.

Тірек ұғымдар:

- ▮ мәліметтер қоры;
- ▮ жазба;
- ▮ өріс.

Ақпаратты сақтау ғана емес, оны тез іздеп табу да маңызды екенін білесіңдер. Компьютер пайда болғанға дейін адамзат ақпаратты түрлі ақпарат тасымалдағыш құралдарда сақтап келді. XX ғасырда, негізінен, емханадағы карточкалар, кітапханадағы каталог, экономика бөліміндегі картотекалар және т.б. қағаз тасымалдағыштары болды.



Қор дегеніміз не? Мәліметтер қоры дегеніміз не?

Мәліметтер қоры — қарастырылып отырған аймақ нысандары туралы мәліметтер жиынтығы, олардың қасиеттері мен өзара қатынастары ретінде берілген ақпараттар.

Сонымен, мәліметтер қорын кітапханалық каталогтер, кез келген мекемедегі кадрлар бөлімінде жұмыс істейтін қызметкерлер туралы ақпараты бар қалталар немесе мектеп журналы деп санауға болады.

Мәліметтер қоры, негізінен, қандай да бір нысандар туралы көптеген ақпаратты сақтауға, іздеуге көмектеседі. Мысалы, қаланың дәріханаларында сатылатын дәрілер туралы мәліметтерді қамтитын мәліметтер қорынан, сатып алушының тапсырысы бойынша, нақты бір дәрінің қай дәріханада сатылатынын және оның бағасы туралы ақпаратты табуға болады. Мәліметтер қорымен негізгі жасалатын амал — ол ақпараттарды алғашқы мәліметтермен толтыру. Содан кейін, мәліметтерді мүмкіндігінше өзгертуге, сұрыптауға, қажетті мәліметтер іздеуге болады (мысалы, дәріханаларға жаңа дәрілерді қабылдау немесе олардың бағаларының өзгерісі және т.б. туралы ақпараттарды енгізу).

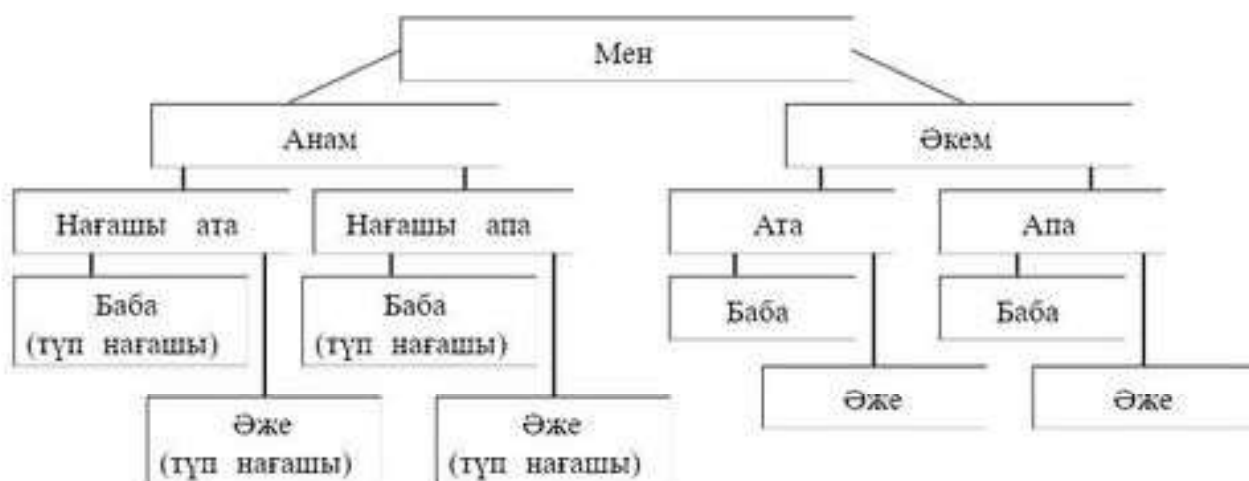


Мәліметтер қоры (МК) — әртүрлі қасиеттер жиынтығына ие нысандар тобы туралы мәліметтерді реттелген пішінде сақтауға мүмкіндік беретін ақпараттық модель.

Мәліметтер қоры *желілік, иерархиялық және кестелік* болып бөлінеді.

Иерархиялық мәліметтер қоры әртүрлі деңгейлі нысандардан тұратын ағаш түрінде көрсетуге болады. Ең жоғары деңгей — бірінші нысан, екіншісі — екінші деңгей нысандары және т.с.с. Осындай нысандар әулеттің шежіресіне байланысты, мысалы, әулет нысанының бір немесе бірнеше ұрпағы болуы мүмкін және олардың бір ғана ата-бабасы болады.

Иерархиялық деректер қорының ең көрнекті мысалы ретінде генеалогиялық ағашты ұсынуға болады (8.1-сурет).

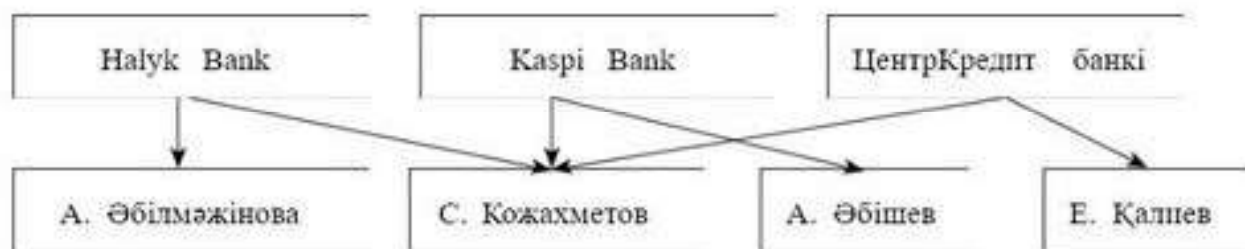


8.1-сурет. Генеалогиялық ағаш

Тағы бір мысал — ол ақпаратты (файлдар түрінде) компьютерде орналастыру. Ағаш түрінде белгілі бір құрылым каталог (буна) болады.

Желілік мәліметтер қоры — бір ата-бабадан көп нысандардың таралуына байланысты иерархиялық мәліметтер қорының жиынтығы болады.

Желілік мәліметтер қорының мысалына интернет желісін жатқызуға болады. Гиперсілтемелер жүздеген миллион құжаттарды бірінғай үлестірілген желілік мәліметтер қорына байланыстырады. Желілік мәліметтер қорының мысалы төменде көрсетілген (8.2-сурет).



8.2-сурет. “Салымшылар” мәліметтер қоры

Енді *кестелік* мәліметтер қорын талқылайық. Кестелік мәліметтер қоры бірдей сипаттағы қасиеттерге ие нысандар жиынтығын қамтиды. Мұндай мәліметтер қоры екі өлшемді кесте түрінде беріледі (8.3-сурет).

Жазба

Өріс

Тегі	Аты	Мекенжайы	Телефон
Әбішев	Аңсаған	Ғ. Мүсірепов к., 364-үй, 512-п.	8(727) 123 45 67
Мусин	Нұржан	М. Жұмабаев к., 968-үй, 245-п.	8(727) 891 01 23
Васильев	Егор	Абай даңғ., 725-үй, 345-п.	8(727) 456 78 90

8.3-сурет

Кестенің бағандары *өрістер* деп аталады. Әрбір өріс оның қасиетінің мәнін көрсететін атауы бар мәліметтерден тұрады.



Мәліметтер қорының өрісі — белгілі бір сипаттағы мәндерді қамтитын кесте бағаны.

Кестенің жолдары нысандар туралы жазбалар болып табылады. Әрбір жазба өрістердегі мәндердің жиынтығынан тұрады.



Мәліметтер қорының жазбасы — мәліметтер қорының өрістерінде орналасқан белгілі бір сипаттағы мәндердің жиынтығын қамтитын кесте жолы.

Бақылау сұрақтары

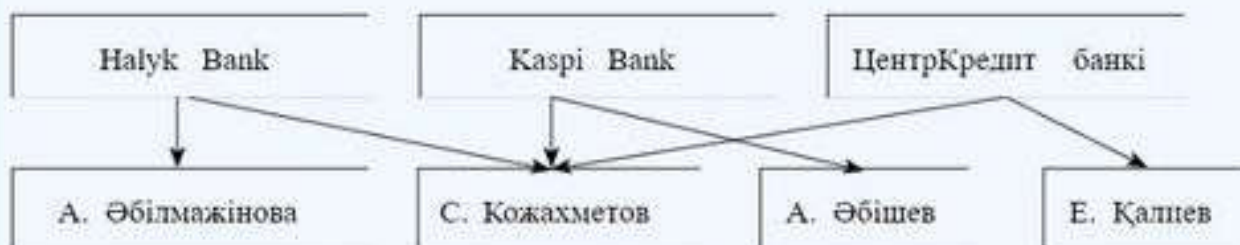


1. Мәліметтер қоры дегеніміз не?
2. Мәліметтер қорының жазбасы және өрісінің өзара айырмашылығы қандай?

Практикалық жұмыс

А деңгейі

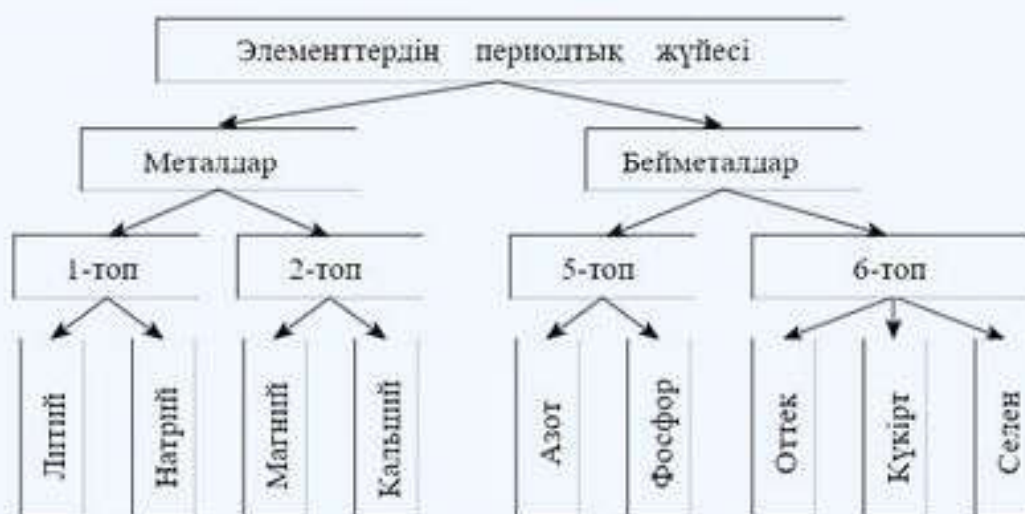
8.4-суретте “салымшылар” мәліметтер қорының желілік құрылымы берілген. Осы құрылымды кестелік түрге түрлендіріңдер.



8.4-сурет. “Салымшылар” мәліметтер қоры

В деңгейі

Төмендегі “Менделеев кестесі” мәліметтер қорының иерархиялық құрылымы берілген (8.5-сурет). Бұл құрылымды кестелік мәліметтер қорына келтіріңдер.



8.5-сурет. “Менделеев кестесінің” мәліметтер қоры

С деңгейі

“Мектеп” иерархиялық мәліметтер қорын “элементтердің периодтық жүйесі” кестесіне сәйкес құрып, толтырындар.

Рефлексия:

Менің білгенім Енді мен ... қолдана аламын.

§9

Электрондық кестелерде мәліметтер қорын құру

Бүгінгі сабақта:

- ▶ MS Excel мәліметтер қорын құруды үйренесіндер.

Тірек ұғымдар:

- ▶ мәліметтер қоры;
- ▶ тізім;
- ▶ өріс;
- ▶ жазба;
- ▶ жол;
- ▶ баған.

MS Excel бағдарламасында мәліметтер қорын құруға және өңдеуге болатынын білдік.

MS Excel бағдарламасының көмегімен мәліметтер қорын құрып және өңдей аласындар. Бұл жағдайда электрондық кестені Excel бағдарламасындағы *тізім* немесе *мәліметтер қоры* деп атап, келесідей терминологияны пайдаланындар:

- тізім жолы — мәліметтер қорының жазбасы;
- тізім бағаны — мәліметтер қорының өрісі.

Баған атауы тек бір ұяшықты алып, кестемен жұмыс жасағанда *өріс* деп аталады.

Мысалы, кестенің A2:F12 аймағын мәліметтер қоры ретінде қарастырайық (9.1-сурет). Осы кестенің A, B, C, D, E, F бағандары — *өрістер*, ал 3-тен 12-ге дейінгі жолдар *жазбалар* деп аталады. A2:F12 аймағындағы өрістерге атаулар беріледі.

MS Excel мәліметтер қоры құрылымына шектеулер қойылады:

— бірінші жолдағы мәліметтердің өріс атаулары қайталанбауы тиіс;

- мәліметтер қорының жазбаларында бос жолдар болуы керек;
- өрістегі (бағандағы) ақпарат біртекті, демек, тек сандар немесе мәтіндер болуы тиіс.

“Күн жүйесіндегі ғаламшарлар” МК қарастырайық.

1	Күн жүйесіндегі ғаламшарлар					
2						
3	Ғаламшар	Период	Арақашықтық	Диаметр	Масса	Серік
4	Меркурий	0,24	58	4,9	0,32	0
5	Шолпан	0,62	108	12,1	1,86	0
6	Жер	1	150	12,8	6	1
7	Марс	1,8	288	6,8	0,61	2
8	Юпитер	11,86	778	142,6	1906,98	16
9	Сатурн	29,46	1426	120,2	570,9	17
10	Уран	84,01	2869	49	87,24	14
11	Нептун	164,8	4496	50,2	103,38	2

9.1-сурет. Күн жүйесіндегі ғаламшарлар туралы ақпарат

9.1-суреттегі өлшем бірліктер:

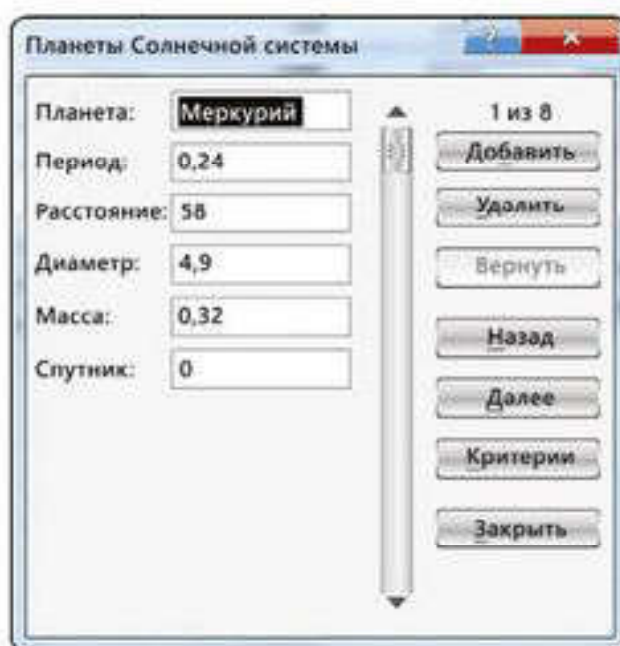
- орбитаны айналу периоды — жыл;
- Күннен орташа арақашықтық — млн км;
- экваторлық диаметр — мың км;
- масса — 10^{24} кг.

Мәліметтер қоры ретінде қарастырылатын кестенің негізгі ерекшелігі оның үлкен өлшемі болып табылады.

MS Excel бағдарламасында мәліметтер қорын құру кезінде орнатылған мәліметтер қорының пішіні ыңғайлы, демек:

- баған тақырыптарынан тұратын жолдың болуы;
- әрбір жазба жеке жолға орналастырылады;
- бірінші жазба тақырыптар жолынан кейінгі жолға орналастырылады;
- жазбалардың ортасында бос жолдар болмауы керек.

Кірістіру (Вставка) → Кестелер (Таблицы) → Кесте (Таблица) командаларын орындау қажет. Содан кейін *Құралдардың* диалогтік терезесін пайдаланып, кестенің ауқымын көрсетіндер (9.2-сурет).



9.2-сурет. Мәліметтермен жұмыс пішіні

Бақылау сұрақтары



1. Мәліметтер қорын кесте түрінде бейнелеудің артықшылықтары мен кемшіліктерін атаңдар.
2. Мәліметтер қорының жазбасы және өрісі дегеніміз не?
3. Мәліметтер қорында өрістердің қандай түрлері болуы мүмкін?
4. Мәліметтер қорында *Пишіндердің* (Формы) көмегімен жұмыс істеудің артықшылықтары мен кемшіліктерін атаңдар.

Практикалық жұмыс

А деңгейі

Мәліметтер қорын құрыңдар.

1. “Күн жүйесіндегі ғаламшарлар” үлгісіне сәйкес мәліметтер қорын құрыңдар (9.1-кесте).

9.1- кесте

“Күн жүйесіндегі ғаламшарлар” МҚ

Ғаламшар	Период	Арақашықтық	Диаметр	Масса	Серік

2. “Период”, “Арақашықтық”, “Диаметр”, “Масса”, “Серік” ұяшықтары үшін сандық *Пишін* (Форма) орнатыңдар.

3. Бірінші жазбаны енгізіндер. Екінші жазбадан бастап *Пишін* (Форма) командасы арқылы кестені толтырыңдар. Форма мәзірін шақыру үшін кесте түрінде деректер ауқымын көрсету және *Кірістіру* (Вставка) → *Кесте* (Таблицы) → *Кесте* (Таблица) таңдау қажет:

— *Жылдам кіру тақтасындағы* (Панель быстрого доступа) тінтуірдің оң жақ батырмасын басып, мәтінмәндік мәзірінен жылдам кіру тақтасын орнату тармағын таңдаңдар. Экранда Excel параметрлері диалогтік терезесінің жылдам кіру тақтасы пайда болады;

— Ашылмалы тізімнен *Пишін* (Форма) таңдап, *Қосы* (Далее) батырмасын басыңдар;

— Excel параметрлері диалогтік терезесін жабу үшін *OK* батырмасын басыңдар.

Осы қадамдарды орындағаннан кейін жылдам жолағы жана белгішені қосады. Оны басқанда деректерді енгізу үшін пішін көрсетіледі. Парактағы кез келген ұяшықты қосып, жылдам жолақтағы Пішін командасын таңдаңдар. Командаларды пайдаланбас бұрын кестенің бірінші жазбасын және іргелес тақырып жолын ерекшелендер.

4. Мәліметтер қорын жұмыс дискісіне “Күн жүйесі.xls” деген атаумен сақтаңдар.

B деңгейі

1. “Қазақстанда мекендейтін құстар мен жануарлар” мәліметтер қорын құрыңдар.

2. Пішіннің көмегімен “Қазақстандағы облыстар” мәліметтер қорына облыс орталығы, ауданы, халқы туралы жазба қосыңдар.

C деңгейі

Кеткен уақыт критерийі бойынша мәліметтер қорын жасаудың қай әдісі ыңғайлы екенін зерттеңдер. Ескертпе: бірдей екі мәліметтер қорын әртүрлі тәсілмен жасадыңдар, жұмсаған уақыттарыңды бағалап, қорытындылаңдар.

Рефлексия:

Менің білгенім Енді мен ... қолдана аламын.

§10 Ақпаратты іздеу әдістері

Бүгінгі сабақта:

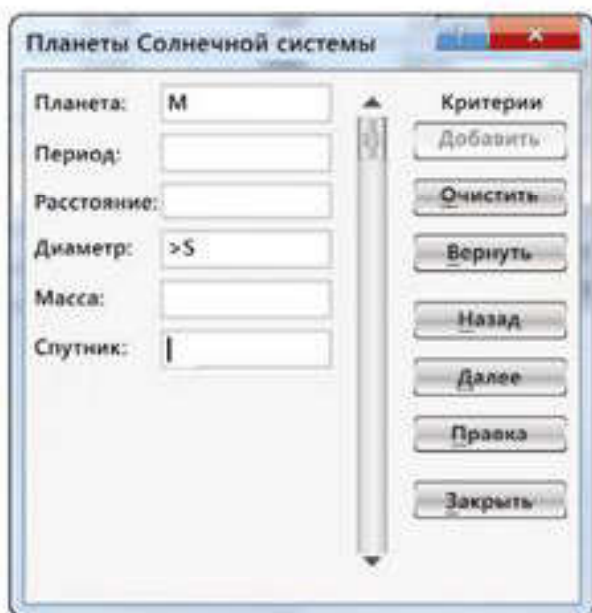
- ▶ мәліметтерді іздеуді, сұрыптауды және сүзуді жүзеге асыруды үйренесіңдер

Тірек ұғымдар:

- ▶ электронды кестедегі тізім;
- ▶ іздеу;
- ▶ жұмыс парағы.

Excel бағдарламасының тізімінде ақпаратты табудың ең оңай жолы — деректер пішінін пайдалану. Негізінен, тізімдерде нақты жазбаларды табу талап етіледі.

Ол үшін *Форма (Пішін)* мәзірін пайдаланыңдар. Содан кейін, *Деректер (Данные)* терезесінде *Критерийлер (Критерий)* батырмасын

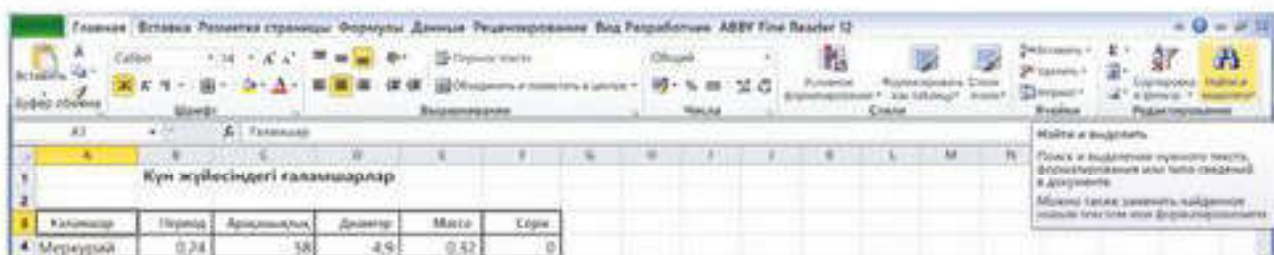


10.1-сурет. Критерийлер белгіленген пішін

басып, іздеуге қажетті сипаттамаларды енгізіндер. Мысалы, мәтіндік өрістен мәндер бойынша іздесендер, іздеген деректерді бірегей анықтау үшін бірінші таңбаларды таңдауларың керек. Егер бірнеше критерийді таңдасандар, іздеу бірнеше критерий бойынша орындалады (10.1-сурет).

? Пішіндерді, мәндерді және формулаларды пайдаланып іздеуді қалай жасайды?

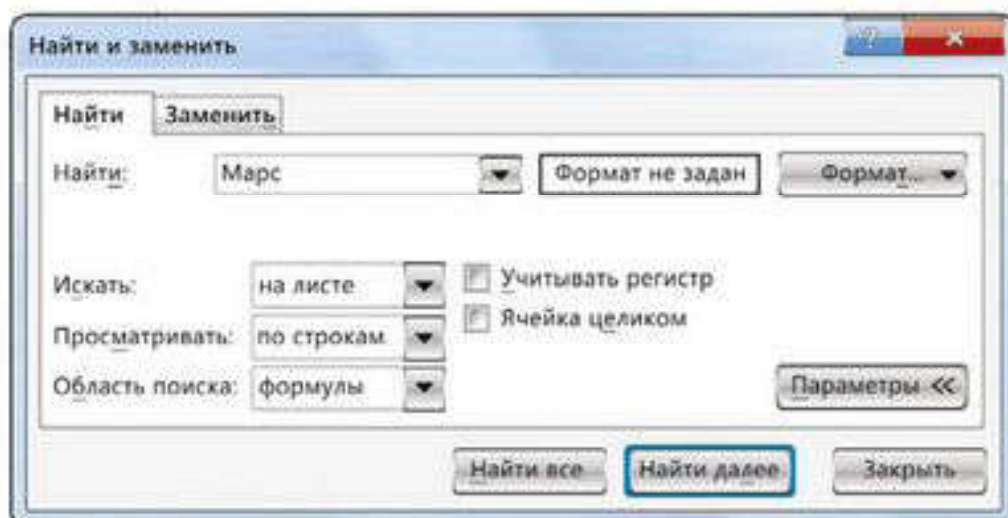
Деректердің пішіндерін пайдалану арқылы іздеу берілген өрістер (бағандар) бойынша жүзеге асырылатынын білесіңдер. Егер белгілі бір мәнді немесе формуланы бүкіл жұмыс парағынан іздеу қажет болса, онда (10.2-сурет) *Өңдеу (Правка)* мәзіріндегі *Іздеу (Найти)* процедурасын қолданған жөн. Ашылатын *Іздеу (Найти)* диалогтік терезесінен *Не (Что)* өрісіне ұяшық мазмұнының фрагментін енгізіндер. Содан кейін *Келесіні іздеу (Найти далее)* (10.3-сурет) батырмасын басындар. Егер іздеу сәтті болса, таңдалған фрагментті қамтитын ұяшық қосылады.



10.2-сурет. Іздеу командасын қолдану

Әдетте, *Іздеу* төмендегі жолдармен орындалады. *Іздеу* үлгісін өзгерту үшін, *Іздеу (Найти)* диалогтік терезесінде баған бойынша іздеуді көрсетулерің керек. Сондай-ақ қандай деректерді іздейтіндерінді, мәндерді немесе формулаларды көрсете аласындар.

Іздеу әрекетін орындаған кезде “*” таңбасын қолдануға болады. “*” таңбасы кез келген әріппен бірге қолданылады. Егер “M*” деп көрсетсек, “Марс” немесе “Меркурий” мәтіні бар ұяшық көрсетіледі.



10.3-сурет. Белгіленген критерийлер пішіні

Сұрақ белгісі “?” сандармен бірге көрсетіледі. Егер “1?” деп көрсек, онда 1-ден басталып, екі орынды сандар жазылып шығады.



Бақылау сұрақтары

1. Электрондық кестедегі тізім дегеніміз не?
2. Тізімдер қандай элементтерден тұрады?
3. Пішінді пайдаланып тізімді қалай құру керек?
4. Пішінді пайдаланып жазбаларды іздеу тәртібін сипаттаңдар.
5. Жұмыс парағында деректерді іздеуді қалай орындауға болады?

Практикалық жұмыс

А деңгейі

Құрылған мәліметтер қорында белгілі бір критерийлер бойынша іздеуді орындандар (мысалы, кез келген әріптің атауы арқылы және т.б.).

В деңгейі

Құрылған мәліметтер қорында белгілі бір критерийлер бойынша ортақ іздеу құралының көмегімен іздеуді орындандар.

С деңгейі

МҚ-да деректерді іздеудің қандай әдісі ең қолайлы болып табылады? Зерттеу жүргізіндер. Іздеу критерийлерін анықтап, қорытынды жасаңдар.

Рефлексия:

Менің білгенім Енді мен ... қолдана аламын.

§11-12 Деректерді сұрыптау және сүзу

Бүгінгі сабақта:

- ▶ мәліметтерді іздеуді, сұрыптауды және сүзуді жүзеге асыруды үйренесіндер.

Тірек ұғымдар:

- ▶ сұрыптау;
- ▶ сүзу;
- ▶ іздеу;
- ▶ деректер.

Деректерді сұрыптау әдісі барлық кестелерге қолданылады. Excel қажет болған жағдайда пайдаланылатын деректерді, кез келген өзгерістер мен толықтырудан кейін реттеу мүмкіндігін береді.

? *Электрондық кестедегі (ЭК) деректерді қалай сұрыптауға болады?*

ЭК-дегі деректерді алфавиттік ретпен өсуі немесе кемуі бойынша, күндер немесе сандардың мәнімен сұрыптауға болады (11.1-сурет).

	A	B	C	D
1	-10	бит	1 қаңтар 2017	20:30
2	-5	bit	3 наурыз 2018	16:30
3	0	S	12 қаңтар 2016	12:30
4	1		3 наурыз 2017	8:30
5	5		12 қаңтар 2015	4:30

11.1-сурет. Берілгендерді сұрыптау әдісі

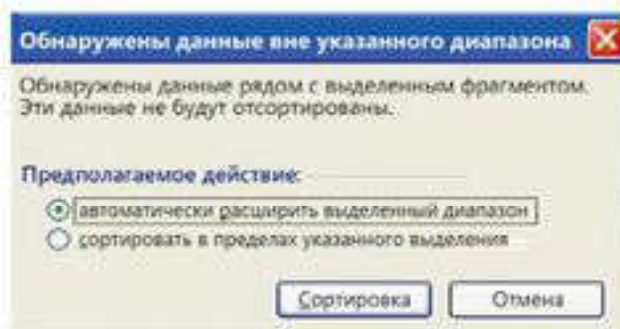
? *“Сұрыптауды” өсу реті бойынша таңдасақ, деректер қалай өзгереді деп ойлайсыңдар?*

Осу реті бойынша сұрыптау:

- сандар — ең аз теріс саннан ең үлкен оң санға дейін;
- мәтін — алфавиттік тәртіпте (сандар, таңбалар, латын алфавиті, орыс алфавиті);
- күндер мен уақыт — хронологиялық тәртіпте.

Бос ұяшықтарды әрқашан тізімнің соңына орналастырады.

Кему реті бойынша сұрыптау кезінде, деректер көрсетілген кері тәртіпте реттеледі .



11.2-сурет. Сұрыптау кезені



Сұрыптау — деректердің мәніне немесе деректер түріне сәйкес тізімдегі салыстырмалы орындарын өзгерту .

Бірнеше сұрыптау режимдері бар. Электрондық кестелерде кез келген бағанды таңдағаннан кейін сұрыпталатын деректер ауқымын кеңейтетін сұрыптау режимі бар (11.2-сурет) .

Кеңейтілген ауқымның нәтижесі 11.3-суретте көрсетілген.

	A	B	C	D
1	5	1	12 қаңтар 2015	4:30
2	1	5	3 наурыз 2017	8:30
3	0	S	12 қаңтар 2016	12:30
4	-5	bit	3 наурыз 2018	16:30
5	-10	бит	1 қаңтар 2017	20:30

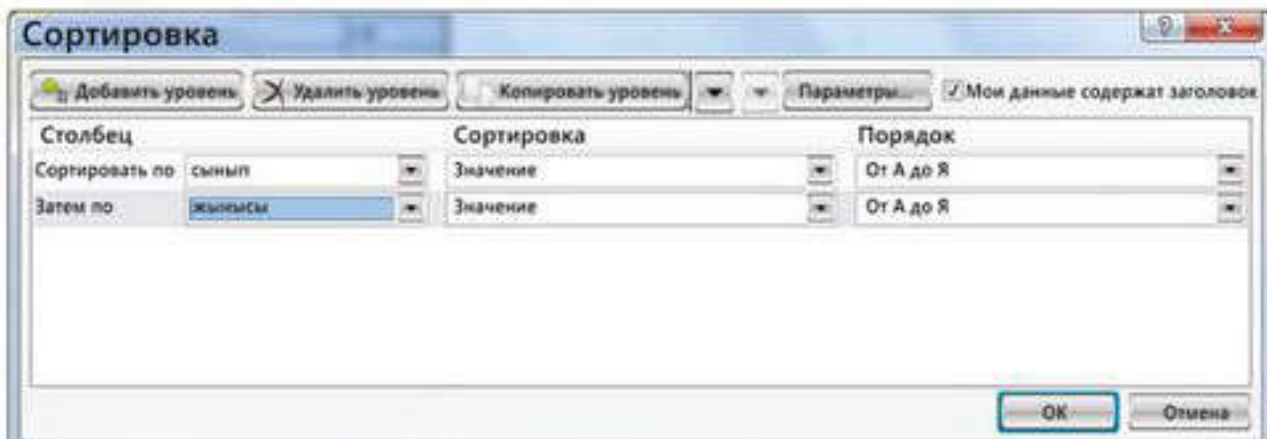
11.3-сурет. Ауқымды сұрыптау диапазоны

Ерекшеленген ауқымда сұрыптау . Кеңейтілген ауқымда сұрыптағаннан кейін жолдардың орналасу реті өзгереді, бірақ олардың тұтастығы сақталады (11.4-сурет).

	A	B	C	D
1	-10	бит	1 қаңтар 2017	20:30
2	5	бит	3 наурыз 2018	16:30
3	0	S	12 қаңтар 2016	12:30
4	1	S	3 наурыз 2017	8:30
5	5	L	12 қаңтар 2015	4:30

11.4-сурет. Сұрыптау әдісінен кейінгі көрініс

Таңдау бойынша сұрыптау қажет болса, оны бірнеше баған арқылы жүзеге асыруға болады. *Бастапқы (Главная) → Сұрыптау және сүзгі (Сортировка и фильтр) → Сұрыптауды баптау (Настраиваемая сортировка)*. Бірінші сұрыптау деңгейін таңдап, оның параметрін реттендер. Келесі сұрыптау деңгейін және т.б. қосындар (11.5-сурет).



11.5-сурет. Бірнеше бағандағы сүзгіден өткізуді баптау

Нәтижесінде алдымен сұрыптау арқылы деректер бірінші кезекте бірнеше баған бойынша сұрыпталады.

Бірінші бағанда, содан кейін екінші бағанда жалғасып ретімен орналасады. Алғашқы кесте және сұрыптау нәтижесі 11.6-суретте көрсетілген.

	А	В	С
1	Аты-жөні	жынысы	сынып
2	Ахметов Д.	ұл	1 А
3	Садықов Т.	ұл	1 Б
4	Баракова А.	қыз	4 Б
5	Ермеков К.	ұл	2 В
6	Ахметова Г.	қыз	1 Б
7	Темірғалиев Б.	ұл	1 А
8	Сафронов А.	ұл	1 А

	А	В	С
1	Аты-жөні	жынысы	сынып
2	Ахметов Д.	ұл	1 А
3	Сафронов А.	ұл	1 А
4	Темірғалиев Б.	ұл	1 А
5	Ахметова Г.	қыз	1 Б
6	Садықов Т.	ұл	1 Б
7	Ермеков К.	қыз	2 В
8	Баракова А.	ұл	4 Б

11.6-сурет. Дереккөз кестесі және сұрыпталған кесте

Сүзу (сүзгіден өткізу). Сүзгілерді пайдаланып іздеу шартын қанағаттандырмайтын алғашқы кесте жазбаларында жасырылған деректерді іздеуге болады (11.7-сурет).

11.8-суреттегі сүзуді қолданып, мына мәліметті аласындар: сүзгі қолданылған кестеде жалақы = 120 000 деп көрсетілген.



Электрондық кестедегі деректерді іздеу — сүзгі түрінде берілген іздеу шарттарын қанағаттандыратын жазбаларды (жолдарды) таңдау болып табылады.

	A	B	C	D
1	Аты-жөні	Еңбекақы, теңге	Жасы	Бөлімі
2	Ахметов А.А.	150 000	45	бухгалтерия
3	Смирнов Е.В.	120 000	28	өндіріс
4	Жакеев С.К.	100 000	21	бухгалтерия
5	Еремеева С.С.	180 000	50	қойма
6	Болатбек Е.С.	100 000	23	өндіріс
7	Низамутдинов В.К.	100 000	24	өндіріс
8	Жабагин С.М.	120 000	29	бухгалтерия
9	Гордиянова Г.В.	120 000	33	қойма

11.7-сурет. Бастапқы кесте

Жазбаларды іздеу шарттары салыстыру операторлары арқылы жасалады (11.8-сурет).

	A	B	C	D
1	Аты-жөні ▼	Еңбекақы, теңге ▼	Жасы ▼	Бөлім ▼
2	Ахметов А.А.	150 000	45	бухгалтерия
5	Еремеева С.С.	180 000	50	қойма

11.8-сурет. Сандық сүзгіні қолдану

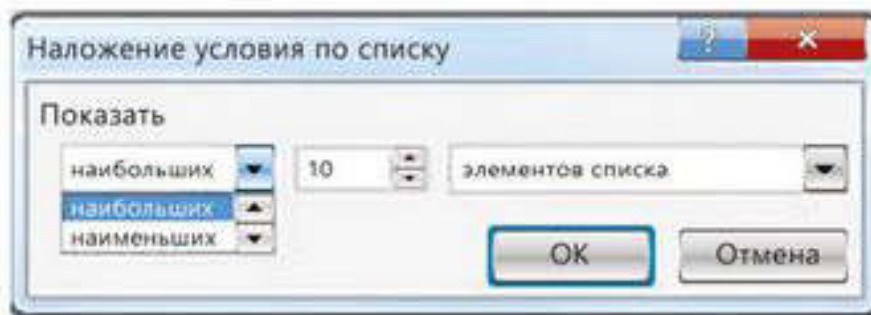
Мәтіндік деректер үшін амалдар :

- тең (барлық таңбалар салыстырылады);
- таңбадан басталады немесе басталмайды (бірінші таңбаларды салыстыру);
- таңбамен аяқталады немесе аяқталмайды (соңғы таңбаларды салыстыру);
- таңбалар тіркесі құрамында немесе құрамында жоқ (мәтіннің әртүрлі бөліктеріндегі таңбалардың қатарын салыстыру).

Мәтіндік сүзгілерден басқа, мәтіндік сүзгілермен бірдей шарттарды қабылдайтын “сандық сүзгілер” бар. Олар тек сандарға ғана қатысты:

- “Ортадан жоғары” немесе “Ортадан төмен” — орташа мәннен жоғары және төмен мәндерді қайтарады. Орта мен бағандардың тізіміндегі барлық сандық мәндерден есептеледі;
- “Алғашқы 10...” — осы элементтің терезесін шақырады (11.9-сурет).

Бұл терезеде қандай элементтерді көрсетуге болады: ең үлкен немесе ең кіші? Егер соңғы өрістегі “тізім элементтері” таңдалса, қандай элементтердің саны көрсетілетін болады? Егер “Элементтер санының %” таңдалса, онда екінші мән берілген пайызды орнатады. Тізімде 10 мән болса, ең жоғары (немесе ең төменгі) мән



11.9-сурет. Алғашқы сандық сүзгі

таңдалады. Ал тізімде 1000 мән болса, онда бірінші немесе соңғы 100 мән таңдалады.

Егер Күн мен уақыт түріндегі деректер болса, онда сүзгілерді күні (дата) бойынша қолдануға болады:

- барлық жазбалар берілген мерзімде көрсетіледі;
- барлық жазбалар көрсетілгеннен кейін, аталған мерзім шығарылады;
- алдымен барлық жазбалар арасындағы диапазонда көрсетілген күндер бойынша шығарылады.

Сүзгілердің уақытқа орай басқа да шарттары бар: берілген уақыттың барлық күндері, жыл басынан, өткен аптада және т.б.

Бақылау сұрақтары



1. Кестені сұрыптау қалай орындалады?
2. Деректер кестесінің қандай сипаттамалары бар?
3. Деректерді бірнеше критерий (талаптар) бойынша қалай сұрыптауға болады?

Практикалық жұмыс

А деңгейі

Кестедегі деректерді сұрыптау:

1. “MS Excel” бағдарламасын ашындар.
2. “Күн жүйесі.xls” құжатын ашындар.
3. Деректерді “Ғаламшарлар” бағаны бойынша өсу тәртібімен сұрыптандар. Сұрыптау нәтижесін 2-параққа көшіріңдер.
4. Деректерді “Арақашықтық” бағаны бойынша кему тәртібімен сұрыптандар. Сұрыптау нәтижесін 2-параққа көшіріңдер.
5. 2-парақ “Ғаламшар”, 3-парақ — “Арақашықтық” деп аталады.

В деңгейі

Мәтіндік, сандық және т.б. деректер үшін сұрыптау мен сүзуден өткізуді бірнеше критерийлер бойынша қолданыңдар.

С деңгейі

Зерттеу жұмысын жүргізіп, қарапайым және құрамдас сүзгілер арасындағы айырмашылықты анықтаңдар. Нәтижесін 11.1-кестесіне жазыңдар.

11.1- кесте

Қарапайым сүзгі		Құрамдас сүзгі	
Шарты	Ерекшелігі	Шарты	Ерекшелігі

Рефлексия:

Менің білгенім Енді мен ... қолдана аламын.

§13 Мәліметтер қорымен жұмыс істеу

Бүгінгі сабақта:

- ▶ электрондық кестеде мәліметтер қорын құру;
- ▶ мәліметтерді іздеуді, сұрыптауды және сүзуді жүзеге асыруды үйренесіңдер.

Тірек ұғымдар:

- ▶ мәліметтер қоры;
- ▶ өріс;
- ▶ критерий;
- ▶ функция;
- ▶ функция аргументі.

MS Excel бағдарламасында мәліметтер қорларымен жұмыс істегенде, функциялар, тізімдер немесе мәліметтер қорының деректерін талдау үшін пайдаланылады. *МҚ функциясы* деп жалпы атауы бар әрбір функцияның үш аргументі пайдаланылады:

- мәліметтер қоры;
- өріс;
- критерийлер (талаптар).

Бұл аргументтер осы функциямен пайдаланылатын жұмыс парағындағы ұяшық аралықтарына сілтеме жасайды.

MS Excel бағдарламасында мәліметтер қорларымен жұмыс істеуге арналған 13 функция бар.



Мәліметтер қорының синтаксисі:
 МҚ функциясы (мәліметтер қоры; өріс; критерийлер (та-
 лаптар)).

Критерий деп берілген шарттарды қамтитын ұяшықтар аралығын айтады.

Төменде деректер қорының кейбір функцияларын келтірілген:

- ДСРЗНАЧ функциясы — тізім бағанындағы мәндердің орташа мәнін табады;
- БСЧЕТ функциясы — тізім бағанындағы ұяшықтар санын есептейді;
- БСЧЕТА функциясы — тізімнің бағанындағы бос емес ұяшықтардың санын есептейді.

Практикалық тапсырмаларды орындаған кезде басқа да функциялармен танысасыздар.

Практикалық жұмыс

А деңгейі

13.1-кестені толтырындар. Тапсырманы орындау үшін интернет ресурстарын пайдаланындар.

13.1- кесте

МҚ функциялар атауы	МҚ функциялардың тағайындалуы
ДСРЗНАЧ	
БСЧЕТ	
БСЧЕТА	
БІЗВЛЕЧЬ	
ДМАКС	
ДМІН	
БДПРОИЗВЕД	
ДСТАНДОТКЛ	
ДСТАНДОТКЛП	
БДСУММ	
БДДІСП	
БДДІСПП	

В деңгейі

Берілген мәліметтер қорын алфавиттік ретпен сұрыптандар:

	A	B	C	D
1	Аты-жөні	Еңбекақы, теңге	Жасы	Бөлім
2	Ахметов А.А.	150 000	45	бухгалтерия
3	Смирнов Е.В.	120 000	28	өндіріс
4	Жакеев С.К.	100 000	21	бухгалтерия
5	Еремеева С.С.	180 000	50	қойма
6	Болатбек Е.С.	100 000	23	өндіріс
7	Низамутдинов В.К.	100 000	24	өндіріс
8	Жабагин С.М.	120 000	29	бухгалтерия
9	Гордиянова Г.В.	120 000	33	қойма

13.2-сурет

Жазбаларды сұрыптандар:

1. “Аты-жөні” өрісін өсу ретімен сұрыптандар.
2. “Еңбекақы” өрісі бойынша, ал содан кейін “Жасы” өрісі бойынша кему ретімен сұрыптандар.
3. “Еңбекақы”, “Жасы” және “Бөлім” өрістерін өсу ретімен сұрыптандар.

С деңгейі

Келесі шарт қанша жазбаларда орындалады:

“Бөлім” = Есепші және “Жасы” < 40?

Рефлексия:

Менің білгенім Енді мен ... қолдана аламын.

4-
ТАРАУPython бағдарламалау
тілінде бағдарламалау
алгоритмі

Бұл тараудан білесіңдер:

- ▶ бағдарламалаудағы бірөлшемді массивтер;
- ▶ бірөлшемді массивтің қосымша әдістері;
- ▶ массивте элементтерді іздеу;
- ▶ массивте элементтерді іздеу және орнын ауыстыру.

Мынаны үйренесіңдер:

- ▶ бірөлшемді массивті пайдалану;
- ▶ бірөлшемді массивті пайдаланып есептер шығаруды.

§14 Бірөлшемді массив

Бүгінгі сабақта:

- ▶ Бірөлшемді массивтерді пайдалана отырып, Python бағдарламалау тілінде бағдарламалар жасауды үйренесіңдер.

Тірек ұғымдар:

- ▶ массив;
- ▶ айнымалы;
- ▶ индекс;
- ▶ массив элементі.

Қазіргі компьютерлердің негізгі мақсаты — көптеген деректерді өңдеу. Бұл ретте мыңдаған (немесе миллиондаған) ұяшықтағы деректердің әрқайсысына хабарласу керек. Әрбір ұяшыққа өз атын беру өте қиын әрі шатастырмау қажет. Бұл жағдайда жеке ұяшыққа емес, әрбір ұяшықтың жеке нөмірі бар ұяшықтар тобына атау береді. Бұл жады аймағы *массив* (немесе кесте) деп аталады.



Массив — ортақ аты бар жадта (көрші ұяшықтарда) орналасқан бір түрдегі айнымалы топ. Массивтегі әрбір ұяшықтың бірегей нөмірі бар.

Массивтермен жұмыс істеу үшін ең алдымен мыналарды үйрену керек:

- қажетті көлемді жақты массивте көрсету;
- деректерді қажетті ұяшыққа жазу;
- массив ұяшығынан деректерді оқу.

Python тілінде “массив” сияқты құрылым жоқ. Оның орнына бір типті нысандар тобын сақтау үшін тізімдер (list деректер түрі) пайдаланылады.

Python-тізімі — бұл әртүрлі нөмірі (индексі) бар элементтер жиынтығы және нөмірлеу әрдайым нөлден басталады.

Бағдарламалау тілдерінің көпшілігінде, қарапайым массивтерге қарағанда, тізім — динамикалық құрылымда, оның өлшемін бағдарламаны орындау кезінде өзгертуге болады (элементтерді жою және қосу).

Күн сайын компьютерге орнатылған қолданбалы қосымшалар арқылы секундына миллионнан астам деректерді өңдейсіңдер. Мысалы, “Карталар” қосымшасында бізге қолайлы маршрутты (жолды) іздеу барысында қандай жолдар жақын екенін, олардың қалай қиылысатынын, қажетті ғимараттың қай жерде орналасқанын нұсқайтын көптеген айнымалылар қолданылады. Осы айнымалы мәндерді ескере отырып, “Карта” қосымшасы қолайлы маршрутты (жолды) табады (14.1-сурет).



14.1-сурет. “Карта” қосымшасы



Компьютерлердегі қосымшалар көптеген деректермен қалай жылдам жұмыс істейді?

Бағдарламалау кезінде кейбір мәндерді сақтау үшін айнымалылар қолданылады. Мысалы, Python-да:

```
name = "Adilhan"
```

Әділхан есімін *name* айнымалысына сақтаймыз. Бағдарламалау кезінде кейбір мәндерді кейін қолдану үшін сақтайды. Егер бір мақсатпен сақталған мәндер көп болса, оларды қалай сақтауға болады? Бағдарламауға әрбір мән үшін жеке айнымалы құру қиын емес пе? Мысалы, егер сендер 5 атауды пайдаланатын болсаңдар, онда:

```
name1 = "Adilhan"
name2 = "Dias"
name3 = "Dariga"
name4 = "Nurbahyt"
name5 = "Aruzhan"
```

Егер тізім өсе түссе, әрбір жана атау үшін жеке айнымалы құру керек. Компьютерлер үлкен, кейде тіпті, миллиардтаған мәндермен жұмыс істейді.

Мұндай қиындықтарды болдырмау үшін, бағдарламалауда деректер массивін жасау мүмкіндігі бар. Массив бір типтегі немесе бір мақсаттағы мәндердің жиынын сақтайды.

Мысалы, егер сендер бес атаудан тұратын мәндерді сақтасаңдар келсе, онда:

```
names = ["Adilhan", "Dias", "Dariga", "Nurbahyt", "Aruzhan"]
```

Осылайша бес атаудан тұратын массив құрасыңдар. Python бағдарламалау тілінде, айнымалылар мәтін, сандық мән және нысан сияқты деректердің түрлерін сақтай алады. Жоғарыда көрсетілген жағдайда *names* айнымалысы Python-да list/тізім (список) деген атпен белгілі массив нысандарын сақтайды.



Массив/Тізім `[]` жақшаларымен белгіленеді. Барлық тік жақшалардың `[]` ішіндегі мәндер массивтің элементтері болып табылады.

Python-дағы массив элементтерін көрсету үшін *print* командасы пайдаланылады. Мысалы:

```
names = ["Adilhan", "Dias", "Dariga", "Nurbahyt", "Aruzhan"]
print(names)
```


Нәтиже:

```
["Adilhan", "Dias", "Dariga", "Nurbahyt", "Aruzhan"]
```



Әрбір массив элементінің жеке мекенжайы, сонымен қатар оны көрсету немесе пайдалану үшін элемент индексі болады. Егер массив құрған болсақ, онда тік жақшаларды одан әрі пайдалану массивтің кейбір индекс элементтерін қолдануға болатынын көрсетеді.

Массив элементтерінің индексі нөлден басталады. Индексті массивті пайдалану үшін мысалы:

```
names = ["Adilhan", "Dias", "Dariga", "Nurbahyt", "Aruzhan"]
print(names[0])
```

Нәтиже:

Adilhan

Жоғарыдағы мысалда *names* массивінің бірінші атауы қалай бейнеленетінін көрдіңдер.

Мысалы:

```
names = ["Adilhan", "Dias", "Dariga", "Nurbahyt", "Aruzhan"]
starosta = names[2]
print(starosta)
```

Нәтиже:

Dariga

Жоғарыда келтірілген мысалдан массивтің үшінші атауын *starosta* айнымалысында сақтадыңдар. Содан кейін осы айнымалының мәнін көрсетейік.

Мысалы:

```
names = ["Adilhan", "Dias", "Dariga", "Nurbahyt", "Aruzhan"]
names[3] = "Daniyar"
print(names)
```

Нәтиже:

```
["Adilhan", "Dias", "Dariga", "Daniyar", "Aruzhan"]
```

Жоғарыда келтірілген мысалда массивтің төртінші элементінің мәнін Даниярға ауыстырдық.

БҰЛ ҚЫЗЫҚ!

Python бағдарламалау тілінде бір индекс мәнінің орнына мәндер ауқымын пайдалануға болады. Мысалы, 2-ден 5-ке дейінгі элементтерді көрсету керек болса, онда келесідей жазамыз:

```
print(names[1:4])
```

Егер 3-тен соңғы мәнге дейін көрсеткілерің келсе, келесідей жазамыз:

```
print(names[2:])
```

4-атаудың алдындағы барлық атауларды көрсету үшін:

```
print(names[:3])
```

Сондай-ақ Python бағдарламалау тілінде теріс мәнді индексті көрсетуге болады. Бұл жағдайда ол массивтің соңындағы элементтерден бастап көрсетеді. Мысалы:

```
print(names[-1])
```



Бақылау сұрақтары



1. Массивті қалай құрады?
2. Массив не үшін қажет?
3. Массивтің кейбір элементін қалай көрсетуге болады?
4. Басқа мән беру арқылы массив элементін қалай өзгертуге болады?

Практикум

А деңгейі

1. Еліміздегі кез келген 6 қаланың атын массивке енгізіндер.
2. Массивті көрсетіндер.
3. Массивтен екінші қаланы көрсетіндер.
4. Массивтен соңғы қаланы көрсетіндер.
5. *Нұр-Сұлтан* деген айнымалы құрып, сол массивке атын сақтаңдар.

B деңгейі

1. 6 саннан тұратын массив құрыңдар.
2. Массивтегі бірдей сандарды көрсетіңдер.

C деңгейі

1. Массивтегі бірінші және соңғы санның орындарын ауыстырыңдар.
2. Массивке тағы бір сан қосыңдар.

Рефлексия:

Менің білгенім Енді мен ... қолдана аламын.

§15–16**Белгіленген сипаттары бар элементті іздеу***Бүгінгі сабақта:*

- бағдарламаны өңдеудің кіріктірілген ортасында бірөлшемді массивті қолдана отырып бағдарламаны құрастыруды үйренесіңдер

Тірек ұғымдар:

- элемент;
- массив;
- айнымалы.

*Мәліметтер қорында іздеуді қалай орындайды?*

Массивте элемент іздеу. Қосымшаны бағдарламалау кезінде программистерде деректер массивінде белгілі бір мәндерді табу туралы сұрақ туындайды. Мысалы, пайдаланушыны белгілі логинінің көмегімен немесе туған күндерінің сәйкестігі бойынша анықтау.

Массивтегі іздеудің логикалық іске асуын қарастырайық. Мысалы, келесі бағдарламаны жазып, сандар массивінде 3 санын табайық.

Біріншіден, деректер массивін жариялайсыңдар:

```
a = []
```

Деректердің бос массивін құрып, содан кейін оны сандармен толтырасыңдар:

```
a += [ int (i) for i in input().split() ]
```

Бірөлшемді массивте іздеуді орындау үшін ізделінетін санды массивтің барлық сандарымен салыстыру жеткілікті :

```
for i in a:
    If i == 3:
        print("YES")
        break
```

Бұл мысалда егер 3 саны массивте бар болса, “YES” жауабы шығады.

Алдыңғы сабақта талқылағанмыздай, Python-да программистің жұмысын жеңілдететін көптеген дайын функциялар бар. Олардың бірі — массивтің элементін іздеу функциясы. 3 саны массивте бар екенін іздеуді келесідей ретте іске асыруға болады:

```
if 3 in a:
    print("YES")
```

“3 in a” өрнегі *a* массивінде 3 саны бар немесе жоқ екенін көрсетеді. Және бұл өрнек егер дұрыс болса true, жалған болса false мәндерін қайтарады.

Егер берілген санның массивте қай орында тұрғанын анықтау үшін келсе, `index()` функциясы қолданылады. Бұл функция егер іздеп отырған элемент массивте бар болса, оның индексін, әйтпесе қателікті қайтарады. Мысалы,

```
a = [int (i) for i in input().split() ]
if 3 in a:
    print ( a.index(3) )
```

Бағдарлама қателіктің қайталануына жол бермеу үшін алдымен массивте берілген санның бар-жоғын тексеріп , содан кейін осы элементтің индексін массивтен шығарады .

Бақылау сұрақтары



1. Массивте элементті логикалық түрде қалай іздеуге болады?
2. Массивте элементтің бар-жоғын тексеру Python-да қандай әдіспен іске асырылады?
3. Элемент массивте бар болса, элемент индексін қайтару қалай жүзеге асырылады?

Есеп шығару мысалы

N сандарының массиві берілген. Массивтен ең үлкен элементті табындар. Бірінші жолда N саны, ал екінші жолда N сандар берілген.

Енгізу	Нәтиже
8	7
1 6 5 7 3 2 4 5	

Алдымен N санын енгізіндер:

```
n = int( input () )
```

Содан кейін, N санынан тұратын массив құрындар:

```
a = [ int (i) for i in input().split() ]
```

Сол сияқты массивтен логикалық түрде анықталған және ең максимал элементті (максимал) іздейік. Ол үшін ең максимал элементті сақтайтын mx айнымалыны құрындар. Бірінші кадамда массивтің бірінші элементінің мәнін максимал ретінде құрылған mx айнымалыда сақтандар:

```
mx = a[0]
for i in a:
    if i > mx:
        mx = i
print(mx)
```

Осылайша егер mx айнымалысының ағымдағы мәнінен үлкен элемент тапсақ, онда mx мәнін үлкеніне өзгертіндер.



Python-да массивтің ең максимал (максимум) және ең минимал (минимум) мәндерін табуға арналған стандартты функциялар бар: `max` және `min`. Мысалы:

```
print (max(a))
```

a сандар массивінің ең максимал (максимум) мәнін басып шығарады:

```
print (min(a)),
```

a сандар массивінің ең минимал (минимум) мәнін басып шығарады.

Сол сияқты массив сандарының қосындысын есептейтін `sum` функциясы бар.

Практикалық жұмыс

А деңгейі

Келесі бағдарламаны құрындар:

1. Бірінші кіріс жолына N санын енгізіндер .
 2. Екінші кіріс жолына N сандарын енгізу арқылы массив құрындар .
 3. Үшінші кіріс жолына K санын енгізіндер .
 4. Массивте K санын іздеңдер . Егер бұл сан массивте болса, “YES”, әйтпесе “NO” сөздерін басып шығарындар .
- Егер K саны екінші шығыс жолында болса, массивте қанша сан бар екенін басып шығарындар .

1-мысал.

Енгізу	Нәтиже
5	NO
3 4 6 7 10	
5	

2-мысал.

Енгізу	Нәтиже
6	YES
10 12 3 4 3 5	2
3	

В деңгейі

Массивтен K санына ең жақын санды іздейтін бағдарламаны құрындар. Бірінші жолда N саны, екінші жолда N сандары, үшінші жолда K саны берілген. Егер массивте K саны жоқ болса, онда массивтің соңына K санын қосу арқылы массивтің ұзындығын басып шығарындар.

1-мысал.

Енгізу	Нәтиже
5	0
4 3 2 4 5	
4	

2-мысал.

Енгізу	Нәтиже
6	3
1 2 3 4 5 6	
4	

3-мысал.

Енгізу	Нәтиже
6	6
1 2 3 4 5 6	
7	

С деңгейі

Массивте K санының соңғы позициясын іздейтін бағдарлама құрындар. Бірінші жолда N саны, ал екінші жолда N сандары, ал үшінші жолда K саны берілген. Егер массивте K саны жоқ болса, онда массивтің соңына K санын қосу арқылы массивтің ұзындығын басып шығарындар.

1-мысал.

Енгізу	Нәтиже
5	3
4 3 2 4 5	
4	

2-мысал.

Енгізу	Нәтиже
6	4
1 2 3 3 3 6	
3	

3-мысал.

Енгізу	Нәтиже
6	6
1 2 3 4 5 6	
7	

Қосымша тапсырма

Есеп. Әділхан Диасқа 1-ден 100-ге дейінгі сандардың біреуін жасырды. Диас Әділханға жауаптың нұсқаларын ұсынды. Әділхан әр жолы жауаптарды көп немесе аз дейді. Осы жағдайда, Диас жауапты қашан және қалай беретінін анықтандар?

Алгоритмді Python-ды бағдарламалау тілінде орындандар.

Рефлексия:

Менің білгенім Енді мен ... қолдана аламын.

§17 Элементтердің орнын ауыстыру

Бүгінгі сабақта:

- ▶ массивте элементтердің орнын алмастыру әдістерін үйренесіңдер.

Тірек ұғымдар:

- ▶ массив;
- ▶ айнымалы.



Массивте элементтердің орнын алмастыруды қалай жүзеге асыруға болады?

Программистер жиі пайдаланатын операциялардың бірі — массив ішіндегі элементтердің орнын алмастыру. Бұл операция массивте элементтерді сұрыптау кезінде жиі қолданылады.

Массивте элементтердің орнын алмастыру мысалдарын қарастырайық. 1-мысал.

```
numbers = [ 3, 4, 10, 23 ,21, 15 ]
```

Үшінші элемент пен бесінші элементтің орнын ауыстырсақ, соған сәйкес айнымалылардың мәндерін ауыстыру жүзеге асырылады. Уақытша сақтау функциясын орындайтын үшінші айнымалы пайдаланылады:

```
num1 = 5
num2 =10
temp = num1
num1 = num2
num2 = temp
```

Тек осы жағдайда ғана массивтің индекстері мен элементтерін пайдаланасыңдар:

```
temp = numbers[2]
numbers[2] = numbers[4]
numbers[4] = temp
```

Python бағдарламалау тілінде массив элементтерін кері тәртіпте қайта реттейтін функцияны *reverse ()* деп атайды.

Мысалы, келесі сандар массивін алыңдар:

```
numbers = [ 1, 32, 21, 34, 4 , 23]
```

Енді *reverse ()* функциясын қолданып массивті басып шығарайық:


```
numbers.reverse()
for i in numbers:
    print(i, end=" ")
```

2-мысал. N саннан тұратын массив берілген. Массивтің барлық элементтерінің оң жағына циклдік ауысуын орындау керек.

Енгізу	Нәтиже
6	4 5 6 8 7 9
9 4 5 6 8 7	

Алдымен N саннан тұратын массивті келесідей құрындар:

```
N = int( input() )
nums = [ int(i) for i in input().split() ]
```

Енді циклді пайдаланып, массивтің көрші элементтерінің орындарын алмастырайық. Оны массивтің соңынан бастаймыз. Демек, бізге массивтегі соңғы элементті көрші элементтермен алмастыру арқылы жылжыту керек:

```
for i in range( N-1, 0, -1 ):
    temp = nums[ i-1 ]
    nums[ i-1 ] = nums[ i ]
    nums[ i ] = temp
```

Енді массивті басып шығарындар:

```
for i in nums:
    print(i, end=" ")
```

Бақылау сұрақтары



1. Массивте элементтердің орнын алмастыруды қалай жүзеге асыруға болады?
2. Python бағдарламалау тілінде қандай функцияның элементтерін кері тәртіппен қайта реттейді?

Практикалық жұмыс

А деңгейі

1. N саннан тұратын массив берілген. `Reverse()` функциясын пайдаланбай, массив элементтерін кері ретпен қайта құрындар.

Енгізу	Нәтиже
7	8 9 7 3 2 6 4
4 6 2 3 7 9 8	

2. N саннан тұратын массив берілген. Максимал және минимал элементтердің орнын ауыстырындар. Массивті бір жолға шығарындар.

В деңгейі

1. N саннан тұратын массив берілген. Массивте элементтерді K элементке оң жаққа жылжытып, оны басып шығарындар. Бірінші жолда N саны, екінші жолда N сандары берілген. Соңғысы K саны.

1-мысал.

Енгізу	Нәтиже
6	2 8 9 4 6 3
4 6 3 2 8 9	
3	

2-мысал.

Енгізу	Нәтиже
8	4 10 23 45 78 1 2 3
23 45 78 1 2 3 4 10	
2	

2. Келесі бағдарламаны жазып, N сандарының массивін құрындар. Содан кейін массивтегі әрбір көршілес элементтердің төмендегідей орнын алмастырындар:

($A[0]$ -ді $A[1]$ -мен, $A[2]$ -ні $A[3]$ -пен және әрі қарай т.с.с.).

Енгізу	Нәтиже
8	5 7 4 6 8 7 1 9
7 5 6 4 7 8 9 1	

С деңгейі

1. Алдыңғы есепті элементтердің орнын алмастыру кезінде қосымша айнымалыны пайдаланбай қайта шешіндер.

Мысал.

Енгізу	Нәтиже
8	5 7 4 6 8 7 1 9
7 5 6 4 7 8 9 1	

2. Бүтін сандардың тізімі берілген. Нөлдік емес элементтердің ретін өзгертпей тізімнің сол жағына, ал барлық нөлдік элементтерді оң жағына орналастыру арқылы тізімді “ығыстыру” қажет. Нөлдік емес элементтердің орналасу ретін өзгертуге және қосымша тізім пайдалануға болмайды.

Енгізу

4 0 5 0 3 0 0 5

Нәтиже

4 5 3 5 0 0 0 0

Рефлексия:

Менің білгенім Енді мен ... қолдана аламын.

§18–19 Екіөлшемді массив

Бүгінгі сабақта:

- ▶ Python бағдарламалау тілінде екіөлшемді массивті қолданып, программа жазуды;
- ▶ массивте итерация қолдануды үйренесіңдер.

Тірек ұғымдар:

- ▶ екіөлшемді массив;
- ▶ матрица;
- ▶ итерация.



Екіөлшемді массив дегеніміз не?

Екіөлшемді массивті кесте түрінде ұсынуға болады. Кестенің әрбір ұяшығында жолдың және бағанның нөмірі бар. Демек, әрбір ұяшық тізім сияқты бір нөмір емес, екі нөмірге ие: жол нөмірі және баған нөмірі. Мұндай тізімдер *екіөлшемді* деп аталады. Ал математикада мұндай құрылым “*матрицалар*” деп аталады.



Массив өлшемі — массив шегінде элементті бірімәнді индекске қажетті индекстер саны.

4 x 3 өлшемді матрица мысалын қарастырайық:

```
matrix = [[-1, 0, 1],
          [-1, 0, 1],
          [0, 1, -1],
          [1, 1, -1]]
```

Бұл операторды бір жолға жазуға болады:

```
matrix = [[-1, 0, 1], [-1, 0, 1], [0, 1, -1], [1, 1, -1]]
```

Екіөлшемді массивті бір оператормен шығаруға болады:

```
print(matrix)
```

Осы тәсілдің кемшілігі — элементтерді алдын ала өңдейтін мүмкіндіктің болмауы.

Нәтиже: `[[-1, 0, 1], [-1, 0, 1], [0, 1, -1], [1, 1, -1]]`



Нақты элементке қалай жазуға болады?

Мысалдар қарастырайық:

```
a = [[1, 2, 3], [4, 5, 6]]
print(a[0])
print(a[1])
print(a[0][0], a[0][1], a[0][2])
```

Нәтиже:

```
C:\Programs\Python\Python36-32\python.exe
[1, 2, 3]
[4, 5, 6]
1 2 3
↔
Process finished with exit code 0
```

Келесі мысалды қарастырайық.

```
a = [[1, 2, 3], [4, 5, 6]]
print(a[0])
print(a[1])
print(a[0][0], a[0][1], a[0][2])
print(len(a))
print(len(a[0]))
```

Бұл мысалда `a[0]` — бірінші жол, `len(a)` функциясы — жолдар саны, `len(a[0])` — бірінші жолдағы элементтер (бағандар) саны.

Нәтиже:

```
C:\Programs\Python\Python36-32\python.exe
[1, 2, 3]
[4, 5, 6]
1 2 3
2
3
Process finished with exit code 0
```

Тізімді өңдеу мен шығару үшін салынған екі цикл қолданылады. Бірінші цикл жол нөмірі бойынша, екінші цикл жолдың ішіндегі элементтер бойынша. Мысалы, сандарды бір жолдың ішінде бос орындармен бөліп, екіөлшемді сандық тізімді экранға жолдап шығаруға болады:

```
for i in range(len(A)):
    for j in range(len(A[i])):
        print(A[i][j], end = ' ')
    print()
```

Демек, циклдер индексі бойынша емес, тізім мәні бойынша қолданылады:

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

Сонымен қатар бір жолды шығару үшін **join** әдісін қолдануға болады:

```
join:
for row in A:
    print(' '.join(list(map(str, row))))
```

Әрине, массивті қолмен толтыру және шығару қолайсыз. Массивті кездейсоқ сандармен толтыру үшін келесі алгоритм қолданылады:

```
1 import random
2 for i in range(N):
3     for j in range(M):
4         matrix[i][j] = random.randint ( 30, 60 )
5         print ( «{:4d}».format(matrix[i][j]), end = "" )
```



Екіөлшемді массив элементтерін белгілеу бірөлшемді массив элементтері сияқты нөлден басталады.

Мысалы, `matrix[2][3]` — төртінші баған үшінші жолдың элементі.

Келесі мысалды қарастырайық. 10 мен 99 сандарының аралығындағы сандарды толтырып, экранға шығарыңдар:

```
a = [] # бос тізім құрастырыңдар
    k = 10 # бастапқы мән, кез келген болуы мүмкін
for r in range(9): # 9 жол
    a.append([]) # бос жол құрастырыңдар
```

```

for c in range(10): # әрбір жолда 10 элемент
a[r].append(k) # кезекті элементті жолға қосындар
k += 1 # есептеуіш мәнін арттырындар
for r in a:
print(r)

```

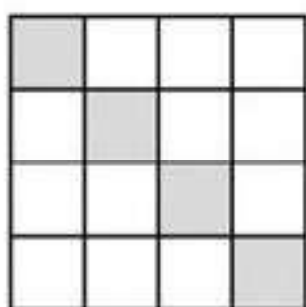
Нәтиже:

```

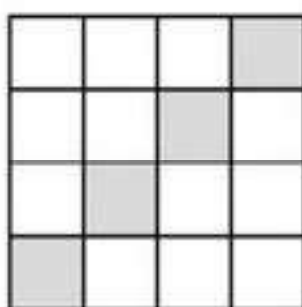
C:\Programs\Python\Python36-32\python.exe
[10, 11, 12, 13, 14, 15, 16, 17, 18, 19]
[20, 21, 22, 23, 24, 25, 26, 27, 28, 29]
[30, 31, 32, 33, 34, 35, 36, 37, 38, 39]
[40, 41, 42, 43, 44, 45, 46, 47, 48, 49]
[50, 51, 52, 53, 54, 55, 56, 57, 58, 59]
[60, 61, 62, 63, 64, 65, 66, 67, 68, 69]
[70, 71, 72, 73, 74, 75, 76, 77, 78, 79]
[80, 81, 82, 83, 84, 85, 86, 87, 88, 89]
[90, 91, 92, 93, 94, 95, 96, 97, 98, 99]

```

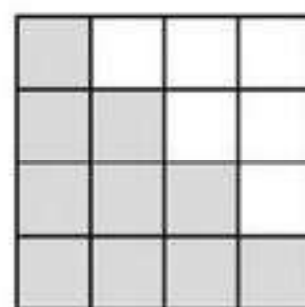
Матрица элементтерін өңдеу. N жолдар мен N бағандардан тұратын A квадрат матрицасының кейбір элементтерін қалай өңдеуге (мысалы, қосу) болатынын қарастырайық. 18.1, *a*-суретте матрицаның басты диагоналі, 18.1, *ә*-суретте екінші (жанама) диагоналі, 18.1, *б*-суретте басты диагональ және оның астында барлық элементтер көрсетілген.



a)



ә)



б)

18.1-сурет

Басты диагональ — $A[0,0]$, $A[1,1]$, ..., $A[N-1, N-1]$ элементтері, яғни жол нөмірі баған нөміріне тең. Бұл элементтерді таңдау үшін бір цикл қажет:

```

for i in range(N):
# A[i][i] жұмыс істейміз.

```

Жанама диагональ элементтері — $A[0, N-1], A[1, N-2], \dots, A[N-1, 0]$.
 Әрбір элемент үшін жол нөмірі мен бағанның сомасы $N-1$ тен, сондықтан таңдау циклін аламыз:

```
for i in range(N):
#  $A[i][N-1-i]$  жұмыс істейміз
```

Осы жағдайда (барлық элементтерді басты диагональда және оның астында өңдеу) салынған цикл қажет: жол нөмірі 0-ден $N-1$ -ге дейін, ал әрбір жол үшін баған нөмірі i -0-ден i -ге дейін өзгереді:

```
for i in range(N):
for j in range(i+1):
#  $A[i][j]$  жұмыс істейміз
```

Матрица бағандарын ауыстыру үшін бір цикл жеткілікті. Мысалы, 2 және 4 бағандарын қосалқы c айнымалысымен ауыстырамыз:

```
for i in range(N):
c = A[i][2]
A[i][2] = A[i][4]
A[i][4] = c
```

немесе Python мүмкіндіктерін қолданамыз:

```
for i in range(N):
A[i][2], A[i][4] = A[i][4], A[i][2]
```

$A[i]$ — бұл i жол элементтерінің тізіміне сілтемесі екенін ескере отырып, екі жолды циклсіз ауыстыруға болады. Сондықтан сілтемелерді ауыстыру жеткілікті. Оператор

```
A[i], A[j] = A[j], A[i]
```

i және j нөмірлері бар матрица жолдарын ауыстырады.

i нөмірлі жолдың көшірмесін жасау үшін бұлай жасауға болмайды:

```
R = A[i]
```

Оның орнына жады көшірмесін жасау керек:

```
R = A[i][:]
```

j нөмірі бар бағанның көшірмесін салу біршама қиын, себебі матрица жадында жолдар бойынша орналасқан. Бұл тапсырмада генераторды пайдалану ыңғайлы:

```
C = [ row[j] for row in A ]
```

Циклде A матрицасының барлық жолдары іріктеледі, олар кезекпен айнымалы `row`-ға түседі. Генератор әрбір жолдан j нөмірімен элементті тандайды және осы мәндердің тізімін жасайды.

Генератордың көмегімен басты диагональ элементтерін жеке массивке оңай бөлуге болады:

$D = [A[i][i] \text{ for } i \text{ in range}(N)]$

Мұндағы A матрицасы N жолдар мен N бағандардан тұрады деп аламыз.

Бақылау сұрақтары



1. Екіөлшемді массив дегеніміз не?
2. Екіөлшемді массивті қалай жазады ?
3. Егер екі массивті біріктірсек не болады?

Практикалық жұмыс

Қазақстанның әртүрлі аймақтарында орналасқан үш метеостанциядан 4 күнгі ауа температурасының көрсеткіштері алынды.

Станция нөмірі	1-күн	2-күн	3-күн	4-күн
1	-8	-14	-19	-18
2	25	28	26	20
3	11	18	20	25

Демек, көрсеткіштердің жазбасы екіөлшемді массивте келесі түрде ұсынылады:

$t[1,1] = -8;$ $t[1,2] = -14;$ $t[1,3] = -19;$ $t[1,4] = -18;$
 $t[2,1] = 25;$ $t[2,2] = 28;$ $t[2,3] = 26;$ $t[2,4] = 20;$
 $t[3,1] = 11;$ $t[3,2] = 18;$ $t[3,3] = 20;$ $t[3,4] = 25.$

А деңгейі

1. Экранға 2-метеостанцияның 4-күнгі және 3-метеостанцияның 1-күнгі температурасын шығарындар.
2. Бос массив құрындар.
3. Массивті n қала атауымен толтырындар.
4. Массивтен әрбір екінші қаланы көрсетіндер (басып шығарындар).

В деңгейі

1. 2-күнгі барлық метеостанциялардың термометр көрсеткіштерін экранға шығарындар.
2. 3-метеостанциядағы орташа температураны анықтаңдар.
3. N саннан тұратын массив берілген. Бірінші жолда N саны, келесі N жолдарында сандар беріледі. Осы массивтен барлық жұп сандарды басып шығарындар.

Мысал.

Енгізу:	Нәтиже:
6	4
4	2
2	10
3	
1	
10	
11	

С деңгейі

1. Ауа температурасы $24—26^{\circ}\text{C}$ жылы болған күндер мен метеостанцияларды экранға шығарындар.
2. N саннан тұратын массив берілген. Бірінші жолда N саны, ал келесі N жолдарында оң және теріс сандар беріледі. Осы массивтен барлық мәні нөлден кіші сандарды анықтаңдар.

Мысал.

Енгізу:	Нәтиже:
5	3
2	
-45	
-1	
23	
-12	

Рефлексия:

Менің білгенім Енді мен ... қолдана аламын.

§20–21

Сұрыптау

Бүгінгі сабақта:

- ▶ массивте элементтерді қалай сұрыптайтынын білесіңдер.

Тірек ұғымдар:

- ▶ массив;
- ▶ айнымалы.



Бағдарламада деректерді әрдайым алфавиттік тәртіпте сұрыптайды. Бағдарламалау кезінде сұрыптау қалай жүзеге асады?

Тізімдерде іздеген кезде барлық деректердің алфавит бойынша сұрыпталғаны іздеуді жеңілдетеді. Сұрыпталу үлкенінен кішісіне және керісінше жасалады. Берілген деректер сұрыпталған болса, оның ішінен қажетті элементті іздеуге белгілі бір жүйені оңай қолдануға болады. Деректермен жұмыс істегенде компьютерлер де олардың белгілі бір реттілікте болғанын қажет етеді.

Бағдарламалауда әртүрлі сұрыптау алгоритмдері бар және олардың кейбіреуін қарастырайық. Математика және информатика саласының ғалымы Дональд Эрвин Кнут барлық сұрыптау алгоритмдерін сипаттау үшін бағдарламалаудағы сұрыптау әдістеріне үлкен-үлкен томдар арнады. Осы әдістердің ең қарапайым түрін қарастырайық. Сонымен қатар *Python* бағдарламалау тілінің барынша жылдам сұрыптау әдістерін қолдануды үйренесіңдер.

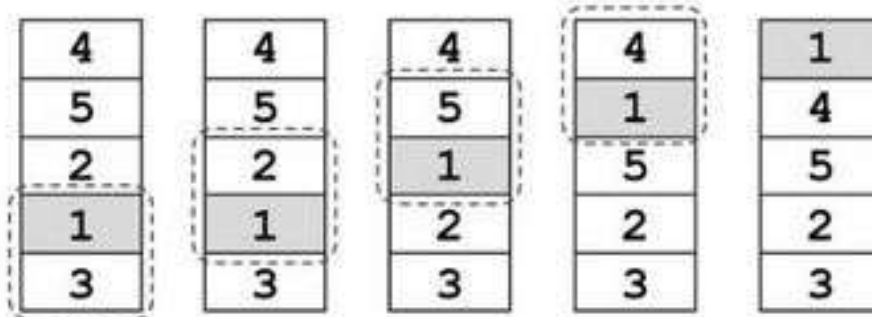
Көпіршікпен сұрыптау. Сұрыптауды түсіну мен іске асыруда ең қарапайым тәсілдердің бірі — *көпіршікпен сұрыптау*. Бүкіл алгоритмнің мәні, судағы көпіршіктің су бетіне ұмтылуы сияқты, әр жолда үлкен сандар массивтің жоғары орындарына ұмтылады. Ол келесі ретпен орындалады. 6 саннан тұратын массив берілген:

```
nums = [6, 3, 4, 1, 9, 8]
```

Енді екі циклдің көмегімен элементтерді жоғары көтере отырып массивтің әрбір элементінен өтейік:

```
for i in range(6):
    for j in range(5, i, -1):
        if nums[j-1] > nums[j]:
            temp = nums[j-1]
            nums[j-1] = nums[j]
            nums[j] = temp
```

Алдымен соңғы элементті алдыңғы элементпен салыстырайық (20.1-сурет). Егер олар дұрыс тұрмаса (аз элемент “төмен” тұрса), олардың орнын ауыстырамыз. Солайша 20.1-суреттегі жұп элементтерді және т.б. қарастырайық.



20.1-сурет. Көпіршікпен сұрыптауға мысал

Sorted() функциясы. Python-бағдарламалау тілінде программистің жұмысын жеңілдететін дайын функциялар бар. *Sorted ()* функциясы массивті өте жылдам уақытта сұрыптауға мүмкіндік береді. Бұл функцияның орындалуы келесідей.

N сандарының массиві берілген. Массивті сандармен толтырыңдар:

```
nums = [ int( i )
         for i in split().input() ]
```

Содан кейін, *nums* массивіне *sorted ()* функциясын қолданыңдар және массивті басып шығарыңдар:

```
nums.sorted()
for i in nums:
    print( i , end=" ")
```

Бақылау сұрақтары



1. Көпіршікпен сұрыптау қалай орындалады ?
2. Python бағдарламалау тілінде қандай функция массив элементтерін сұрыптайды ?

Практикалық жұмыс

А деңгейі

1. *N* сандарының массиві берілген . Бұл жиынды кему ретімен сұрыптап, басып шығарыңдар .

Мысал.

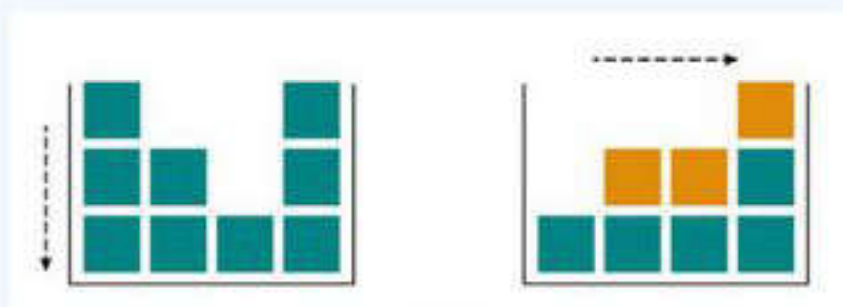
Енгізу	Нәтиже
6	8 7 6 4 3 2
3 2 4 8 6 7	

2. Қорапта қатарластырып жиналған ойыншық кубиктерінің *n* бағандары бар. Бастапқыда қораптың гравитациясы барлық кубиктерді төменге тартады. Гравитацияны ауыстыру үшін барлық кубиктерді қораптың оң жақ қабырғасына қарай тарту қажет. 20.2-суретте кубиктердің бастапқы және соңғы орналасуы көрсетілген: позицияларын өзгерткен кубиктер қызғылт сары түспен ерекшеленген.

Қораптағы ойыншық кубиктердің алғашқы орналасуы берілген. Гравитацияны ауыстырғаннан кейін *n* бағандарының әрқайсысындағы кубиктердің санын табатын бағдарламаны жазындар.

Мысал.

Енгізу	Нәтиже
4	1 2 2 3
3 2 1 2	



20.2-сурет

В деңгейі

1. Диасқа маңызды тапсырма берілді. Оған бағдарламауға жанадан қатысқандарды қолдау міндеті жүктелді.

Енді Диаста қолдау көрсетуге мұқтаж *n* олимпиадалық спортшылар бар. Әр ойыншының өз деңгейі бар. Егер бір программист үшін Олимпиадаға қатысушылардың ішінен одан деңгейі кем немесе артық бір ойыншы табылса, Диас сол программистке көмектесуді дұрыс деп санайды.

Диастың қанша қатысушыға қолдау көрсететіндігін есептейтін бағдарлама құрындар.

1-мысал.

Енгізу	Нәтиже
2	0
1 5	

2-мысал.

Енгізу	Нәтиже
3	1
1 2 5	

Ескерту :

1-мысалда Диас 1-деңгейлі Олимпиада қатысушыларына қолдау көрсете алмайды, себебі рейтингісі 1-ден кем Олимпиада ойыншысы жоқ. Сондай-ақ оның 5-деңгейлі Олимпиада қатысушыларына да қолдау көрсете алмайды, себебі рейтингісі 5-тен артық Олимпиада ойыншысы жоқ.

2-мысалда Диас 2-деңгейлі Олимпиада қатысушысына қолдау көрсете алады, себебі деңгейі 2-ден кем және артық Олимпиадаға қатысушы бар.

2. Массивті өсу ретімен сұрыптау барысында *көпіршік* сұрыптау алгоритмі қанша рет алмастыру жасайтынын анықтаңдар.

1-мысал.

Енгізу	Нәтиже
5	0
1 2 3 4 5	

2-мысал.

Енгізу	Нәтиже
5	10
5 4 3 2 1	

С деңгейі

1. Алдиярда көлемі a_1, a_2, \dots, a_n гигабайт n флэш-карталар және m гигабайт үлкен көлемді файл бар.

Егер Алдияр өзінің файлы ерікті өлшеммен бірнеше бөлікке бөліп сақтағысы келсе, оған ең кем дегенде қанша флэш-карталар қажет болатынын табыңдар.

Бірінші жолда n , екіншіде m , үшіншіде n флэш-карталардың сандары берілген.

1-мысал.

Енгізу	Нәтиже
3	2
5	
2 1 3	

2-мысал.

Енгізу	Нәтиже
3	3
6	
2 3 2	

Ескерту :

1-мысалда Алдиярға, бірінші және үшінші 2 флеш-карталары керек.

2-мысалда Алдиярға барлық 3 флеш-карталары кажет.

2. Ох түзудің бойында орналасқан мемлекеттерде үш дос тұрады. Біріншісі x_1 , екіншісі x_2 , ал үшінші x_3 нүктесінде. Жаңа жылды бірге қарсы алу үшін оларға бірігіп, ең аз дегенде қанша қашықтық жүріп өту керек екенін есептейтін бағдарламаны жазыңдар.

1-мысал.

Енгізу	Нәтиже
7 1 4	6

2-мысал.

Енгізу	Нәтиже
20 30 10	20

Рефлексия:

Менің білгенім Енді мен ... қолдана аламын.

§22–23**Элементті жою және кірістіру***Бүгінгі сабақта:*

- Бағдарламаны өңдеудің кіріктірілген ортасында бірелшемді массивті қолдана отырып бағдарламаны құрастыруды үйренесіңдер.

Тірек ұғымдар:

- массив;
- айнымалы;
- индекс.



Python-да массивті пайдалану арқылы бағдарламалаудың артықшылықтары қандай?

Python бағдарламалау тілінің кітапханасында көптеген дайын функциялар бар. Оларды пайдалану туралы толық ақпаратты <https://docs.python.org/> сайтынан оқып таныса аласындар. Бұл дайын функциялар бағдарламаның жұмысын айтарлықтай жеңілдетеді, сондықтан, Python-да бағдарламалау әлдеқайда жеңіл.

Index функциясы. *Index* функциясы массив элементінің индексін түзейді. Егер элемент жоқ болса, онда бағдарлама қатені түзетеді. Мысалы:

```
numbers = [ 4, 5, 34, 23, 121]
print( numbers.index(34) )
```

Нәтиже:

2

Append функциясы. Осы әдісті пайдалана отырып, массивке элемент қосу әдістерін қарастырайық:

```
numbers += [int (input() )]
```

Массивті кіріс санын қамтитын басқа массивке біріктіріп түрлендірейік.

Append функциясы массивтің соңына элементтерді қосуды жеңілдетеді. Ол келесідей орындалады:

```
numbers = [ 1, 2, 45, 3, 6, 4]
numbers.append( 5 )
print(numbers)
```

Нәтиже:

[1, 2, 45, 3, 6, 4, 5]

Осылайша біз массив соңына жаңа элементті қосамыз

Элементті енгізейік:

```
numbers.append( int (input () ) )
```

Сол сияқты *remove ()* функциясын пайдаланып, элементті массивтен алып тастаймыз.

Extend функциясы. Python бағдарламалау тіліндегі айнымалылар әртүрлі нысандарды сақтайды. Мысалы, ол массивті массив ішінде келесідей сақтауы мүмкін:

```
numbers = [4, 5, 2, [3, 4], 6]
```

Біз көріп отырғандай, 3-индекс пен [3,4] массиві сақталған. Осындай массивтің элементтеріне сілтеме жасау үшін келесі әдісті қолданайық:

```
print( number [3][0] )
```

Нәтиже:

3

Егер *append* функциясы массивтің соңына кез келген нысанды қосса, онда массивті де қосады. Мысалы:

```
numbers = [ 1,2,3 ]
numbers.append( [3,4] )
print(numbers)
```

Нәтиже:

[1, 2, 3, [3, 4]]

Extend функциясы алдымен массивті бөлек элементтерге ажыратады, содан кейін оларды массивке қосады.

Мысалы:

```
numbers = [1,2,3 ]
numbers.extend( [3,4] )
print(numbers)
```

Нәтиже:

[1, 2, 3, 3, 4]

Split функциясы. Бұл функция мәтінді жеке сөздерге бөледі және оларды массивте сақтайды. Мысалы:

```
text = "Hello world! I like coding"
text_new = text.split()
print(text_new)
```

Нәтиже:

["Hello", "world!", "I", "like", "coding"]

Жоғарыда келтірілген мысалдан *text* айнымалысы болғанын көрдік. Содан кейін *text_new* айнымалысын құрайық, сөздерді бөлек массив ретінде сақтайық.

Split функциясын сандық мәндерді бір жолға енгізгенде пайдалануға болады. Python тілі көбінесе веб-бағдарламалауда қолданылады. Бір жолдағы нысандарды Python мәтін ретінде қарастырады, бірақ *split* функциясын пайдаланып, мәтінді жеке сөздерге бөліп, мәтіннің типін сандық мәндерге ауыстырады. Мысалы:

```
numbers = [ int(i) for i in input().split() ]
```


Бақылау сұрақтары



1. Append және extend функцияларының арасында қандай айырмашылықтар бар?
2. Python-да сандарды бір жол ретінде енгізіп, содан кейін ол мәндердің сандық түрде сақталуын қалай орындауға болады?

Практикалық жұмыс

А деңгейі

1-тапсырма

1. N сандарының массивін құрындар.
2. Массивке сандарды қосу үшін *append* функциясын пайдаланындар.
3. Массив сандарының қосындысын есептеңдер.

2-тапсырма

1. Мәтінді сақтайтын массив құрындар.
2. Енгізілген мәтіндегі сөздердің санын есептеңдер.

В деңгейі

Бағдарламаны құрындар. Бір жолға берілген сандардың қосындысын есептеңдер.

Мысал.

Енгізу	Нәтиже
4 5 6 8 7	30

С деңгейі

Бағдарламаны құрындар. Бір жолға сандар түзілген. Көршілес элементтерден үлкенірек сандардың санын есептеңдер.

Мысал.

Енгізу	Нәтиже
1 3 2 4 3	2

Рефлексия:

Менің білгенім Енді мен ... қолдана аламын.

5- ТАРАУ

Python бағдарламалау тілінде 2D ойын жасау

Бұл тараудан білесіңдер:

- ▶ 2D ойынның жасауды;
- ▶ PyGame кітапханасы туралы;
- ▶ кейіпкерлерді анимациялауды.

Мынаны үйренесіңдер:

- ▶ PyGame кітапханасын қосуды;
- ▶ ойынның терезесін жасау үшін PyGame кітапханасының дайын модульдерін қолдануды;
- ▶ ойынның артқы фонын жасауды;
- ▶ дайын кейіпкерлерді ойынға жүктеуді;
- ▶ кейіпкердің қозғалысын программалауды;
- ▶ кейіпкерлерді пернетақтадан басқаруды;
- ▶ дайын сценарий бойынша ойын жасауды;
- ▶ ойын нәтижесін есептеу алгоритмін жүзеге асыруды.

§24–25 PyGame кітапханасы

Бүгінгі сабақта:

- ▶ 2D ойын жасауды;
- ▶ PyGame кітапханасын қосуды;
- ▶ ойынның терезесін жасау үшін PyGame кітапханасының дайын модульдерін қолдануды үйренесіңдер.

Тірек ұғымдар:

- ▶ PyGame кітапханасы;
- ▶ `pygame.display.set_mode(resolution=(0,0), flags=0, depth=0)` функциясы;
- ▶ `pygame.font.SysFont(name, size, bold=False, italic=False)` функциясы;
- ▶ `pygame.display.update()` функциясы.

Python тілі бірнеше кітапханаларды қамтиды, олар бағдарламалау тілін әртүрлі функциялармен толықтыруға мүмкіндік

береді. Кітапханалардың бірін қолданып, бағдарламаның кодында жалықтыратын және қайталанатын әрекеттерді орындамауға болады.

PyGame кітапханасы — Python тілінде интерфейсімен ойындарды жасауға арналған арнайы кітапхана. PyGame артықшылығы — бұл кітапхананың көмегімен жасалынған қосымшалар андрондта, сонымен қатар компьютерлерде де жұмыс жасайды. PyGame кітапханасының көмегімен көптеген ойындар және қосымшалар дайындалады (тізімді www.pygame.org ресми сайтынан қарауға болады).

PyGame кітапханасын орнату үшін PyGame жүктеу парақшасына кіріп (<http://www.pygame.org/download.shtml>), жүктеуге болатын кітапхананы таңдау қажет.

Ескерту. Оқулық басып шығару кезінде Python 3.x қарағанда 2.x көпшілікке белгілі (2.x тармағының соңғы нұсқасы 7). Барлық мысалдар 2.6 нұсқасы үшін келтірілген.

Содан кейін ойын үшін терезе жасаңдар. Ол үшін PyGame жүктейміз, бірақ алдымен бұл кітапхананы кодқа импорттаңдар. Келесі кодты енгізіндер:

```
import pygame, sys
from pygame.locals import *
```

Кітапхананы импорттағаннан кейін оны жүктеуге болады. Ол үшін келесі код қолданылады:

```
pygame.init()
```

Енді кадр жасауға болатын терезені құрастырыңдар. Ол үшін келесі функция қолданылады:

```
pygame.display.set_mode(resolution=(0,0), flags=0,
depth=0), мұндағы
```

- *resolution* — *x* және *y* осі бойынша терезенің өлшемі сақталатын сандар тізімі;
- *flag* — терезе жалауы (толық экран, өзгертілетін өлшемдер және т.б.);
- *depth* — түс тереңдігі.

```
screen = pygame.display.set_mode((400,100), 0, 32)
```

Осы мысалда өлшемі 400x100, түс тереңдігі 32-ге тең терезе жүктеледі. Содан кейін терезенің атын жазыңдар. Ол үшін келесі функция қолданылады:

```
pygame.display.set_caption(title, icontitle=None),
```

мұндағы

- *title* — терезе атауы бар жол;
- *icontitle* — атаудағы белгіше.

```
pygame.display.set_caption(«My First PyGame Windows»)
```

Терезеде мәтін енгізу үшін мәтін координаталарын және қаріп өлшемін білу қажет. Мәтін үшін *helloText* айнымалысын, ал координаталар үшін *x* және *y* айнымалыларын, қаріп өлшемі үшін *fontSize* айнымалысын таңдаңдар. Онда:

```
helloText = «Hello, World and GCUP»
(x, y, fontSize) = (10, 40, 14)
```

Қаріп келесі функциямен беріледі:

```
pygame.font.SysFont(name, size, bold=False,
italic=False),
```

мұндағы

- *name* — қаріп аты;
- *size* — өлшемі;
- *bold/italic* — қасиеті.

```
myFont = pygame.font.SysFont("None", fontSize)
```

Бұл мысалда “None” атаулы қаріп және *fontSize* айнымалысында сақталатын өлшеммен қолданылады.

Қаріптің түсі үшін *fontColor* функциясы қолданылады. Сандар — ол RGB жүйесінде түстің векторы (0 — минималды мән, 255 — максималды мән). Мысалы:

```
fontColor = (255, 255, 0)
```

bgColor функциясы терезе фонының түсін анықтайды. Мысалы:

```
bgColor = (255, 255, 255)
```

Мәтінді суретке түрлендіру үшін *fontImage:* функциясы қолданылады:

```
fontImage = myFont.render(helloText, 0, (fontColor)),
```

мұндағы

- *myFont* — мәтінді суретке түрлендіреді;
- *text* — мәтін;
- *antialias* — қаріпті тегістеу;
- *color* — мәтін түсі;
- *background* — фон түсі (міндетті емес параметр).

Енді кадр жасандар. Ол үшін кадрды артқы фон түсімен бояндар, ол `screen` нысанының `fillcolor`, `rect=None`, `special_flags=0`) функциясына арналған. Мысалы:

```
screen.fill(bgColor)
```

Кадрды көрсету үшін `pygame.display.update()` функциясы қолданылады. Бұл функция үшін параметрлер қолданылмайды. Мысалы:

```
pygame.display.update()
```

PyGame кітапханасы жасаған бірінші программаларды келтірейік:

```
import pygame, sys
from pygame.locals import *
pygame.init()
screen = pygame.display.set_mode((400,100),0,32)
pygame.display.set_caption(«My First PyGame Windows»);
helloText = «Hello, World and GCUP»
(x,y,fontSize) = (10,40,40)
myFont = pygame.font.SysFont(«None», fontSize)
fontColor = (255,255,0)
bgColor = (255,255,255)
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()
```

Бақылау сұрақтары



1. PyGame кітапханасының мүмкіндіктерін атаңдар.
2. PyGame кітапханасы қандай функцияларды ұсынады және оларды қолданғанда не үйренесіңдер?

Практикалық жұмыс

А деңгейі

Параграфта келтірілген программаны теріндер және орындандар.

В деңгейі

Есептің кодында (А деңгейінен) қаріптің мәтінін өзгертіндер. Бұл программаны компиляцияландар және орындандар.

С деңгейі

Биіктігі 600 пиксель және ені 300 пиксель терезе құрындар және кез келген мәтін енгізіндер, мысалы, аты-жөнінді енгізуге болады.

Рефлексия:

Менің білгенім Енді мен ... қолдана аламын.

§26–27

Артқы фон және ойынның кейіпкерлері. Кейіпкерлерді анимациялау

Бүгінгі сабақта:

- ▶ кейіпкерлерді анимациялауды;
- ▶ ойын кейіпкерлерінің қозғалысын программалауды;
- ▶ кейіпкерлерді пернетақтадан басқаруды;
- ▶ өз ойындарыңды жасауды үйренесіңдер.

Тірек ұғымдар:

- ▶ кейіпкерлерді анимациялау;
- ▶ қозғалыстарды бағдарламалау;
- ▶ кейіпкерлерді басқару.

Python-да кейіпкерлерді анимациялау және артқы фонды анықтау үшін келесі мысалды қарастырайық:

```
1 import sys, pygame
2 pygame.init()
3
4 size = width, height = 320, 240
```

```

5   speed = [2, 2]
6   black = 0, 0, 0
7
8   screen = pygame.display.set_mode(size)
9
10  ball = pygame.image.load(«ball.bmp»)
11  ballrect = ball.get_rect()
12
13  while 1:
14      for event in pygame.event.get():
15          if event.type == pygame.QUIT: sys.exit()
16
17      ballrect = ballrect.move(speed)
18      if ballrect.left < 0 or ballrect.right > width:
19          speed[0] = -speed[0]
20      if ballrect.top < 0 or ballrect.bottom > height:
21          speed[1] = -speed[1]
22
23      screen.fill(black)
24      screen.blit(ball, ballrect)
25      pygame.display.flip()

```

1-жол — PyGame-нің барлық модульдері бар кітапхананы жүктеңдер.

2-жол — бұл модульдердің әрбіреуін жүктеңдер.

8-жол — графикалық терезе ашындар.

10-жол — доптың суретін жүктеңдер. Суретті жүктеп, доптың суретін енгізгеннен кейін *ballrect* деп аталатын айнымалыны жасаңдар. PyGame тіктөртбұрышты аймақты ұсынатын ыңғайлы утилитарлы *Rect* деп аталатын нысанды қамтиды.

13-жол — бағдарлама дайын деп есептеледі. Циклдің ішінде доптың қозғалысын, оның секіруін тексеріңдер.

17—21-жолдар — *ballrect* айнымалысын ағымдағы жылдамдықпен жылжытыңдар. Егер доп экран шегарасынан тыс қозғалса, онда жылдамдықты сол бағытта айналдырыңдар.

23-жол — экранды RGB түспен толтырыңдар.

24-жол — экранда доптың суретін салыңдар. “Surface.blit()” әдісі арқылы суреттер өңделеді.

Енді көрінетін дисплейді жанарту қажет. PyGame дисплейді буфер арқылы басқарады. Содан кейін “pygame.display.flip()” әдісін шақырыңдар. Салынған сурет экранда шығады.



Бақылау сұрақтары

1. Терезе фонын қалай өзгертеді?
2. Дайын нысандарды ойын үшін қалай жүктейді?

Практикалық жұмыс

А деңгейі

Параграфта ұсынылған программаны теріндер. Программаны компиляцияландар және орындандар.

В деңгейі

Есептің кодында (А деңгейі) жасалынатын ойын терезесінің түсін, фонын өзгертіндер.

С деңгейі

А деңгейінен фонды өзгертіндер (шөптің түсіне ұқсас түске) және доп шынайы көріну үшін кішігірім көлеңке қосындар.

Рефлексия:

Менің білгенім Енді мен ... қолдана аламын.

§28–29

Шарттарды бағдарламалау

Бүгінгі сабақта:

- ▶ кейіпкерлерді пернетақтадан басқаруды;
- ▶ дайын сценарий бойынша ойын жасауды;
- ▶ ойын нәтижесін есептеу алгоритмін жүзеге асыруды үйренесіндер.

Тірек ұғымдар:

- ▶ шартты оператор;
- ▶ қатынас операторлары;
- ▶ күрделі шарттар;
- ▶ ойын элементі мен нысандарын құру.

Кез келген бағдарламада негізгі шарт болады. Нақты әрекеттердің орындалу/орындалмауына байланысты кодтың сәйкесінше бөлігін орындап, бағдарлама өзінің тәртібін өзгертеді.

Шартты операторлар қандай да бір шарттарға байланысты әрекеттерді орындаудың бір нұсқасын таңдауға мүмкіндік береді.

Қарапайым, шартты оператордың құрылымы: егер <шарт> орындаймыз: қандай да бір әрекет орындалады.

Мысалы :

```
if a<b:
print (a+b)
else:
print (a-b)
```

“Егер $a < b$, содан кейін $a + b$ сомасын шығарыңдар, әйтпесе айырымда шығарыңдар $a - b$ ”.

Шартты конструкциясын еске түсіріп, бағдарлама кодын жазайық және талдайық. “CollidingBalls” — бұл қарапайым 2D ойын. Ойын жасау үшін шартты конструкцияларды және кітапханалардағы *tkinter* мен *random* қолданайық.

- Терезені құрып, оқиғаға байланысты тінтуірмен жұмыс жасауды үйренесіңдер.

- Түрлі түсті доп — негізгі ойын нысаны тінтуірді басқанда оның пайда болуын қарастыра аласыңдар.

- Тінтуір арқылы доптың қозғалып, экранның шетіне дейін секіріп баруын басқаруды ойластырамыз.

- Допты ұрып қозғалтатындай басқа доппен соқтығысуына әсер етіп, керек қозғалысқа келтіру қажет.

- Қауіпті шарды анықтау керек. Доп оларға тиіп кетсе жеңілесіңдер. Бағдарлама кодын қарастырайық.

1—13-терезе мен доптың негізгі жолдарында қасиеттері және ойынның қалай ойналатыны сипатталған.

```
1. HEIGHT = 480
2. BG_COLOR = 'white'
3. MAIN_BALL_COLOR = 'blue'
4. MAIN_BALL_RADIUS = 25
5. BAD_COLOR = 'red'
6. COLORS = ['aqua', 'fuchsia', BAD_COLOR, 'pink',
'yellow', BAD_COLOR, 'gold', 'chartreuse', BAD_COLOR]
7. NUM_OF_BALLS = 9
8. MAX_RADIUS = 35
9. MIN_RADIUS = 15
10. DELAY = 8
11. INIT_DX = 1
12. INIT_DY = 1
13. ZERO = 0
```

14—46-жолдары доптың қозғалысы мен негізгі қасиеті, олардың терезе шегарасы және бір-бірімен өзара байланысын сипаттайды.

```

14. # ball class
15. class Ball():
16. def __init__(self, x, y, r, color, dx=0, dy=0):
17. self.x = x
18. self.y = y
19. self.r = r
20. self.color = color
21. self.dx = dx
22. self.dy = dy
23. def draw(self):
24. canvas.create_oval(self.x - self.r, self.y -
    self.r, self.x + self.r, self.y + self.r, fill=self.
    color,
    outline=self.color if self.color != BAD_COLOR else
    'black')
25. def hide(self):
26. canvas.create_oval(self.x - self.r, self.y -
    self.r, self.x + self.r, self.y + self.r, fill=BG_
    COLOR, outline=BG_COLOR)
27. def is_collision(self, ball):
28. a = abs(self.x + self.dx - ball.x)
29. b = abs(self.y + self.dy - ball.y)
30. return (a * a + b * b) ** 0.5 <= self.r + ball.r
31. def move(self):
32. # collision with the walls (Қабырғаға соқтығысу)
33. if (self.x + self.r + self.dx >= WIDTH) or (self.x -
    self.r + self.dx <= ZERO):
34. self.dx = -self.dx
35. if (self.y + self.r + self.dy >= HEIGHT) or (self.y
    - self.r + self.dy <= ZERO):
36. self.dy = -self.dy
37. # therballscollision (басқа доппен соқтығысу)
38. for ball in balls:
39. if self.is_collision(ball):
40. if ball.color != BAD_COLOR: # not a bad ball
        ball.hide()
        balls.remove(ball)
        self.dx = -self.dx
        self.dy = -self.dy

```

```

41. else: # badball (жеңіліс алып келетін қауіпті доппен
    соқтығысу кезіндегі әрекеті)
    a. self.dx = self.dy = 0
42. self.hide()
43. self.x += self.dx
44. self.y += self.dy
45. if self.dx * self.dy != 0:
46. self.draw()

```

47—67-жолдарда ойынның компьютерлік тінтуірмен өзара байланысын сипаттайды (яғни, тінтуірді басқа кезде қандай әрекеттер орындалады).

```

47. # process the mouse events
48. def mouse_click(event):
49. global main_ball
50. if event.num == 1: # left mouse button
51. if 'main_ball' not in globals(): # старт
52. main_ball = Ball(event.x, event.y, MAIN_BALL_RADIUS,
    MAIN_BALL_COLOR, INIT_DX, INIT_DY)
53. if main_ball.x > WIDTH / 2:
54. main_ball.dx = -main_ball.dx
55. if main_ball.y > HEIGHT / 2:
56. main_ball.dy = -main_ball.dy
57. main_ball.draw()
58. else: # turn left
59. if main_ball.dy * main_ball.dx > 0:
60. main_ball.dy = -main_ball.dy
61. else:
62. main_ball.dx = -main_ball.dx
63. elif event.num == 3: # right mouse button: turn right
64. if main_ball.dy * main_ball.dx > 0:
65. main_ball.dx = -main_ball.dx
66. else:
67. main_ball.dy = -main_ball.dy

```

68—81-жолдарда ойын терезесіндегі кездейсоқ қаспелтері бар шарларды кездейсоқ шашырауға арналған бағдарлама генераторы сипатталған.

```

68. # create a list of objects-balls
69. def create_list_of_balls(number):
70. lst = []
71. while len(lst) < number:

```

```

72. next_ball = Ball(random.choice(range(MAX_RADIUS,
    WIDTH - MAX_RADIUS)),
    random.choice(range(MAX_RADIUS, HEIGHT - MAX_
    RADIUS)),
    random.choice(range(MIN_RADIUS, MAX_RADIUS)),
    random.choice(COLORS))
73. is_collision = False
74. for ball in lst:
75. if next_ball.is_collision(ball):
76. is_collision = True
77. break
78. if not is_collision:
79. lst.append(next_ball)
80. next_ball.draw()
81. return lst

```

82—88-жолдарында бағдарлама генераторы жасаған қауіпті доппен соқтығысқан кезінде ойын аяқталады. Ойыншының жеңілетіні сипатталады.

```

82. # count the number of bad balls
83. def count_bad_balls(list_of_balls):
84. result = 0
85. for ball in list_of_balls:
86. if ball.color == BAD_COLOR:
87. result += 1
88. return result

```

89—98-жолдарда басты бағдарламалық цикл сипатталған.

```

89. # games main loop
90. def main():
91. if 'main_ball' in globals():
92. main_ball.move()
93. if len(balls) - num_of_bad_balls == 0:
94. canvas.create_text(WIDTH / 2, HEIGHT / 2, text=»YOU
    WON!», font=»Arial 20», fill=»lime»)
95. main_ball.dx = main_ball.dy = 0
96. elif main_ball.dx * main_ball.dy == 0:
97. canvas.create_text(WIDTH / 2, HEIGHT / 2, text=»YOU
    LOSE!», font=»Arial 20», fill=»red»)
98. root.after(DELAY, main)

```

99—112-жолдарда ойын терезесінің 1—13-жолдарында көрсетілген қасиеттері бар өрнектер қайта сипатталады.

```

99. # create a window, the canvas and start game
100. root = tkinter.Tk()
101. root.title(«Colliding Balls»)
102. canvas = tkinter.Canvas(root, width=WIDTH,
    height=HEIGHT, bg=BG_COLOR)
103. canvas.pack()
104. canvas.bind('<Button-1>', mouse_click)
105. canvas.bind('<Button-2>', mouse_click, '+')
106. canvas.bind('<Button-3>', mouse_click, '+')
107. balls = create_list_of_balls(NUM_OF_BALLS)
108. num_of_bad_balls = count_bad_balls(balls)
109. if 'main_ball' in globals(): # for restarts
110. del main_ball
111. main()
112. root.mainloop()

```



Бақылау сұрақтары

1. Python терезелерімен жұмыс істеу үшін қандай кітапхана қажет?
2. Python-да кездейсоқ элементті генерациялау үшін қандай кітапхана қажет?
3. Шартты конструкцияларды ойын элементінің тәртібіне қарай қалай басқаруға болады?

Практикалық жұмыс

А деңгейі

Параграфта келтірілген бағдарламаны іске қосындар. Бұл бағдарламаны компиляция жасап, іске қосындар.

В деңгейі

Тапсырма кодынан (А деңгейінен) BG_COLOR, HEIGHT терезесі жасаған сипаттарды өзгертіп, басқа терезе параметрлерімен тәжірибе жасаңдар.

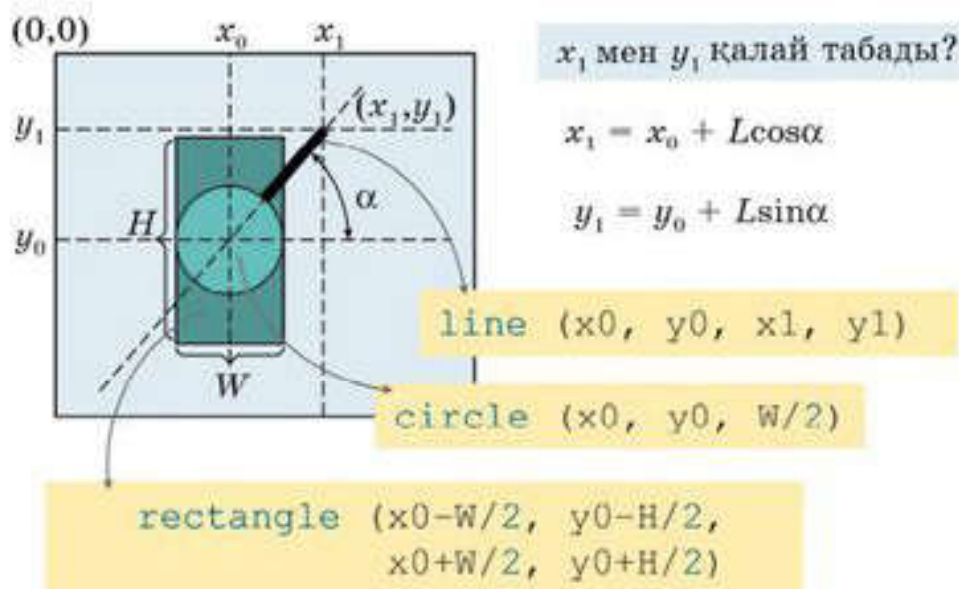
С деңгейі

Тапсырма кодында (А деңгейінен) тінтуірмен өзара әрекеттесу үшін құрылған доптардың сипаттарын өзгертіңдер.

Жобалық жұмыс.

“Зенбірегі айналатын танк” 2D ойынын құрастыру

Есептің қойылуы. Математикалық модельді сипаттау.



Программаны сипаттау:

```
1: from graph import *
2: import math
3: H = 60; W = 30; L = 40 # танк өлшемдері
4: x0 = 200; y0 = 400; angle = 90 # зенбірек
5: brushColor («#6b8e23»)
6: rectangle(x0-W/2, y0-H/2, x0+W/2, y0+H/2)
7: a = angle*math.pi/180 # радианға аудару
8: x1 = x0+L*math.cos(a)
9: y1 = y0-L*math.sin(a)
10: penSize(5)
11: line(x0, y0, x1, y1)
12: penSize(1)
13: brushColor("#556b2f")
14: circle(x0, y0, W/2)
15: run()
```

Кодты түсіндіру:

6 жол — танк корпусын сипаттау;

7—11-жол — оқпен;

12—15-жолдар — танк мұнарасы.

Зенбіректің айналуын анимациялау үшін келесі код үзіндісі қолданылады:

```

1: def keyPressed(event):
2:     if event.keycode == VK_LEFT:
3:         drawGun(angle+5) # солға 5 градусқа
4:     elif event.keycode == VK_RIGHT:
5:         drawGun(angle-5) # оңға 5 градусқа
6:     elif event.keycode == VK_ESCAPE:
7:         close()
8: ...
9: onKey(keyPressed)

```

Зеңбіректі салу және айналдыру. Алғашқыда суретін салайық, содан кейін тек координаталарын өзгертеміз:

```

1: def drawGun(angleNew):
2:     global angle, gun # глобальды айнымалылар
3:     angle = angleNew # жаңа бұрышты есте сақтау
4:     aRad = angle*math.pi/180 # радианға аудару
5:     x1 = x0 + L*math.cos(aRad)
6:     y1 = y0 - L*math.sin(aRad)
7:     if gun == None: # егер бірінші рет ...
8:         gun = line(x0, y0, x1, y1)
9:     else: # егер зеңбірек салынса
10:         changeCoord(gun, [(x0,y0), (x1,y1)] ) # жаңа
координаталардың массиві

```

Зеңбіректі салу және айналдыру . Ол жұмыс қалай істейді?

```

1: gun = None # зеңбірек салынбады
2: drawGun(angle) # бірінші рет саламыз
3: gun = line(x0, y0, x1, y1)
4: # енді gun ... адресін қамтиды
5: def keyPressed(event):
6:     drawGun(angle+5) # айналдырамыз
7: changeCoord(gun, [(x0,y0), (x1,y1)] )
8:     # координаталарын өзгертеміз

```

Сонымен, толық программаны ұсынайық:

```

1: from graph import *
2: import math
3: def keyPressed(event):
4:     ...
5: def drawGun(angleNew):
6:     ...
7: H = 60; W = 30; L = 40

```

```

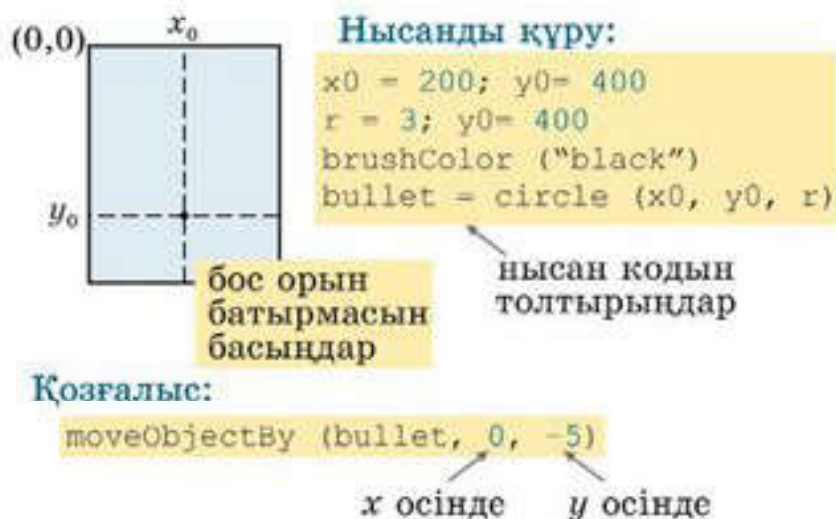
6: x0 = 200; y0 = 400; angle = 90
7: gun = None
8: brushColor («#6b8e23»)
9: rectangle(x0-W/2, y0-H/2, x0+W/2, y0+H/2)
10: penSize(5)
11: drawGun(angle)
12: penSize(1)
13: brushColor («#556b2f»)
14: circle(x0, y0, W/2)
15: onKey(keyPressed)
16: run()
    
```

Кодты түсіндіру:

- 2-3-жолдар — процедуралар ;
- 4—6-жолдар — бастапқы мәндер;
- 8-жол — корпус;
- 9-10-жолдар — зеңбірек;
- 11—14-жолдар — мұнара.

Зеңбіректен ату алгоритмін қарастырайық.

Есептің қойылуы. Математикалық модельді сипаттау.



Анимацияны қалай ұйымдастырады?

```

1: def update():
2:   global isFlying, bullet
3:   if isFlying: # егер ұшса...
4:     y = coords(bullet)[1]
5:     if y < 0: # егер ұшып кетсе...
6:       isFlying = False
7:       moveTo(bullet, x0-r, y0-r)
    
```



```

8:         else: # әрі қарай ұшады...
9:             moveObjectBy(bullet, 0, -5)
...
10: onTimer(update, 30) # әрбір 30 м/с сайын шақырылады

```

Қозғалыс қалай орындалады?

```

1: def keyPressed(event):
2:     global isFlying
3:     if event.keycode == VK_SPACE:
4:         isFlying = True # ұшады!
5:     elif event.keycode == VK_ESCAPE:
6:         close() # жұмысты аяқтау
...
7: onKey(keyPressed)

```

Толық программа.

```

1: from graph import *
2: def update():
...
3: def keyPressed(event):
...
4: x0 = 200; y0 = 400; r = 3
5: brushColor("black")
6: bullet = circle(x0, y0, r)
7: isFlying = False
8: onKey(keyPressed)
9: onTimer(update, 30)
10: run()

```

Тәрелкелерді ату.

Есептің қойылуы. Математикалық модельді сипаттау :

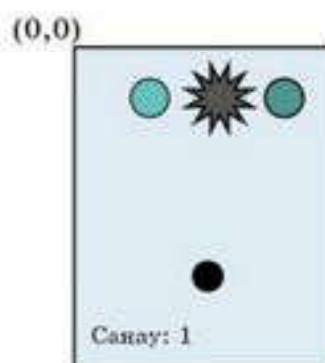
1. Тәрелкелер нысанын құрастыру *create Plates.*
2. Тәрелкелерді қозғалту *movePlates.*
3. Кез келген тәрелкеге тиегінін тексеру *checkCollision.*
4. Нақты тәрелкеге тиегінін тексеру *hit.*

Тәрелкелер массивін құрастырайық:

```

1: def createPlates( N ):
2:     global plates # глобальды массив

```



```

3:  yPlates = 100 # барлығында y-координата
4:  plates = [] # массив әзірше бос
5:  for i in range(N):
6:      brushColor( randColor() )
7:      p = circle(randint(0,500), # ортасы x
8:                yPlates, # ортасы y
9:                randint(10,20)) # радиус
10:     plates.append(p) # массивке қосу
...
11: createPlates( 5 ) # процедураны шақыру

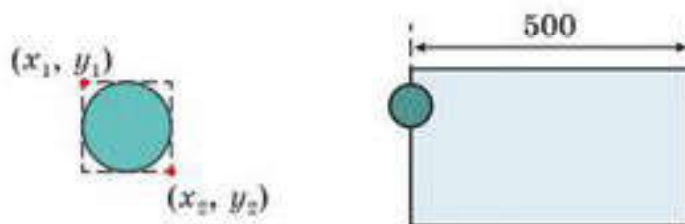
```

Тәрелкелердің қозғалысы.

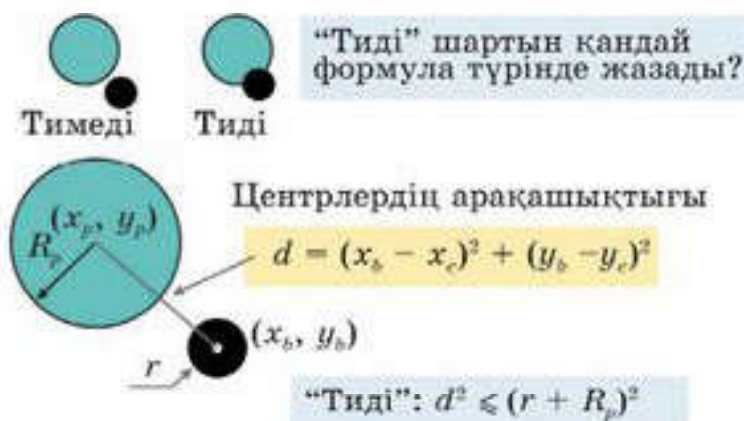
```

1: def movePlates():
2:     global plates # глобальды массив
3:     for p in plates: # әрбір тәрелке үшін
4:         moveObjectBy(p, -2, 0) # солға 2 қозғалу
5:         x1, y1, x2, y2 = coords(p)
6:         if x1 < 0: # егер шегінен шықса...
7:             # оңға... секіру
8:             moveObjectBy(p, randint(500,600), 0)

```

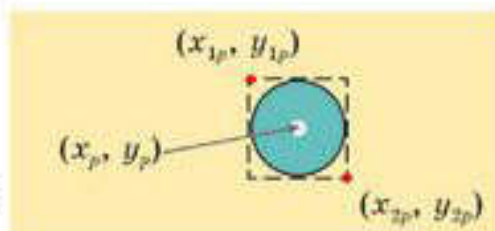
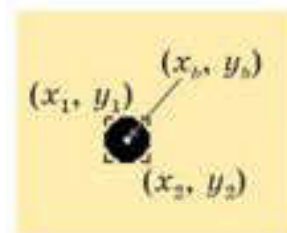


Оқ тәрелкеге тиді ме?



```

1: def hit(p):
2:     global bullet
3:     # оқтың координаталары
4:     x1,y1,x2,y2 = coords(bullet)
5:     xb = x1 + r # оқтың ортасы
6:     yb = y1 + r
7:     # тәрелке координаталары
8:     x1p,y1p,x2p,y2p = coords(p)
9:     xp = (x1p + x2p) / 2
10:    yp = (y1p + y2p) / 2
11:    Rp = (x2p - x1p) / 2
12:    d2 = (xb-xp)**2 + (yb-yp)**2
13:    return d2 <= (Rp+r)**2
    
```



Бұл функцияларды қалай шақырады?

```

1: def update():
2:     global isFlying, bullet
3:     movePlates()
4:     if isFlying: # егер ұшса...
5:         y = coords(bullet)[1]
6:         if y < 0: # егер ұшып кетсе...
7:             isFlying = False
8:             moveTo(bullet, x0-r, y0-r)
9:         else: # әрі қарай ұшады
10:            moveObjectBy(bullet, 0, -5)
11:            checkCollision()
...
12: onTimer(update, 30) # әрбір 30 с сайын шақырылады
    
```

Ойынның қорытындысын қалай шығаруға болады?



score қалай және қашан өзгереді?

Басында: `score = 0`

Тиген кезде: `score += 1`

Тамга (label элемент типі)

Тамганы құру:

```
lbl = label("Санаяу: 0", (10, 200), bg = "white")
```

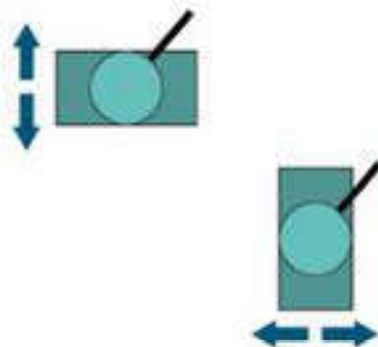
Тамғаның мәтінін өзгерту:

```
lbl["text"] = "Санаяу: " + str(score)
```

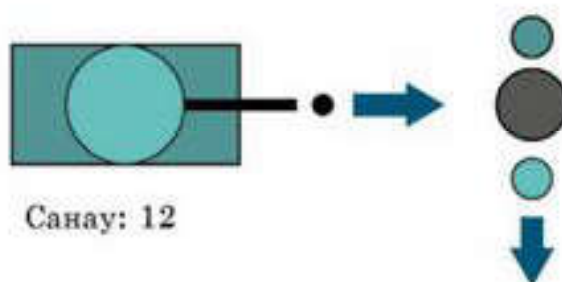
Сандардың жолы

Қосымша тапсырмалар

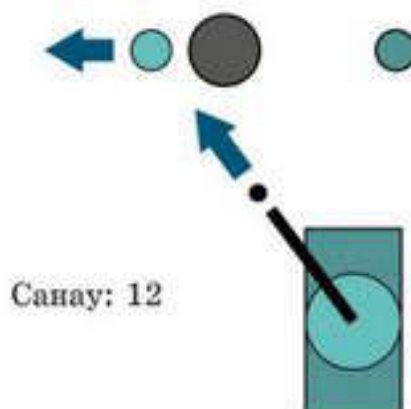
1. Екі танк құрастыру қажет, біреуінің зеңбірегі “солға” және “оңға”, екіншісінің зеңбірегі “жоғары” және “төмен” батырмаларымен басқарылады.



2. Ату сол жақтан оңға болатын (тәрелкелер жоғарыдан төменге ұшады) тапсырманы орындау. Тұмсығынан оқ ұшатын танк салу қажет.



3. Зеңбірегі айналатын танк құрастыру. Оқ оқпаннан бағыты бойынша атып шығады. Неғұрлым кішкентай тәрелкеге тиген жағдайда ойыншы көбірек ұпай жинайды.



ГЛОССАРИЙ

E-mail — электрондық пошта, интернет желісінің негізгі түрлерінің бірі.

IP-адрес — 0 мен 255 аралығында бір-бірінен нүкте арқылы бөлінген төрт саннан тұратын жазбаны білдіреді, мысалы 192.168.0.13.

URL — файлдың немесе беттің интернеттегі адресі.

Web-бет — “Бүкіләлемдік өрмектің” негізгі құрылымдық элементі, мәтіндік және (немесе) графикалық ақпаратты, сонымен қатар интернеттің басқа құжаттарына сілтемелерді қамтитын құжат.

Айналыру жолағы — құжат терезесінің немесе тізімнің оң жағында немесе төменгі жағында орналасқан. Құжат өзінің терезесіне сыймаған жағдайда, құжат бойымен жылжуға арналған. Бағыты көрсететін жүтіргіш және нұсқауышты қамтиды. Сызғыштан өзгеше болады.

Ақпарат — адамның хабарда болу деңгейін көтеретін бізді қоршаған әлем туралы мәліметтер.

Ақпарат тасымалдаушысы — ақпаратты сақтауға арналған материалдық нысан.

Ақпараттық қауіпсіздік — ақпараттық ресурстардың сақталу және ақпараттық ортада қоғам мен тұлғаның заңдық құқығының қорғалу жағдайы.

Ақпараттық модель — нысан немесе процесс туралы ақпараттардың жиынтығы.

Алгоритм — есептің шешімін қамтамасыз ететін нақты әрекеттерді орындау ережелерінің жиынтығы.

Алмасу буфері — Windows жұмысы кезінде бөлектелген жедел жад аймағы. Ол құжаттар мен қосымшалардың арасында мәтіндік және графикалық ақпараттарды алмастыруға арналған.

Алфавит — символдардың анықталған жиынтығы. Символдар – алфавиттің әріптері деп аталады.

Алфавитті-цифрлы ақпарат — дербес компьютерге тінтуірді (әріптер, сандар, қызметтік белгілер) бір рет басу арқылы енгізуге болатын кодтық кестенің символдары.

Бағдарлама — программистпен әзірленген және компьютер орындайтын нұсқаулық жиынтығы.

Бағдарламалау — компьютерлердің бағдарламалық қамтамасыз етуін әзірлеу бойынша кәсіби іс-әрекет.

Бағдарламалық жасақтама — компьютерде қолданылатын барлық бағдарламалардың жиынтығы.

Байт — 8 бит жиынтығы, компьютермен бір бүтін ретінде қабылданады.

Бит — табиғатта белгілі ақпараттың ең кіші бірлігі. Биттің мағынасы — 0 немесе 1, оны “ажыратылды-қосылды”, “жок-иә”, “ақиқат-жалған” сияқты түсіндіруге болады.

Бұлттық технологиялар (есептеулер) — мәліметтерді үлестіріп өңдеу технологиясы, онда компьютерлік ресурстар пайдаланушыға интернет-сервис ретінде ұсынылады.

Дизъюнкция — әрбір екі қарапайым сөзбен біріктірілген мәлімдеме болып табылатын логикалық операция. Екі мәлімдеменің мәні бір уақытта жалған болған кезде ғана мәні жалған болады және түпнұсқа мәлімдемелерінің кемінде біреуі шын болғанда мәні шын болады.

Жедел жад — арнайы электронды ұяшықтардың жиынтығы, оның әрбіреуі нөл және бірден тұратын нақты құрамын сақтайды.

Жедел жадның көлемі — компьютердің өнімділігіне әсер ететін маңызды сипат. Мегабайт және гигабайтпен өлшенеді.

Жүйе — өзара байланысқан нысандарының жиынтығы.

Интернет — компьютерлік желілерді біріктіретін ғаламдық желі.

Кенейту — файл атының бөлігі.

Кіріс ақпарат — адам немесе құрылғы қабылдайтын нысандар туралы ақпарат.

Компьютер — әртүрлі ақпаратты енгізуге, өңдеуге және бейнелеуге арналған құрылғы.

Компьютер сәулеті — компьютерді ұйымдастыруды және құрылымдық элементтердің қызмет ету принциптерін сипаттау. Ол дербес компьютердің негізгі құрылғыларын және олардың арасындағы байланыстар құрылымын қамтиды.

Компьютерлік этика — компьютерді қолданатын адамдардың тәртібін зерттейді.

Кроссплатформалық бағдарлама — бұл әртүрлі операциялық жүйелерге (Windows және Linux) арналған нұсқасы бар бағдарлама.

Мәзір — экранда ашылатын нысандар (операциялар, жалаулар) тізімі. Бұл нысандардың арасында таңдау жасалынады.

Мәліметтер — компьютер жадында екілік түрінде беріліп, өңделінетін ақпарат.

Мәліметтер қоры (МҚ) — бірдей қасиеттер жинағына не нысандар тобы туралы мәліметтерді реттелген түрде сақтауға мүмкіндік беретін ақпараттық модель.

Мәліметтер қоры жазбасы — мәліметтер қоры өрістерінде орналасқан мәндерді қамтитын кестенің жолы.

Мәліметтер қоры жолы — кестенің бағаны, ол қасиеті белгіленген мәндерді қамтиды.

Модель — нақты нысан туралы жеңілдетіліп көрсетілу.

Монитор — ақпаратты экранда көрсету құрылғысы.

Мультимедиа — ол бағдарламалық қамтамасыз етудің көмегімен компьютерде әдеттегі ақпаратты (мәтін немесе графика) дыбыс және қозғалатын суреттермен (бейнефильмдерді де жасауға болады) біріктіруге мүмкіндік беретін арнайы технология.

Нысан — бір бүтін ретінде қарастырылатын қоршаған әлемнің кейбір бөлігі.

Нысандар қасиеті — нысанды басқа нысандардан айыруға болатын белгілердің жиынтығы.

Провайдер — пайдаланушыларға интернетке шығуға мүмкіндік беретін өндіруші.

Процессор — бағдарлама командаларын автоматты түрде оқып, оларды декодтауға және орындауға арналған құрылғы.

Разрядтылық — берілген бір командада процессордың өңдей алатын екілік разрядтардың максималды саны.

Сетикет, нетикет — көбі ұстанатын желідегі қарым-қатынас жасау, тәртіп сақтау ережесі, интернет-қоғамдастық мәдениеті мен салт-дәстүрі.

Символдың ақпараттық салмағы — алфавиттің бір символынан тұратын ақпарат саны.

Сұрыптау — тізімде мәліметтердің орнын типі немесе мәніне сәйкес өзгерту.

Сыртқы жад — энергияға тәуелсіз жад. Оларға магнитті және оптикалық дискілер, флэш-жад жатады. Ақпаратты ұзақ уақытта сақтауға және бір компьютерден басқа компьютерге көшіру үшін қолданылады.

Таңбалық жол — нөл таңбасымен ('0') аяқталатын таңбалар тізбегі.

Терістеу — бұл әрбір қарапайым өрнекке бастапқы пікір теріс болатындай құрама пікір сәйкестендіретін логикалық операция.

Утилита — жүйелік бағдарламалардың функцияларын жетілдіру немесе кеңейту үшін қолданылатын қызметтік бағдарламалар, олардың негізгі міндеті компьютерлік жүйені баптау, жөндеу және тексеру жұмыстарын автоматтандыру болады (операциялық жүйені жетілдіруге және басқа мүмкіндіктеріне арналған арнайы қызметтік бағдарламалар).

Хаттама — белгілі ережелер, компьютерлер арасында ережелерге сәйкес ақпараттың алмасуы жүзеге асырылады.

Шығыс ақпарат — кіріс ақпаратты адам немесе құрылғының түрлендіруі нәтижесінде алынған ақпарат.

МАЗМУНЫ

1-тарау . АҚПАРАТПЕН ЖҰМЫС

§ 1. Ақпараттың сипаттары	4
§ 2-3. Құжаттармен бірігіп жұмыс істеу	11
§ 4. Желілік этикет	16

2-тарау . КОМПЬЮТЕРДІ ТАҢДАУ

§ 5. Компьютердің конфигурациясы	22
§ 6. Бағдарламалық жасақтаманы таңдау	25
§ 7. Компьютерді таңдау	30

3-тарау . МӘЛІМЕТТЕР ҚОРЫ

§ 8. Мәліметтер қоры	36
§ 9. Электрондық кестелерде мәліметтер қорын құру	40
§ 10. Ақпаратты іздеу әдістері	43
§ 11-12. Деректерді сұрыптау және сүзу	46
§ 13. Мәліметтер қорымен жұмыс істеу	51

4-тарау . PYTHON БАҒДАРЛАМАЛАУ ТІЛІНДЕ БАҒДАРЛАМАЛАУ АЛГОРИТМІ

§ 14. Бірөлшемді массив	54
§ 15-16. Белгіленген сипаттары бар элементті іздеу.	59
§ 17. Элементтердің орнын ауыстыру	64
§ 18-19. Екіөлшемді массив	67
§ 20-21. Сұрыптау	74
§ 22-23. Элементті жою және кірістіру	78

5-тарау . PYTHON БАҒДАРЛАМАЛАУ ТІЛІНДЕ 2D ОЙЫН ЖАСАУ

§ 24-25. PyGame кітапханасы	82
§ 26-27. Артқы фон және ойынның кейіпкерлері. Кейіпкерлерді анимациялау	86
§ 28-29. Шарттарды бағдарламалау	88
Глоссарий	101



Учебное издание

**Кольева Наталья Станиславовна
Шевчук Елена Владимировна
Ержанов Эльдар Бауржанович**

ИНФОРМАТИКА

**Учебник для 9 классов общеобразовательных школ
(на казахском языке)**

**Редакторы А. Сабдалиева
Техникалық редакторы И. Тарапунец
Суретші-дизайнер Л. Мельник
Корректоры Г. Тұрмағамбетова
Компьютерде беттеген С. Жұмагелдиева**

**Баспаға Қазақстан Республикасы Білім және ғылым министрлігінің
№ 0000001 мемлекеттік лицензиясы 2003 жылы 7 шілдеде берілген**

ИБ № 5918

**Басуға 20.06.19 қол қойылды. Піншімі 70x100^{1/16}. Офсеттік қағаз.
Қарп түрі “ММ Мектептік”, Офсеттік басылыс. Шартты баспа табағы 8,39.
Шартты бояулы беттаңбасы 34,21. Есептік баспа табағы 3,42.
Таралымы 15 000 дана. Тапсырыс №**

**“Мектеп” баспасы, 050009, Алматы қаласы, Абай даңғылы, 143
Факс: 8(727) 394-42-30, 394-37-58
Тел.: 8(727) 394-41-76, 394-42-34
E-mail: mektep@mail.ru
Web-site: www.mektep.kz**

