

Р.А. Қадырқұлов, Ә.Д. Рысқұлбекова,
Г.К. Нұрмұханбетова

ИНФОРМАТИКА

Жалпы білім беретін мектептің
8-сынып оқушыларына арналған оқулық

*Қазақстан Республикасының
Оқу-ағарту министрлігі
ұсынған*

АЛМАТЫҚІТАП БАСПАСЫ
2023

Шартты белгілер:

- | | | | |
|--|---------------------|--|------------------------------|
| | – сабактың мақсаты | | – жеке жұмыс |
| | – ойлан | | – ойлануға берілген сұрақтар |
| | – жаңа білім | | – тапсырма |
| | – практикалық жұмыс | | – үй тапсырмасы |
| | – талдау | | – үштілді сөздік |
| | – жинақтау | | – женіл тапсырма |
| | – бағалау | | – орташа тапсырма |
| | – топтық жұмыс | | – күрделі тапсырма |
| | – жұптық жұмыс | | – CD-дегі тапсырмалар |

«Информатика» оқу-әдістемелік кешенінің құрамына оқулық, мұғалімге арналған әдістемелік нұсқау кіреді. «Информатика» электрондық оқулығымен баспаңың www.topiq.kz платформасында таныса аласыз. Оқулықта интернеттен тәуелсіз жұмыс істеуге арналған CD бар.

Қадырқұлов Р.А. т.б.

Қ 15 **Информатика:** Жалпы білім беретін мектептің 8-сынып оқушыларына арналған оқулық. / Р.А. Қадырқұлов, Ә.Д. Рысқұлбекова, Г.К. Нұрмұханбетова – Алматы: АЛМАТЫКІТАП БАСПАСЫ, 2023. – 152 бет; суретті.

ISBN 978-602-02-1238-8

ӘОЖ 373.167.1
КБЖ 32.973 я 72

ISBN 978-602-02-1238-8

© Қадырқұлов Р.А., Рысқұлбекова Ә.Д.,
Нұрмұханбетова Г.К., мәтіні, 2021

© «АЛМАТЫКІТАП БАСПАСЫ» ЖШС, 2021

Құрметті оқушы!

«Информатика» пәнін оқу барысында сен компьютердің құрылғылары, компьютерлік желілер, ақпаратты өлшеудің алфавиттік әдістері туралы біліп, программалау тілінде алгоритмдер құру мен программа-лау негіздерін үйренесің. Жаңа білім мен тәжірибелі меңгеруде түрлі сұрақтарға жауап табуға, бірлесе талқылау арқылы тұжырымдар жасауға, практикалық-шығармашылық оқу тапсырмаларын орындау кезінде жеке ақпараттық-коммуникативтік сауаттылығынды жетілдіруге мүмкіндік аласың.

Оқулықта берілген модельдеу, жобалау мен зерттеу тапсырмалары қызықты әрі пайдалы. Жаңа тақырыптар қарапайымнан күрделіге бірте-бірте ауысып отырады. Тапсырмалар өз бетінше орындау дағдыларын игеруге арналған. Оқулықта берілген материалға жан-жақты талдау жасап, қорытындылап, өмірде қолдана білу үшін үлгі жобалар жасап үйренесің.

Оқулықтың басым бөлігіндегі тақырыптар программалау мен алгоритмдеуге бағытталған. Қазіргі қоғамда алгоритмдеудің рөлі оны пайдаланудың техникалық аспектілерімен ғана айқындалмайды. Алгоритмдік қатынас – адамның күнделікті өмірі мен жұмысының ажырамас бөлігі. Көп жағдайда адам қызметінің нәтижесі оның әрекеттерінің алгоритмдік мәнін қарашалықты дәл білетініне тәуелді: әр мезетте, қандай ретте не істеу керек; әрекеттер қорытындысы қандай болуы керек? Бұл белгілі дәрежеде алгоритмдерді құрастыру мен пайдалана білуге қатысты.

Оқулықта алгоритм құрастыру мен жазу түсініктері беріліп, қазіргі кезде кең тараған Python программалау тілінің ерекшеліктері де қарастырылған.

«Информатика» пәнінен алған білімің мен жаңа тәжірибел ғолашақта үлкен көмегін тигізері анық.

«Информатика» пәнінің қыр-сырын жан-жақты меңгеру үшін ынта мен ықылас, зейін мен табандылық қажет екенін есіңнен шығарма!

Оқуда сәттілік тілейміз!

1.1 Ақпаратты өлшеу. Алфавиттік әдіс



Ақпарат көлемін анықтауда алфавиттік амалды қалай қолдануға болады?



Ойлан

- «Ақпаратты өлшеудің алфавиттік әдісі» дегеніміз не?
- Оны қандай жағдайларда қолданады?



Жаңа білім

Алфавиттік әдіс

Ақпаратты өлшеудің алфавиттік жолы ықтималдық түрғыдан өлшеудің бір әдісіне жатады. Мұнда қандай да бір алфавиттің (символдық хабардың) символдарынан құрастырылған мәтіндегі ақпарат санын өлшеу қарастырылады. Ақпараттың мұндай өлшемінің мәтін мағынасына қатысы жоқ. Ақпаратты өлшеудің алфавиттік жолы – компьютерлер мен ақпараттық техникаларда айналып жүрген ақпаратты өлшеуге қолдануға болатын жалғыз әдіс.

А	Ә	Б	В	Г	Ғ
Д	Е	Ё	Ж	З	И
Й	К	Қ	Л	М	Н
Ң	Ң	Ө	П	Р	С
Т	Ұ	Ұ	Ү	Ү	Ф
Һ	Ц	Ч	Ш	Щ	Ъ
Ы	І	І	Ҙ	ҙ	Я

Бұл тақырыптың тірегі – «алфавит» ұғымы.

Алфавит – ақпаратты ұсыну үшін қолданылатын, шектелген символдар жиыны. Алфавиттегі символдардың толық саны **алфавиттің қуаттылығы** деп аталады.



Егер алфавиттегі барлық символ *бірдей жиілікпен кезігеді* (тең ықтималды) деп жорамалдасақ, онда әр (X) символдың беретін ақпарат көлемі Хартли теңдеуімен есептеледі.

$$N = 2^x$$

N – алфавиттің қуаттылығы

X – символдың ақпараттық салмағы

Бұдан мынадай тұжырым шығады: K символдан тұратын бүкіл мәтіндегі ақпарат көлемін (I) есептеу үшін, әр символдың (X) ақпараттық салмағын K-ға көбейтеді.



хабардың ақпараттылығы – информационность сообщения – information message
алфавиттік әдіс – алфавитный метод – alphabetic method

$$I = K \cdot X$$

I – ақпарат (саны) көлемі
 K – ақпараттағы символ саны (қуаттылығы)
 X шамасы – символдың ақпараттық салмағы

Ақпарат санының шамасын мәтіннің ақпараттық көлемі деп атауга болады. Ақпаратты тасымалдауга жарамды алфавиттің ең аз қуаты 2-ге тең. Мұндай алфавит **екілік алфавит** деп аталады.

Екілік алфавиттегі символдың ақпараттық салмағы $2^X = 2$ болғандықтан, ақпараттың көлемі 1 битке тең. Сонымен екілік алфавиттің бір символы 1 бит ақпарат береді. Бұл жағдаймен оқушылар компьютердің ішкі тілінің алфавиті – екілік кодтаумен танысқанда кездеседі.

Бит – ақпаратты өлшеудің негізгі бірлігі. **Байт** – қуаттылығы 256-та тең алфавиттің бір символының салмағы. Себебі $256 = 2^8$ болғандықтан, 1 байт = 8 бит болады.

Ақпараттың өлшемі оның мазмұнына емес, мәтіннің көлемі мен алфавит қуаттылығына байланысты болады.

Мысал. 20 символдан тұратын хабар 64 символдық алфавиттің әріптерімен жазылған. Ол қандай ақпарат мөлшерін көрсетеді?

Жауабы: 120 бит ақпарат мөлшерін көрсетеді.

Түсініктеме. Бұл жерде алфавит қуаты $N=64$ -ке тең, символдар саны $K=20$ болады. Онда Хартли формуласы бойынша ақпараттың салмағы (X) 6 битті, ақпарат саны (I) $20 \cdot 6 = 120$ битті құрайды.

Талдау

Үлгі тапсырмалардың шығару жолдарына талдау жа-сандар. Ойларыңды ортаға салындар.

1-үлгі есеп

32 символдық алфавиттің көмегімен жазылған хабар 40 символдан тұрады. Хабардың ақпараттық көлемі қандай?

Берілді:

$$N = 32$$

$$K = 40$$

Табу керек: $I = ?$

Шешуі:

$$N = 2^X \Rightarrow 32 = 2^X \Rightarrow X = 5;$$

$$I = K \cdot X \Rightarrow I = 40 \cdot 5 \Rightarrow I = 200 \text{ бит} = 40 \text{ байт}$$

Түсініктеме. 200 бит хабардың ақпараттық көлемін 5 бит ақпарат салмағына бөліп, байт арқылы өрнектедік.

Жауабы: 40 байт.

2-үлгі есеп

Кітаптың 150 беті компьютерде терілген. Әр бетте 40 жол, әр жолда 60 символ бар. Кітаптағы ақпарат қөлемін анықта.

Берілді:

$$K_1 = 150$$

$$K_2 = 40$$

$$K_3 = 60$$

$$N = 256$$

Табу керек: $I - ?$

Шешуі:

Ақпарат санының формуласы: $I = K \cdot X$

$$N = 2^x \Rightarrow 256 = 2^x \Rightarrow X = 8 \text{ бит};$$

$$K = K_1 \cdot K_2 \cdot K_3; \quad K = 150 \cdot 40 \cdot 60; \quad K = 360\,000; \\ I = 8 \text{ бит} \cdot 360\,000 = 2\,880\,000 \text{ бит} = 360\,000 \text{ байт} = 351,56 \text{ Кб.}$$

Түсініктеме. Компьютерде қолданылған алфавиттің күші 256-та тен.

Жауабы: 351,56 Кб.

3-үлгі есеп

Қазақ алфавиті 42 әріп, тыныс белгілері мен 10 цифрды қоса есептегендегі жалпы 64 символ деп қарастырайық. Ағылшын алфавиті 26 әріп, тыныс белгілерімен бірге 32 символ делік. Қазақ алфавитіндегі 70 символдан құралған хабар мен ағылшын алфавитіндегі 70 символдық хабардың ақпараттық қөлемдерін салыстыр.

Берілді:

$$N_1 = 64$$

$$N_2 = 32$$

$$K_1 = 70$$

$$K_2 = 70$$

Табу керек: $\frac{I_1}{I_2} - ?$

Шешуі:

Ақпарат санының формуласы: $I = K \cdot X$

$$N_1 = 2^{x_1} \Rightarrow 64 = 2^{x_1} \Rightarrow X_1 = 6 \text{ бит};$$

$$N_2 = 2^{x_2} \Rightarrow 32 = 2^{x_2} \Rightarrow X_2 = 5 \text{ бит};$$

$$I_1 = 70 \cdot 6 \text{ бит} = 420 \text{ бит}; \quad I_2 = 70 \cdot 5 \text{ бит} = 350 \text{ бит};$$

$$\frac{I_1}{I_2} = \frac{420}{350} = 1,2$$

Жауабы: 1,2 есе үлкен.



Жинақтау



Ақпаратты өлшеудің алфавиттік әдісіне есеп құрастыр. Есеп мазмұнының (есептің мөттіні) қызықты болуына назар аудар.



Бағалау



Ақпаратты өлшеудің алфавиттік әдісінің қандай маңызы бар?



Ойлануға берілген сұрақтар

1. «Алфавит» дегеніміз не?
2. Алфавиттің қуаттылығы нені өлшейді?
3. «Алфавиттегі символдың ақпараттық салмағы» дегеніміз не?
4. Мәтіннің ақпараттық көлемін алфавиттік түрғыдан қалай өлшеуге болады?
5. Ақпаратты алфавиттік өдіспен өлшеуде қандай формулалар мен шамалар қолданылады?
6. «Байт», «килобайт», «мегабайт», «гигабайт» дегеніміз не?



Тапсырма

1. Оңтүстік Америкада Амазонка өзені жағалауын мекендейтін үндістердің бүтінге дейін белгісіз болған бір тайпасы табылды. Тайпаңың алфавиті бар болғаны 8 әріптен тұрады. Сонда бұл алфавиттің бір әрпі қанша ақпарат көлемін береді?
2. Жер ғаламшарының даму кезеңдерінде бірнеше өркениет өмір сүріп, жоғалған. Сол өркениеттердің бірінде белгілі бір хабардың бір символының ақпараттық көлемі 6 бит болған делік. Жоғалған өркениет алфавитінде қанша символ бар екенін анықта.
3. 128 символдық алфавиті бар үндістер тайпасының 30 әріптен тұратын хатын шабарман көрші тайпаға жеткізді. Ол қандай ақпарат көлемін жеткізді?
4. 16 символдық алфавитте 512 символдық хабар қанша килобайтты құрайды?



Үй тапсырмасы

Зерттеу тақырыбы	Зерттеу бағыттары мен сұрақтары
Ақпаратты өлшеу әдістері	<ul style="list-style-type: none"> • Ақпаратты өлшеудің туындау себептері • Қонеден бүтінге дейінгі тарихы • Бізді қоршаған ақпарат • Ақпаратты өлшеу бірліктерін қолдануды зерттеу • Ақпаратты өлшеу әдістерін зерттеу

1.2

Ақпаратты өлшеу. Ықтималдық әдісі



Ақпарат көлемін анықтауда ықтималдық әдісін қалай қолдануға болады?



Ойлан

- Ақпарат көлемін анықтау не үшін қажет?
- Ақпарат көлемін анықтаудың қандай түрлерін білесін?



Жаңа білім

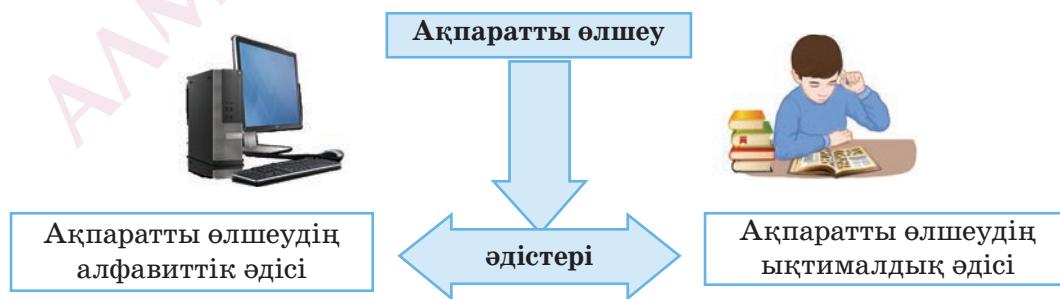
Ақпаратты өлшеу

Адамзаттың қолымен жасалған әр алтуан құрылғылар мен жүйелердің көмегімен белгілі бір дәрежеде ақпарат жинау мен өндөу жүргізіледі.

Хабар – ақпаратты жіберу процесінде қабылдаушыға түсетін ақпараттық ағын.

Хабар – біз еститін сөздер (радиодан берілетін хабар, мұғалімнің түсіндірмесі), қабылдайтын бейнелер (фильм, бағдаршам сигналы) және оқытын кітап мәтіні.

Хабардың ақпараттылығы дегеніміз – қабылдаушыға берілген хабардың маңыздылық дәрежесі. Ақпараттың көлемін анықтау үшін өлшем бірлігін енгізу қажет. Ақпараттың саны сол ақпаратты алушы адамға берілетін білімнің көлемімен айқындалады. Ақпаратты өлшеудің екі түрі бар. Олар: ақпаратты өлшеудің алфавиттік әдісі мен ақпаратты өлшеудің ықтималдық әдісі (*1-cурет*).



1-сурет. Ақпаратты өлшеу әдістері



ықтималдық әдісі – вероятностный метод – probabilistic method
алфавиттің қуаттылығы – мощность алфавита – power of the alphabet

Ақпарат көлемін анықтаудың ықтималдық әдісі

Адам білімінің белгісіздігін екі есе азайтатын хабар **1 бит** ақпарат әкеледі деп саналады.

Бит – өзара тең ықтималды екі тәжірибелің (оқиғаның) белгісіздігін өлшейтін бірлік.

1-мысал. Жеребеге лақтырылған тынның реверс («герб») беті түссе, 1 бит ақпарат әкеледі. Оқиғаның мүмкін болу саны 2-ге тең («герб» немесе «сан») және екеуі де **тең ықтималды**.

Түсініктеме. Сен үшін «бит» анықтамасын қабылдау қыын болуы мүмкін, себебі онда «білімнің белгісіздігі» деген түсініксіз ұғым бар. Алдымен оның мағынасын ашайық. Оны түсіну үшін мысал қарастырамыз.

Тынды лақтырудың алдында біз үшін **білімнің белгісіздігі** 2-ге тең болатын. Өйткені тынның екі беті бар, ал қай бетінің түсетіні белгісіз. Демек, нәтиже саны $2^x = 2$ дәрежелік теңдеуін шешкенде $X = 1$ бит деп шығады.

2-мысал. «Сүйек» ойында алты жақты кубик қолданылады. Кубиктің әрбір лақтыруынан ойыншы қанша бит ақпарат алады?

Түсініктеме. Кубиктің әр жағының түсінің **тең ықтималды**. Сондықтан кубикті бір лақтырғанда 6 жағының біреуі ғана үстіне қарап түседі. Сонда кубиктің қай жақ бетінің түсетінін анықтай алмай тұрған біз үшін **білімнің белгісіздігі** 6 есе азаяды. Ақпарат саны $2^x = 6$ теңдеуінен табылады.

Шектелген әрі мүмкін болатын N тең ықтималды оқиғаның біреуінің болуы тек **кездейсоқ оқиға** болып саналады.

Американдық инженер **Ральф Хартли** 1928 ж. ақпарат алу үрдісін **алдын ала берілген, шектелген N тең ықтималды хабардың жиынынан бір дерек хабарды таңдау** деп қарастырған. Белгілі бір хабарда N тең ықтималды оқиғалардың біреуінің болғаны туралы хабар алынды делік. Онда хабардағы ақпарат саны X бит және N саны Хартли формуласымен байланыстырылады. Ральф Хартли формуласының негізгі көрінісімен сен жоғарғы сыйнаптарда танысады.

$$N = 2^x$$

N – мүмкін болатын ықтималды оқиға саны
 X – ақпарат көлемі



1-кесте. Бит санына қарай оқиға санын анықтау

Бит саны	0	1	2	3	4	5	6	7	8	9	10
Оқиға саны	1	2	4	8	16	32	64	128	256	512	1024

Түсініктеме. Кестеге қарап (1-кесте), оқиға санымен сәйкес келетін бит санын анықтауға болады. Мысалы, сабакта 5 балдық жүйе бойынша «5», «4», «3», «2» деген төрт бағаның бірін алу мүмкін болатын тең ықтималды 4 оқиға бар («1» деген бағаны қарастырмаймыз). Осы жағдайда кестеден қарастырсақ, 4 оқиғаға 2 бит ақпарат сәйкес келеді екен. Әрине, бұл оқиға саны 2 санының бүтін дәрежесіне сәйкес келсе, қолайлы болады. Ал оқиға саны 7 немесе 23 болса, онда математиканың «Логарифм» бөліміндегі білім қажет болады. Логарифмдермен сен жоғарғы сыйынштарда танысадасың.



Талдау

Үлгі тапсырмалардың шығару жолдарына талдау жасаңдар. Ойларыңды ортаға салыңдар.

1-үлгі есеп

Оқушылар хауызға барды. Хауыздың 8 жолағы бар. Нұсқауды оларға 3 жолақтағанда жүзуге болатынын ескертті. Осы хабардан оқушы қанша ақпарат көлемін қабылдады?

Есептің берілуін тәмендегі үлгі бойынша жаз.

Берілді:

$$N = 8$$

Табу керек: $X - ?$

Шешімі:

$$N = 2^x$$

$$8 = 2^x \Rightarrow X = 3.$$

Түсініктеме. «2-нің қандай дәрежесі 8-ге тең?» деген сұраққа жауап береміз.

Жауабы: 3 бит.

2-үлгі есеп

«Қанат көпқабатты үйдің үшінші қабатында тұрады» деген хабар 4 бит ақпарат береді. Қанат неше қабатты үйде тұрады?

Берілді:

$$X = 4$$

Табу керек: $N - ?$

Шешімі:

$$N = 2^x \Rightarrow N = 2^4 \Rightarrow N = 16.$$

Түсініктеме. Есептің жауабына Қанаттың қай қабатта тұратыны өсер етпейді. Қанат 7-қабатта тұрса да, нәтиже дәл осындай болады.

Жауабы: 16 қабат.

3-үлгі есеп

Мектеп кітапханасында 16 стеллажға кітаптар қойылған. Әр стеллажда 8 сөре бар. Кітапханашы Қанатқа оның іздейген кітабы 7-стеллаждың 5-сөресінде тұрғанын айтты. Оқушы қанша ақпарат көлемін қабылдады?

Берілді:

$$N_1 = 16$$

$$N_2 = 8$$

Табу керек: $X - ?$

Шешімі:

$$N_1 = 2^{x_1}; N_2 = 2^{x_2};$$

16 = 2^{x_1} болса, $X_1 = 4$;

8 = 2^{x_2} болса, $X_2 = 3$.

Оқушының қабылдаған жалпы ақпараты:

$$X = X_1 + X_2$$

$$X = 4 + 3 = 7$$

Жауабы: 7 бит.

Түсініктеме. Есептің жауабында стеллаж
бен сөрелер саны есепке алынады.



Ойлануға берілген сұрақтар

- Хабар алушы адамға берілетін білім көлемі немен анықталады?
- Ақпарат көлемін анықтауда «ықтималдық амалы» дегеніміз не?
- Білімнің белгісіздігін екі есе азайтатын хабар неге тең?
- Ральф Хартли формуласы нені сипаттайты?
- «Кездейсоқ оқиға» дегеніміз не?



Тапсырма



- 1... N аралығындағы санды табуда 9 бит ақпарат қабылданған болса, N саны нешеге тең?
- «Ақан әліппесін жоғалтты» хабарында 4 бит ақпарат бар. Ақанның қанша оқулығы бар?
- Өлшемі 8x8 шахмат тақтасын кодтауға қажет ақпарат көлемі қандай?
- Көлемі 10 бит болған хабар қабылданғы. Алынған деректен қандай мөлшерде ақпарат құруға болады?
- «Қорапта түрлі түсті 16 шар бар. Қораптан ақ шар алынды» деген хабар қандай ақпарат көлемін көрсетеді?
- 64 тең ықтималды оқиғаның болуы жайлы хабардың көлемі қанша?
- «Оқушыны қызықтырган кітап 5-сөреде орналасқан» деген хабар 3 бит ақпарат болды. Стеллажда қанша сөре бар?
- «Қорапта 8 ақ және 32 көк шардан тұратын жалпы саны 40 шар бар. Қораптан ақ шарды және көк шарды алды» деген хабарда ақпарат көлемі қандай?
- 1... N аралығындағы санды табуда 6 бит ақпарат қабылданған болса, N саны нешеге тең?



Үй тапсырмасы

- Көлемі 8 бит болған хабар қабылданғы. Алынған деректен қанша мөлшерде ақпарат құруға болады?
- Мектеп кітапханасында кітап қойылған стеллаж саны 5 бит ақпарат, ал бір стеллаждағы сөре саны 4 бит ақпарат береді. Мектеп кітапханасында неше стеллаж бар? Әр стеллажда қанша сөре бар?

1.3

Компьютердің ішкі құрылымы мен қосымша құрылғылары



Дербес компьютердің ішкі құрылымы қалай жасақталған?



Ойлан

- «Бионика» дегеніміз не?
- Биониканың компьютерге қандай қатысы бар?
- Компьютерді құрастыруда қандай идеялар негізге алынған?

Бионика (грек. *βίον* – тіршілік ететін) – жанды табиғаттың үйімдастыру принциптерін, қасиеттерін, функциялары мен құрылымын, яғни жанды табиғат нысандары мен олардың өнеркәсіптік баламасын техникалық құрылғыларда және жүйелерде пайдалантын қолданбалы ғылым.



Жаңа білім

Компьютердің ішкі құрылымы

Дербес компьютер – өмбебап техникалық жүйе. Оның конфигурациясын (негізгі конфигурациясы: жүйелік блок, монитор, пернетақта) қажетінше өзгертуге болады. Жүйелік блок – негізгі құрылғы. Оның ішіндегі құрылғылар ішкі, ал сырттан қосылған құрылғылар сыртқы құрылғылар деп аталады.

Жүйелік блоктың ішкі құрылғылары:

- аналық тақша;
- процессор немесе микропроцессор;
- жедел жад;
- қатқыл дискідегі деректер жинақтауыш;
- алмалы дискілермен жұмыс істеуге арналған дискжетек;
- CD және DVD дискжетектері;
- бейнеграфикалық бейімдеуіш;



компьютердің ішкі құрылымы – внутреннее устройство компьютера –
internal-computer structure

аналық тақша – материнская плата – motherboard

слот кеңейтілімі – слот расширения – expansion slot

- дыбыстық тақша;
- желілік тақша;
- енгізу-шығару порттары;
- қоректендіру блогы.

Аналық тақша

 **Аналық тақша** (Материнская плата; ағыл. *motherboard*) – орталық процессор, жедел жад микросхемалары, дыбыстық және бейнекарталар сияқты компьютердің ішкі құрылғылары орналастырылған негізгі мөрлі тақша. Ол – кез келген компьютердің негізгі құрамдас бөлігі. Кейде бас немесе жүйелік тақша деп те аталады (*1-сурет*). Аналық тақша мыс өткізгіштер – жолдар желісімен қапталған. Олардың бойымен деректер тақшада құрастырылған микросхемалар мен компьютердің басқа құрылғылары қосылатын слоттарға жеткізіледі. Әрбір құрылғы белгілі бір слотқа қосылады. Аналық тақша өлшемі BIOS, процессор типі, чипсет типі, желілік шина жиілігі, кэш-жад өлшемі, жедел жад типі (слоттар саны мен жедел жадтың ең үлкен көлемі), контроллер, қатқыл дискінің типі т.б. параметрлермен сипатталады.



1-сурет. Аналық тақша

Процессор – көптеген жартылай өткізгішті элементтерден тұратын, компьютерде барлық есептеу мен ақпарат өндеу жұмыстарын жүргізетін электронды микросхема. Жартылай өткізгіш кристалл өте таза кремнийден жасалады. Оны жасауда вакуумдық бүрку, қоспаларды иондық түрде енгізу, фотолитография т.б. жоғары сапалы технологиялар қолданылады.



Бір кристалда жұз мыңнан астам өзара байланысқан электрондық молекулалар орналасып, күрделі ақпаратты түрлендіру жұмыстарын орындайды. Процессор разрядтылығы оның бір тактіде өз регистрлерінде қанша деректер *битін* қабылдап, өндейтінін көрсетеді. Процессордың ең қарапайым элементар ішкі операцияларының орындалуы тактілік жиілікпен өлшенеді. Жиіліктің өлшем бірлігі – Гц (Герц).

Қоректендіру блогы – кернеуі, жиілігі, қуаты белгілі электр энергиясымен компьютерді жасақтайтын құрылғы. Әдетте компьютерлер 5 немесе 12 Вольттық төменгі кернеуді пайдаланады. Қоректендіру блогтарының айырмашылықтары олардың қуаттылығында және кернеуді тұрақтандыруында.



Түрлі түсті сымдар (өткізгіштер) кернеу мөлшерлерін айқындайды: сарыға + 12В, қызылға + 5В, қара түске 0В кернеу беріледі. Қуаты 150 Ваттан 320 Ватқа дейін болады. Қоректендіру блогында желдеткіш бар. Ол қоректендіру блогын ғана емес, жүйелік блокта орналасқан құрылғыларды да салқындатады. Процессорды салқындатып тұратын арнайы желдеткіш (кулер) бар.

Шина – компьютер құрылғылары арасындағы электрлік сигналдарды басқаратын әрі олар арқылы деректер тасымалдауды жүзеге асыратын жалғастыру арналары. Шинаның негізгі үш түрі бар. Олар: деректер шинасы, адрестік шина және командалық шина. Деректер шинасы – деректердің екілік разрядтарын компьютер құрылғыларының арасында параллель жеткізуге арналған жалпы шинаның бөлігі. Жад адресін жеткізуге арналған жалпы шинаның бөлігін немесе енгізу-шығару портын **адрестік шина** деп атайды. Шинаның түрлері: ISA, MCA, EISA, SCSI, VLB, PCI, AGP, USB т.б.



Чипсет (ағыл. *chipset* – микросхема жиынтығы) – үқсас міндет атқаратын микросхемалар жиынтығы. Орталық процессордың шалғай құрылғылармен деректер алмастыру жұмысын жасақтайтын, аналық тақшада орналасқан микросхемалар жиынтығы чипсет деп аталаады.



BIOS микросхемасы – аналық тақшада орналасқан компьютердің негізгі құрылғыларының бірі. BIOS микросхемасы жүйе параметрлерін орнатуға арналған. Енгізу-шығарудың базалық жүйесі BIOS мынадай қызметтер атқарады:

- компьютерді іске қосқанда автоматты тестілеуді (POST) орындауды;
- аппараттық қуралдардың параметрлерін баптайды;
- операциялық жүйені дискіден орнатады;
- жүйелік, аппараттық үзілүлердің қызметтің қамтамасыз етеді.



Қазіргі кезде BIOS-тың көптеген түрі кездеседі. Бірақ олардың арасында дербес компьютерге арналған және көп тараған түрлері: AwardBIOS, AMIBIOS, PhoenixBIOS және UEFI.

2000 жылға дейін AwardBIOS үлкен сұранысқа ие болды. Бұл Pentium II, Pentium III процессорлары бар компьютерлерде қолданылған есік BIOS түрі. Егер ол істемей қалған жағдайда, компьютерді қайта қосу мүмкін болмаған. Қазіргі кезде AwardBIOS негізінен есік компьютерлерде кездеседі. Бұл BIOS-тың орнын қазіргі кезде PhoenixBIOS иеленді.

AMIBIOS – өте кең таралған BIOS. BIOS-тың бұл түріне компьютер құрылғысының жұмысына қажетті өзгерістер енгізу үшін көптеген өндөулер жасалды.

UEFI (Unified Extensible Firmware Interface – кеңейтілген дәнекер интерфейс) – BIOS-тың ең соңғы нұсқасы (*2-сүрет*). Ол компьютердің енгізу мен шығару жүйесіндегі емес, операциялық жүйеге ұксатып жасалған. UEFI-мен жұмыс істеудің негізгі артықшылықтары ретінде қолайлы интерфейсін, тінтуірмен басқаруды, ғаламтор көмегімен жаңалау мүмкіндігін, жылдам қосылуын атап өтуге болады.



2-сүрет. UEFI BIOS-ы

Сопроцессор – негізгі процессордың қызметтік мүмкіндігін толықтыратын мамандандырылған арнайы процессор. Сопроцессордың өз программасы болмайды, ол негізгі процессордың командалар жүйесі мен регистрлерін кеңейтіп, сонымен бірігіп жұмыс атқарады. Ол жүйені басқармайды, тек орталық процессордан команда күтеді. Арифметикалық операциялардың орындалу уақытын 80% немесе одан да қысқартуға мүмкіндік береді. Бейнесхемалық кескіндерді өндөу де жылдамдайды.

Слот кеңейтілімі – компьютердің аналық тақшасындағы кеңейтілімнің алып-қосу ойығы. Слот графикалық адаптерді, тізбек пен параллель интерфейстерді, дыбыс карталарын шинамен байланыстырады.



Компьютердің слот кеңейтілімдері:

- AGP – бейнекарта.
- AMR – модем, дыбыс картасы.
- CNR – модем, желі картасы, дыбыс картасы.
- EISA – SCSI, желі картасы, бейнекарта.
- ISA – желі картасы, дыбыс картасы, бейнекарта.
- PCI – желі картасы, дыбыс картасы, бейнекарта.
- SCSI, PCIe – бейнекарта, желі картасы, дыбыс карталары т.б.

Аналық тақшага қосымша жалғанатын тақшалар

Бейнекарта (Видеокарта) – кескіндерді сақтау үшін ғана пайдаланылатын меншікті жедел жады бар құрылғы. «Тұрғылықты жері» бойынша бұл жад көбінесе **бейнежад** (*Video RAM* немесе *VRAM*) деп аталады. Бейнежад көлемі неғұрлым үлкен болса, компьютер кескіндер мен бейнероликтерді соғурлым сапалы көрсетеді.



Дыбыстық тақша компьютерге жоғары сапалы дыбыс шығаруға өрі жазуға мүмкіндік береді. Дыбыстық тақша ұсынатын дыбыспен жұмыс істеудің кеңейтілген мүмкіндіктері компьютерлік ойындарда, басқа да заманауи программаларда қолданылады.



Желілік тақша компьютерді компьютерлер желісіне қосуға мүмкіндік береді. Желілік тақшалардың бірнеше түрі бар: **Ethernet**, **Token Ring** және сымсыз желілермен байланысуға арналған тақшалар. Ең танымалданысы – **Ethernet** пен сымсыз желілер (**Wi-Fi**).



Енгізу-шығару порттары – корпустың алдыңғы немесе артқы тақтасындағы ажыратқыштар. Оларға әдетте кабель арқылы әртүрлі құрылғылар қосылады. Порттарға қосуға болатын құрылғылардың саны мен түрі компьютер порттарының саны мен түріне тәуелді.





Дербес компьютердің қосымша құрылғылары



3-сурет. Компьютердің қосымша құрылғылары



Талдау



3-суретке талдау жасаңдар.

Талдау барысында:

- компьютердің қосымша құрылғыларының атауларын анықтаңдар;
- құрылғылардың қызметін, олардың қандай қосымша тақшаларға жалғанатынын сипаттаңдар;
- бұлардан басқа тағы қандай құрылғылар бар екендігін анықтаңдар.



Жинақтау



Компьютердің қосымша құрылғыларын пайдаланудың ортақ ережесін құрастырыңдар. Өздерің құрастырған ереженің әрбір пунктіне түсініктеме беріңдер.



Бағалау



Компьютердің құрылымындағы аналық тақшаның рөлін бағала.



Ойлануға берілген сұрақтар

1. Компьютердің ішкі құрылғыларына не жатады?
2. «Аналық тақша» деген не?
3. Аналық тақшага қандай құрылғылар жалғанады?
4. Дербес компьютердің қосымша құрылғыларының жалпы қызметі қандай?
5. «Процессор» деген не?
6. Бейнекарта жадының атқаратын қызметі қандай?
7. Компьютерде қандай қосымша тақшалар бар?
8. Жүйелік блоктың енгізу-шығару порттарына қандай құрылғылар жалғанады?



Үй тапсырмасы

№	Зерттеу тақырыптары	Зерттеу сұрақтары мен бағыттары
1	Қағазға басу құралдарының даму тарихы	<ul style="list-style-type: none"> • Адамзат өркениетінде қағазға жазу мен басудың ежелден келе жатқан қандай құралдарын білесің? • Заманауи қағазға басу құралдарының жұмыс негіздері • Принтердің қандай түрлері бар, олар қалай жұмыс істейді? • «3D принтер» деген не?
2	Мен және интернет	<ul style="list-style-type: none"> • Интернеттің пайда болуы мен даму тарихын зерттеу • Компьютерді интернетке қосуға қажет құралдар мен олардың түрлерін зерттеу • Интернеттің Қазақстандағы даму барысын зерттеу
3	USB құрылғысы және оның дамуы	<ul style="list-style-type: none"> • USB порттарының қалай пайда болғанын зерттеу • USB порттарының алдағы даму бағыты қандай?
	«Компьютер сөүлеті» дегеніміз не?	<ul style="list-style-type: none"> • Заманауи компьютерлер сөүлетінің даму үрдісін зерттеу

1.4

Процессор мен оның сипаттамалары



Компьютер процессорының қызметі мен оның негізгі сипаттамалары түралы не білесін?



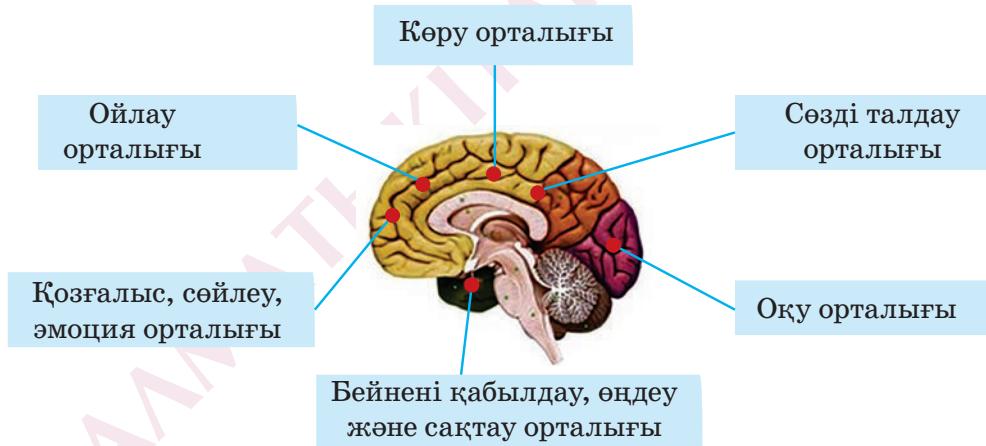
Ойлан

- Техникада адамның моделіне қандай мысалдар келтіре аласын?
- Адам мен компьютер жұмысы арасында қандай үқастықтар бар?
- Компьютер адамның қандай іс-әрекеттерін орындай алады?
- Компьютер адамды толық алмастыра ала ма?



Жаңа білім

Адамның миы ағзадағы барлық органның жұмысын реттеп отырады (1-сурет). Адамның миы компьютерге үқсайды.



1-сурет. Ақпаратты қабылдау, өндеу және сақтауға арналған адам миының аймақтары

Компьютер – берілген программа бойынша жұмыс істейтін қарапайым «миы» бар, ақпаратты өндеуге арналған электрондық құрылғы. Фалымдар көбінесе адамның миын компьютермен салыстырылады.



оптикалық дискжетектер – привод оптических дисков – odd optical disk drive
енгізу-шығару порттары – порты ввода-вывода – ports input-output

рады (1-кесте). Дегенмен де компьютердің бөлшектерін үнемі жетілдіріп, ауыстырып отырғанмен, оның жұмысын адамның миымен салыстыруға келмейді. Адам миының құрамы органикалық болғандықтан, оны ешбір бөлшекпен ауыстыру мүмкін емес және жасанды есептеу жүйелерімен салыстырғанда анағұрлым қарқынды. Қазіргі компьютерлердегі ең үлкен жедел жад көлемі 128 Гбайтты, ал ең үлкен қатқыл диск көлемі 14 Тб құрайды. Адам миының жады 2,5 Пбайтқа (\approx 2500 Тб) тең, яғни компьютер жадынан 180 есе үлкен.

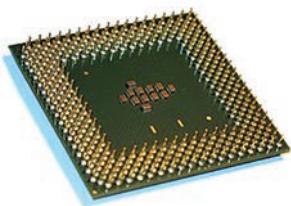
1-кесте. Адамның миы мен компьютерді салыстыру

Қызметі	Адам	Компьютер	Құрылғылар
Ақпаратты сақтау	Зерде	Жад құрылғылары	Тұрақты, жедел, кэш-жад және сыртқы жад
Ақпаратты өндөу	Ойлау жүйесі	Процессор	Процессор
Ақпаратты қабылдау	Сезім мүшелері	Енгізу құрылғылары	Пернетақта, микрофон, сканер, камера, тінтуір
Ақпаратты тасымалдау	Сөйлеу, қозғалыс жүйесі	Шығару құрылғылары	Дыбыс колонкалары, принтер, плоттер

Орталық процессор (процессор) – CPU (ағыл. *Central Processing Unit, processing* – өндөу) дербес компьютердің «миы».

Процессор – программаға сәйкес операцияларды орындауға арналған компьютердің ең қымбат микросхемасы. Онда есептеулер мен деректерді өндөу жұмыстары орындалады.

Компьютердің есептеу жылдамдығы процессорға тікелей байланысты. Өндірісте процессорды негізінен **Intel** және **AMD** фирмалары жасап шығарады. **Intel**-дің екіядролық процессоры **Core 2 Duo**, ал **AMD**-нің **Dual-Core** деп аталады. Процессордың жиілігі жоғарылаған сайын оның жылу бөлуі де артады. Бұл мәселе бір корпусқа екі процессор орнату арқылы шешіледі. Осындаид процесорларды **көпядролы** деп атайды. Қазіргі кезде процесорлар нарығында **Intel** фирмасының процесорлары басымдықта ие болып тұр. Процесорлар разрядтылығы, тактілік жиілігі мен кэш-жадының көлемімен ерекшеленеді.

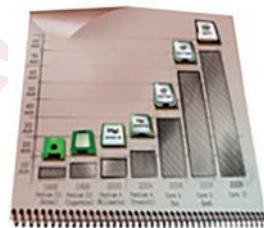


Процессордың негізгі параметрлері

Процессордың разрядтылығы – бір тактте өндей алатын биттер санымен анықтады. Бірлік ақпараттың ең кішісін **екілік разряд** дейді. ЭЕМ әлемінде бір разряд бит (*bit* – бөлік) деп аталады. Егер ЭЕМ бір ретте сегіз разрядты өндей алса, онда процессор **8 разрядты** деп аталады; егер 32 разрядты болса, онда процессор **32 разрядты** деп аталады. Процессор микросхемаларының разрядтылығы қаншалықты көп болса, ол ақпаратты соншалықты тез өндейді.

Процессордың тактілік жиілігі (*frequency*). Разрядтылықтан басқа, процессордың маңызды қызметтерінің бірі – тактілік жиілік. Тактілік жиілік – бұл процессор бір секундта орындаған операциялардың саны. Ол мегагерцпен есептеледі. Бір мегагерцте секундына миллион такт болады. 100 МГц секундына жүз миллион такт болып есептеледі. Бір тактіде процессор қандай да бір фрагментті есептейді. Тактілік жиілік қанша жоғары болса, процессор түскен деректерді сонша тез өндейді.

Процессордың кэш-жады. Процессор өндейтін деректерді жедел жадтан алады. Әдетте процессор өзінде еш нәрсені де сактамайды. Онда деректер өнделетін үяшықтар өте аз. Бұл жұмыс үяшықтары *register* деп аталады. Процессордың жұмысын жеделдешу үшін көптен бері кэштеу технологиясы қолданылып жүр. Кэш – шағын үяшықтық жад. Ол буфердің қызметін атқарады. Егер бір нәрсе жалпы жадта өнделсе, оның көшірмесі кэш-жадқа түседі. Егер бұл деректер тағы керек болып жатса, алысқа жүгінбей, буферден алынады. Процессорға деректер керек болса, ол бірінші регистрді тексереді. Ол жерде керек дерек болмаса, жақын жердегі кэш-жадты тексереді. Кэш-жад болмаса, онда жедел жадты тексереді. Егер сұралған деректер мұнда да жоқ болса, онда процессор қатқыл HDD дискіге сұраныс жасайды.



Процессордың құрылышы

Процессордың ядролар саны

Бүгінде процессорлар біздің өмірімізде маңызды рөл атқаралының техникалық құрылғыға айналды. Оларды адамдар әскери сала мен медицинада, білім беру мен бизнесте, тұрмыста түрлі мақсаттағы электрондық құрылғыларда пайдаланады. Процессорлар тек қана компьютерде емес, теледидар, телефон, тоңазытқыш сияқты құрылғыларда да қолданылады. Микропроцессорларды дамыту электрондық құрылғылардың функционалдық және есептеу мүмкіндіктерін арттырады.

Заманауи процессорлар екі немесе көпядролы болып жасалуда. Алғашқы екіядролы процессорларда екі тәуелсіз ядро болды. Ол әрқайсысы бірінші мен екінші дәрежелі кэш-жады бар бірдей құрылымды ядролардан тұрады. Ойынға арналған ноутбуктың өнімділігі жоғары, себебі ол 4 ядролы процессормен жабдықталады (*2-сүрет*).



2-сүрет.

4 ядролы процессор

Қазір ядролар ортақ екінші дәрежелі кэш-жадты құрайды, ол процессордың өнімділігін одан сайын арттырады. Көпядролы процессорды қолдану компьютердің өнімділігін жоғарылатады. Олардың бір уақыттың өзінде бірнеше команда ағынымен жұмыс істей мүмкіндігі бар. Басқаша айтқанда, **көпядролы процессор** дегеніміз – бірнеше процессордың жиынтығы. Олар аналық тақшаға біріктіріліп, өзара байланыстырылады. Көпядролы процессордың өнімділігі деректерді бірнеше ядро арасында бөліп алып өндеу арқылы артады. Процессор ядросы құрылышының қарапайым схемасы *4-сүретте* берілген.

Американдық Intel компаниясы өзінің алғашқы 8 ядролы Intel Core i7-5960X Extreme Edition процессорын ұсынды (*3-сүрет*). Бұл процессорда 1 млрд транзистор орналастырылған.



3-сүрет.

8 ядролы процессор



4-сурет. Процессор ядросы құрылышының қарапайым схемасы

Арифметикалық-логикалық құрылғы барлық математикалық және логикалық операцияны орындаиды.

Басқару құрылғысы процессордың командаларды тізбекпен орындаудың қамтамасыз етеді.

Кэш-жад – ақпаратты уақытша сақтауға арналған шағын көлемді құрылғы. Кэш-жадта жедел жадтың ең көп қолданылатын үяшықтары мен командалары сақталады.

Регисторлар – командалар мен процессорда өндөлген ақпараттарды сақтайтын, процессор ішінде орналасқан жад үяшықтары.

Сопроцессор – қалқымағы үтірі бар сандарды өндеуге арналған қосымша процессор элементі.

Процессор интерфейсі – микропроцессор мен сыртқы құрылғылар арасындағы деректер ағыны мен олардың көлемін (форматтарын) басқаратын құрылғылар.

Жүйелік шина – компьютердің барлық құрылғысын байланыстыратын ақпараттық магистраль.

Талдау

Процессор жұмысы мен адам миының жұмысын салыстыра отырып талдаңдар.



Жинақтау



Адамның жан дүниесінде болатын ақпараттық процестер жайлы ойыңды жаз. Ол жазбаца адамның қабылдаған ақпарат өсерінен күйзеліске ұшырауы, ренжуі немесе қуануы мысал болсын.



Бағалау



Процессордың компьютер жұмысын ұйымдастырудың маңызын бағалаңдар.



Ойлануға берілген сұрақтар

1. Процессордың қызметі қандай?
2. Процессор қандай параметрлерімен ерекшеленеді?
Оларды сипатта.
3. Процессордың құрылышын сипатта.
4. Көпядролы және бірядролы процессорлардың бір-бірінен қандай айырмашылығы бар?



Үй тапсырмасы

Зерттеудің тақырыптары	Зерттеудің сұрақтары мен бағыттары
Central Processor Unit	<ul style="list-style-type: none"> • Алғашқы процессорлардың пайда болуы мен даму тарихын зерттеу • Процессор шығаратын фирмалардың болашаққа бағытталған жұмыс мақсаттары қандай? • Процессорлардың жұмыс істеу принциптерін зерттеу
Бионика	<ul style="list-style-type: none"> • «Бионика» деген не? • Биониканың ғылым ретінде қалыптасу бағыттарын зерттеу • Жәнді табиғат пен техниканың үйлесімділігін зерттеу • Бионика ғылымының адамзатқа берер пайдасын зерттеу, болашақтағы дамуын болжау
Жан-жануарлар мен құстардың техникадағы модельдері	<ul style="list-style-type: none"> • Жәнді табиғатты қошірмелуеу, модельдеу • Құстар мен жануарлардың дene құрылышының ерекшеліктерін, тіршілігін зерттеу • Табиғаттың тірі құрылымдарының конструкциясын түрмиста, техникада т.б. салаларда пайдалану

1.5 Компьютерлік желілер



Желінің өткізу қабілетін анықтау үшін қандай есептеулер жүргізіледі?



Ойлан

- Компьютерлік желілер қандай негізгі параметрлермен бағаланады?
- Компьютерлік желілерді құру үшін қандай құрылғылар қажет?



Жаңа білім

Компьютерлік желіге пайдаланылатын құралдар

Компьютерлік желі (ағыл. *computer network*) – хабарламалармен алмасу және желінің ресурстарын бірлесіп пайдалану үшін байланыс арналары мен коммутация құралдары арқылы бір жүйеге қосылған компьютерлер жиынтығы.

Желілер шағын немесе үлкен кабельдер арқылы тұрақты жалғанған немесе телефон желілері мен сымсыз арналар арқылы уақытша жалғанған болуы мүмкін. Ең үлкен желі – интернет, ол – бүкіләлемдік желілер тобы.

Желілік коммутатор немесе **хаб, свич** (ағыл. *switch* – аудиостырып-қосқыш) – бірнеше компьютерді ортақ бір желіге біріктіретін құрылғы.



Желілік тақша компьютерлерді желіге қосуға мүмкіндік береді. Оның 10, 100, 1000 Мбит/с жылдамдықпен жұмыс істейтін түрлері бар.

Роутер сымсыз желіге қосылу құрылғысы интернетке жалғануға әрі кабель жоқ ортада желіде жұмыс істеуге мүмкіндік береді.

Патч-корд коммутатор бауы (шнур коммутатора) – электр құрылғыларын өзара байланыстыратын электрлік немесе оптоталшықты сымдар. Олар ұзын-

 **желінің өткізу қабілеті** – пропускная способность сети – **network bandwidth**
сенімділік – надежность – **reliability**

дығымен ерекшеленеді. **Патч-кордтың** екі үшінда **ағытпалы жалғағыш** (коннектор) орнатылған.

ADSL – әлемдегі ең көп тараған, интернетке қосылуға арналған құрылғы. **ADSL** аббревиатуrasesы ағылшынша «**Asymmetric Digital Subscriber Line**» – «Ассиметриялық цифрлық абоненттік желі» деген мағынаны білдіреді.

Бұлттық технология. Бұлттық есептеулер

Соңғы кезде ерекше дамыған жаңа технологиялардың бірі – **бұлттық технология**. Ағылшынша «*Cloud technology*» немесе «*Cloud computing*» (*Cloud* – бұлт; *technology* – технология; *computing* – есептеулер) «**бұлт/бұлттық есептеулер**» немесе «**бұлт/бұлттық технология, қызметтер**» деп те аталады. Бұлттық есептеудерді қарапайым сөзben былай түсіндіруге болады: тұтынушы өз компьютерінде белгілі бір программаны іске қосқанда, негізгі есептеулер мен ондағы дереккөздер интернеттегі шалғай серверлерде орындалып, сол жерде сақталады. Жұмыс нәтижесі тұтынушының компьютерінде стандартты веб-браузердің терезесіне шығарылып көрсетіледі.

Cloud computing-ті жүзеге асыратын серверлер **есептегіш бұлттар** деп аталады. Бұлттық технологияда жұмыс істеудің өдегетегі бағдарламалармен жұмыс істеуден айырмашылығы – тұтынушы өз компьютерінің ресурстарын емес, өзіне интернет-қызметі ретінде берілген шалғайдағы мықты серверлердің ресурстарын пайдаланады. Сол арқылы тұтынушы өз дереккөздерімен жұмыс істеуге толық мүмкіндік алады, бірақ сол дереккөздер орналасқан операциялық жүйеге, программалар базасына, есептегіш серверлердің жұмысына өсер ете алмайды.

Бұлттық технологияны пайдаланудың тиімді жақтары:

- үлкен ресурстарды қажет ететін қыын есептерді шешу үшін, тұтынушы өзінде жоқ серверлер мен программаларды бұлттардың көмегімен пайдаланады;
- тұтынушы кез келген жерден, қалаған уақытта интернетке қосылған компьютерлік құрылғымен жұмыс істей алады;
- тұтынушы компьютерлік құрылғының мүмкіндігіне төуелді болмайды;
- тұтынушы өз құжаттарымен еш қыындықсыз бөлісіп, әріп-тестерімен бірлесе жұмыс істей алады;
- білікті мамандар тапшылығын сол жобаны «**бұлтқа шығару**» арқылы шеше алады.

Көптеген компаниялар қазір «**булттар жарысына**» қосылып, өз қызмет түрлерін ұсынуда. Қазақстан нарығында жұмыс істейтін Ресей мен Беларусь компаниялары бар. Ал «**бұлттық шешімдер**» ұсынып жүрген отандық компаниялардың қызмет түрлері қазір даму үстінде.

Желінің өткізу қабілеті

Компьютер желісінің тиімді жұмысы төмендегі параметрлермен өлшенеді:

- **Желінің өткізу қабілеті** (деректерді тасымалдау жылдамдығы) уақыт бірлігі ішінде байланыс арнасы арқылы өтетін ақпараттың мөлшерімен анықталады. Өлшем бірліктері – Кбит/с, Мбит/с, Гбит/с.
- **Сенімділік** – ақпаратты айналаудан пайдалану өмір сүру қабілеті.
- **Құны** – желіге қосымша компьютерлерді жалғау мүмкіндігінің бар болуы.

Желінің өткізу қабілетін өлшеу бірліктері арасындағы қатынас ақпарат көлемін өлшеу бірліктерімен бірдей:

$$1 \text{ байт/с} = 2^3 \text{ бит/с} = 8 \text{ бит/с};$$

$$1 \text{ Кбит/с} = 2^{10} \text{ бит/с} = 1024 \text{ бит/с};$$

$$1 \text{ Мбит/с} = 2^{10} \text{ Кбит/с} = 1024 \text{ Кбит/с};$$

$$1 \text{ Гбит/с} = 2^{10} \text{ Мбит/с} = 1024 \text{ Мбит/с}.$$

Желінің өткізу қабілеті оның техникалық мүмкіндігі мен пайдаланылған кабельдерге байланысты.

1-кесте. Кабельдің техникалық сипаттамасы

Байланыс түрі	Өткізу қабілеті Мбит/с	Сенімділігі	Кеңейтілу мүмкіндігі
Электр кабельдері:			
Есулі қосақ	100-ге дейін	Тәмен	Қарапайым
Коаксиальді кабель	10-ға дейін	Жоғары	Құрделі
Телефон торабы	1-2	Тәмен	Қыындықсыз
Оптоалышқ кабель	100–200	Өте сенімді	Қыындықсыз

Кез келген **байланыс арнасының өткізу қабілеті** шектеулі болады. Бұл сан құрылғының қасиеті мен желіде пайдаланылған кабельге байланысты. Тасымалданған **ақпарат көлемі Q** төмендегі формууламен анықталады.

$$Q = q \cdot t$$

q – арнаның өткізу жылдамдығының қабілеті (өлшемі – секундтағы биттер саны)

t – тасымалдау уақыты



Талдау



Желінің өткізу қабілетін есептеуге мысалдар қарастырайық. Үлгі тапсырмаларды шығару жолдарына талдау жасаңдар. Ойларыңды ортаға салыңдар.

1-үлгі есеп

20 Кбит/с жылдамдықпен 5 секунд ішінде қандай ақпарат көлемін тасымалдауға болады?

Берілді:

$$q = 20 \text{ Кбит/с}$$

$$t = 5 \text{ с}$$

Табу керек: $Q - ?$

Шешімі:

$$Q = q \cdot t$$

$$Q = 20 \cdot 1024 \text{ бит} \cdot 5 = 20480 \text{ бит/с} \cdot 5 \text{ с} = \\ = 102400 \text{ бит} : 8 = 12800 \text{ байт} : 1024 = \\ = 12,5 \text{ Кбайт.}$$

Жауабы: 12,5 Кбайт.

2-үлгі есеп

Өлшемі 525 Кбайт файлды 33 600 бит/с жылдамдықпен қанша уақыт ішінде тасымалдауға болады?

Берілді:

$$q = 33600 \text{ бит/с}$$

$$Q = 525 \text{ Кбайт}$$

Табу керек: $t - ?$

Шешімі:

$$Q = q \cdot t \Rightarrow t = \frac{Q}{q} \quad 1 \text{ байт} = 8 \text{ бит}$$

$$Q = 525 \cdot 1024 \text{ байт} = 537600 \text{ байт}; \\ Q = 537600 \text{ байт} \cdot 8 = 4300800 \text{ бит}; \\ t = 4300800 \text{ бит} : 33600 \text{ бит/с} = 128 \text{ с.}$$

Жауабы: 128 с.

3-үлгі есеп

Өлшемі 1687,5 Кбайт файл 4 минутта тасымалданды. Желінің тасымалдау жылдамдығын анықта.

Берілді:

$$t = 4 \text{ минут} = 240 \text{ с}$$

$$Q = 1687,5 \text{ Кбайт}$$

Табу керек: $q - ?$

Шешімі:

$$Q = q \cdot t \Rightarrow q = \frac{Q}{t}$$

$$Q = 1687,5 \cdot 1024 \text{ байт} = 1728000 \text{ байт}; \\ Q = 1728000 \text{ байт} \cdot 8 = 13824000 \text{ бит}; \\ q = 13824000 \text{ бит} : 240 \text{ с} = 57600 \text{ бит/с.}$$

Жауабы: 57600 бит/с.



Ойлануға берілген сұрақтар

1. «Компьютерлік желі» деген не?
2. Компьютерде желіні ұйымдастыру үшін қандай құрылғылар керек?
3. Желілік коммутатордың қызметі қандай?
4. «Патч-корд» деген не?
5. Тасымалданған ақпарат көлемі қандай формуламен есептеледі?
6. Компьютер желісінің тиімді жұмысы қандай параметрлермен өлшенеді?
7. Желінің өткізу қабілетінде қолданылатын кабельдер қанша-лықты маңызды?



Тапсырма

1. Деректерді ADSL қосылысы арқылы тасымалдау жылдамдығы 256 000 бит/с. Осы қосылыс арқылы файлды тасымалдауға 16 секунд уақыт керек. Файлдың көлемін Кбайтта анықта.
2. Деректерді тасымалдау құрылғысының жылдамдығы 56 000 бит/с. Көлемі 210 000 байт файлды қанша уақытта жібереді?
3. Деректерді ADSL қосылысы арқылы тасымалдау жылдамдығы 2 621 440 бит/с. Осы қосылыс арқылы 10 Мбайт файлды тасымалдауға кететін уақытты секундпен есептеп тап.
4. Деректерді ADSL қосылысы арқылы тасымалдау жылдамдығы 5 242 880 бит/с. Осы қосылғыш арқылы өлшемі 1 Гбайт файлды тасымалдауға кететін уақытты минутпен есептеп тап.
5. Деректерді ADSL қосылысы арқылы тасымалдау жылдамдығы 128 000 бит/с. Осы қосылғыш арқылы 7 минутта тасымалданатын файлдың көлемін тап.
6. 14 400 бит/с жылдамдығы бар модем 225 Кбайт көлемдегі ақпаратты қанша уақытта тасымалдайды?
7. Деректерді ADSL қосылысы арқылы тасымалдау жылдамдығы 16 800 бит/с. 64 символдық алфавит көмегімен жазылған хабар 20 480 символдан тұрады. Осы ақпаратты желі арқылы тасымалдауға қанша уақыт қажет?



Үй тапсырмасы

1. Компьютерде кітаптың 128 беті терілді, әр бетте 32 қатар, әр қатарда 64 символ бар. Компьютерде 256 символдық алфавит қолданылады. Осы кітаптағы ақпарат көлемін 2 минут 8 секундта тасымалдаған тасымалдаушы құрылғының жылдамдығын анықта.
2. Жылдамдығы 28 800 бит/с модем өлшемі 640 · 480 пикселден тұратын түрлі түсті растрлық бейнені (түстің әр пикселі 3 байтпен кодталады) қанша уақытта тасымалдайды?

2.1 Компьютерді пайдаланудың кері аспектілері



Адам ағзасына электронды құрылғылардың қандай кері өсері бар? Олардан қорғанудың қандай тиімді тәсілдерін білесің?



Ойлан

- Адам ағзасына тұрмыстық техникадан келетін қандай жағымсыз өсерлердің айта аласың?
- «Электромагниттік сәуле» дегеніміз не? Оның адам ағзасына өсері қандай?
- Компьютерді шектен тыс пайдаланудың жағымсыз өслерлері жайлышы не білесің?



Жаңа білім

Электромагниттік сәулелердің адам ағзасына өсері

Электромагнит өрісі – материяның ерекше түрі, оның көмегімен электрлік зарядталған бөлшектер өзара өрекетке туседі. Жұмыс істеп тұрған электр құрылғылары молекула, электрондар, иондар мен атомдарды қозғалысқа өкеледі. Адамның жасушалары да молекулалардан тұратынын ескерсек, күшті электромагнит ағзада жасуша молекулаларының тербелісін, қозғалысын тудырады. Осындай қозғалыс ағза жасушаларының жағымсыз өзгерістеріне алып келеді.

Біз күнделікті өмірде қолданатын электр құралдарының денсаулығымызға қаншалықты зиянды екенін білмейміз. Электромагниттік толқындар кез келген үйде, мекемеде, жалпы адам жүретін барлық ортада бар. Электромагниттік толқындар тұрғын үйдің теледидарында, үтікте, мұздатқышта, микротолқынды пеш пен шансорғышта, компьютер мен үялды телефонда да болады.

Электромагниттік өрістер барлық тіршілік иелеріне, оның ішінде адам ағзасына да кәдімгідей зиян келтіреді. Биологиялық және медициналық зерттеу нәтижелері көрсеткендегі, электромагниттік



жағымсыз аспекті – негативный аспект – negative aspect
электромагниттік сәуле – электромагнитное излучение – electromagnetic radiation

өрістердің адамның жүйке жүйесіне, ішкі мүшелеріне, физиологиялық дамуына тигізетін теріс зардалтары бар. Сонымен қатар кейінгі кезде жұргізілген зерттеулер нәтижесінде электромагниттік өрістердің өсерінен көптеген ауру түрлерінің пайда болатыны анықталған. Электромагниттік өрістер адамның ми құрылышына өсер ете отырып, ойлау мен есте сақтау қабілетін бұзады. Электромагниттік толқын көздерінің сәуле шығару көрсеткіштері 1-кестеде берілген. Адам үшін электромагниттік сәулені қабылдаудың шекті мөлшері 0,2 мкТл-дан (микротесла) аспауы керек.

1-кесте. Электромагнит сәуле шығару көздерінің көрсеткіштері

Электромагниттік толқын тарататындар	Сәуле шығару көрсеткіші (мкТл)	Қалыптан тыс (есе)
Компьютер	1–100	5–500
Ұялы телефон	40	200
Тоңазытқыш	1	5
Электр шәйнек	10	50
Микротолқынды пеш	8–100	40–500
Фен	15–17	75–85
Трамвай мен троллейбус	150	750
Метро	300	1500

Негізінен, компьютердің алдында ұзақ уақыт отырып жұмыс істейтін адамдардың басы жиі ауырып, көру қабілеті нашарлайды.

Компьютерден таралатын электромагниттік толқынның адам ағзасына зияны көп.

Мысалы:

- көздің көру қабілеті төмендейді;
- омыртқаның қисаюына әкеледі;
- жүйкеге салмақ түседі;
- адам тез шаршап, денені әлсіздік басады;
- адам көп отырған кезде қан айналымы бәсендейді, иммунитет төмендейді.

Осындай ауруларға шалдыққың келмесе, компьютердің алдында көп отырмауга тырыс. Арнайы ережелерді сақтап, жаттығу жасап отыру қажет. Өз денсаулығыңа көп көңіл бөл.

Компьютер экранының электромагниттік сәулеленуінің әсері

Жұмыс істеуге отырғанда сағат сайын демалыс үйымдастырып, жаттығу жасау керек. Ноутбуқты тізеге қойып отыруға болмайды. Құшті радиация тарайтындықтан, адам әртүрлі ауруларға шалдығуы мүмкін. Компьютер мен ноутбук пайдаланғанда көзге қорғаныс көзілдірігін киу қажет. Мысалы, суға түсіп келгеннен кейін компьютердің алдында отырса, радиацияны қабылдау мөлшері біршама азаяды екен. Компьютер немесе ноутбук сатып аларда «S» белгісінің бар-жоғына назар аудару керек. Бұл белгі – компьютер мен ноутбукта радиацияға қарсы қорғаныстың бар екенінің дәлелі. Есінде болсын, компьютердің жанына немесе артына отыруға болмайды, оның радиацияны ең көп тарататын бөлігі – сол тұстары. Компьютерді орналастырғанда артқы жағын терезеге немесе қабырға жаққа қаратқан жән, одан таралған радиация сыртқа кетеді. Компьютер мен ноутбуктан жылы бу шығады, оның әсерінен адам ағзасындағы кальций мөлшері азаяды. Соның әсерінен тырнақ сынғыш, шаш түскіш келеді. Компьютер, ноутбук пен теледидар сияқты тұрмыстық заттар орналасқан бөлмелерде мүмкіндігінше пластик терезелер емес, ағаш терезелер болғаны дұрыс. Себебі бөлмедегі радиацияны пластик терезе сыртқа шығармай, тосқауыл болады. Үйықтап жатқанда үялы телефондарды жақын жерде қалдырма. Дүниежүзілік денсаулық сақтау үйымы үялы телефондар ғана емес, жалпы барлық электронды құрылғы ағзаға зиянды екендігін айтады.



Компьютерді пайдаланудың негізгі ережелері

- Монитор экраны көзден 50, тіпті 70 сантиметр қашықтықта орналасуы керек.
- Экранның жоғарғы бөлігі көздің тұсынан төмен тұрсын.
- Жұмыс істейтін бөлмеде бірнеше компьютер болған жағдайда, олардың арақашықтығы 1 метрден кем болмасын.
- Компьютердің арт жағы қабырғаға қарап тұруы керек.

- Компьютер орнатылған бөлмеде ылғалды тазалау күн сайын жүргізіліп, бөлме күніне бірнеше рет желдетілуі тиіс.
- Пернетақта мен тінтуірді күн сайын арнайы суланған микроталшық матамен сұрту керек. Осылайша бактериялардың көбеюіне жол берілмейді.
- Компьютер алдында әрқашан омыртқаға зиян келмейтіндей дұрыс отыр.
- Үзіліс кезінде көзге жаттығулар жаса: көзді жоғары-төмен, онға-солға айналдыра қозғалт, алысқа көз жібер, терезеге қара, сонаң соң көзді жақын маңайға бағытта.
- Экран бетіне түсіп тұратын сөүлелі шағылысы дақтарын жою керек. Мұндай сөүлелі шағылысы дақтары жоғарыда, артта орналасқан люстрадан, терезеден түскен жарықтан, үстел шамынан түсіү мүмкін. Сол себепті мониторды сөндіріп, экран бетінің шағылыспай, қалыпты жағдайда түрған-түрмағанын тексеріп ал. Егер сөүлелі шағылысы дақтары байқалса, экран алдына айна қойып, оның қайдан түсіп түрғанын анықта.
- Компьютердің алдында жұмыс істеген оқушы әрбір 20 минуттық жұмыстан соң 15 минут демалуы тиіс. Ересек адамдар 2 сағаттан соң 10–15 минут демалғаны жөн.

Аспект (лат. *aspectus* – көзқарас) – зерттеліп жатқан түсінікке, затқа немесе құбылысқа деген әрқылы көзқарас.



Талдау

Күнделікті өздерің пайдаланып жүрген электр құрылғыларының ішінен 3 құрылғыны таңдап алыңдар. Осы құрылғылардың адам ағzasына өсерін анықтап жазыңдар.



Жинақтау

Компьютердің адам ағzasына өсер ететін жағымсыз аспектілерін анықтау мақсатында, 10 сұрақтан тұратын сауалнама ұсыныңдар.



Бағалау



Компьютердің адам ағзасына кері өсерін болдырмаудың 5 жолын көрсетіп, дәлелді мысал келтір. Себебін түсіндір. Қорытынды есеп жаз.



Ойлануға берілген сұрақтар

1. Электромагнитті сөүленің адам ағзасына өсері қандай?
2. Электромагнитті сөүленің кері өсерінен пайда болатын қандай ауруларды атайды аласың?
3. Компьютерден тарайтын электромагниттік толқынның ағзаға қандай жағымсыз өсері бар?
4. Компьютердің адам ағзасына кері өсерімен қалай күресу керек?



Үй тапсырмасы

№	Зерттеу тақырыптары	Зерттеу сұрақтары мен бағыттары
1	Компьютердің адам денсаулығына өсері	<ul style="list-style-type: none"> • Компьютердің адам ағзасына өсері бар ма? • Компьютердің алдында қанша уақыт отыруға болады? • Компьютердің адам ағзасына жағымсыз өсерін қалай төмендетуге болады?
2	Компьютерлік ойындарды ойнау пайдалы ма, зиянды ма?	<ul style="list-style-type: none"> • Компьютер ойындарын ойнаудың оқушы ағзасына қандай өсері бар? • Ойын ойнауда компьютердің зиянды өсерін төмендетудің қандай жолдары бар?
3	Күнделікті қолданыстағы электр құрылғыларының ағзага өсері	<ul style="list-style-type: none"> • Электромагниттік сөүленің ағзаға қандай өсері бар? • Қолданыстағы электр құрылғылары ағзага қаншалықты қауіпті? • Электр құрылғыларының адам ағзасына жағымсыз өсерін қалай төмендетуге болады?

2.2. Желідегі қауіпсіздік

 Желі қолданушыларының қауіпсіздігін қамтамасыз ету ережелерін білесің бе?

Ойлан

- «Ақпараттық қауіпсіздік» дегенді қалай түсінесің?
- Интернетте жұмыс істегендеге қандай қауіпсіздік ережелерін сақтайсың?
- Интернеттегі жұмыс барысында қандай қауіп-қатер туындауы мүмкін?
- Интернеттегі алайқтық жайлы не білесің?

Жаңа білім

Ақпараттық қауіпсіздік

Ақпараттық технологиялардың даму деңгейі мен барлық саланы цифрандыру үрдісі мемлекеттік органдардың, банктер мен жеке компаниялардың алдында ақпараттық қауіпсіздік мәселесін ең өзекті мәселе етіп қойып отыр.

Ақпараттық қауіпсіздік мемлекеттік ақпараттық ресурстардың, сондай-ақ ақпарат саласында жеке адамның құқықтары мен қоғам мүдделері қорғалуының жай-күйін сипаттайды.

Ақпараттық қорғау – ақпаратты ұрлау, жогалту, рұқсатсыз жою, өзгерту, маңызына тимей түрлендіру, рұқсатсыз көшірмесін жасау мен бұғаттаудың алдын алу үшін жүргізілетін шаралар кешені.

Ақпараттық қауіпсіздіктің маңызды З жайын атап кетуге болады.

1. Қолжетімділік – санаулы уақыт ішінде керекті ақпараттық қызмет алуға болатын мүмкіндік. Ақпараттың қолданысы – ақпараттың, техникалық құралдар мен өндеу технологияларының ақпарат-қа кедергісіз (бөгетсіз) қол жеткізу қабілетімен сипатталатын қасиеті.

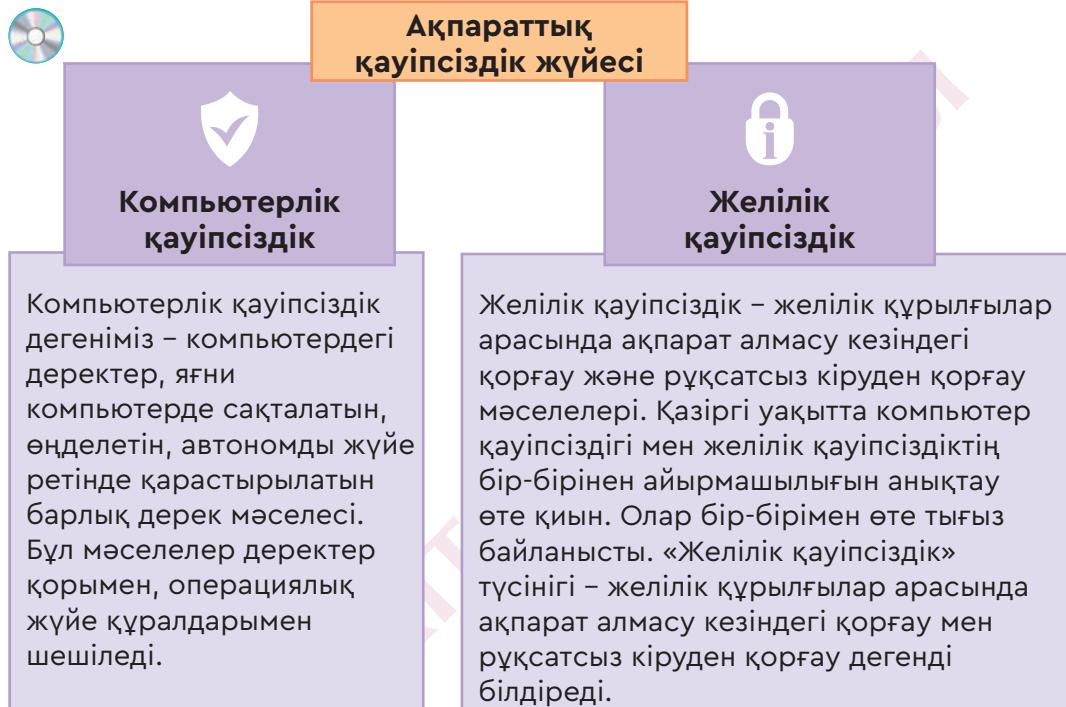


желідегі қауіпсіздік – сетевая безопасность – network security
ақпараттық қауіпсіздік – информационная безопасность – information security

2. Тұтастық – ақпараттың бүрмаланудан және заңсыз өзгертуден қорғалуы. Ақпарат кездейсоқ немесе әдейі бүрмаланған (бұзылған) кезде есептеу техника құралдары ақпараттың өзгермеуін қамтамасыз етеді.

3. Құпиялыштық – заңсыз қол жеткізуден немесе оқудан қорғау.

Ақпараттық қауіпсіздік жүйесі 2 бағытта қарастырылады (1-схема).



1-схема. Ақпараттық қауіпсіздік

Интернет желісіндегі алаяқтық

Қазіргі таңда интернет желісін, өлеуметтік желілерді қолданбайтын адам аз. Уақыт талабына сай мұндай өлеуметтік желілер адамдар арасында қарым-қатынас орнатуға ғана емес, ауқымды мәселелерді талқылауға, белгілі бір оқиға не құбылыс туралы қоғамдық пікір қалыптастыруға ықпалы зор, үлкен қүшке айналып отыр.

Интернетті теріс піғылдарын жүзеге асыру үшін пайдаланатындар да көп. Желіде де өмірдегі сияқты, басқалардың ақшасын оңай



иеленгісі келетін алаяқтар пайда болды. «Интернетте кездесетін алаяқтықтың қандай түрлері бар, олардан қалай қорғанамыз?» деген сұраққа жауап іздеп көрейік. Интернетте алаяқтықтың неше түрлі жолдары кең етек жайып келеді. Олардан сақтану үшін, алаяқтық түрлерінен үнемі хабардар болып отыруымыз керек. Енді солардың кең тараған түрлеріне тоқталайық.

Лотерея жеңімпазы немесе алыс туысқанның мұрагері. Электрондық поштаңа әртүрлі халықаралық бағдарламалардың атын жамылған күмәнді үйимнан не жеке адамнан «Лотерея ойынының өте үлкен сомасының иегері болдың» деген хабар келеді. «Қайтыс болғалы жатқан шетелдік бай туысқанызыз сізге мұра қалдырмақшы» деген де хат түседі. Үтысыңды немесе мұранды алу үшін, құжаттарды рәсімдеуге ақша талап етеді.

Сикырлы өміяндар – желідегі ең көп тараған алаяқтық тәсіл. «Электронды өміянға N көлемдегі ақша аударсаң, екі-үш есе көп ақша қайтарып аласың» деген хабарламаға сеніп, ақша аударасың. «Аз мөлшердегі ақшаны екі есе етіп қайтарып алуың мүмкін», – деп те алдайды. Ақша салсан, ақшан қолды болады.

Жұмыс немесе ақша табуды ұсыну – бүгінде желіде кең тараған алаяқтықтың бірі. Мысалы, «мәтін теруші» деген жұмысқа жалақы ұсынады. Бірақ жұмысқа тіркелу үшін, деректер енгізуге үлкен белгілі бір мөлшерде ақша сұрайды. Сұраған ақшасын аударсаң, жұмыс та жоқ, ақша да жоқ.

Интернетте ақша табуды үйретудің керемет өдісі – ескі, бірақ кең тараған өдістердің бірі. Алаяқтықтың бұл түрі желіде ақша табуға болатын өдістерді үйрететін сайттар тізімін ұсынады. Белгілі бір сомадағы ақшаны аударсаң, олар сол өдістерді үйретуге дайын. Ақша аударылғаннан кейін ақша да, өдіс те жоқ екенін байқаймыз.

Қайырымдылық акциялары. Бұл да өлеуметтік желідегі алаяқтықтың кең тараған түрі. Мұнда алаяқтар ауыр дертке шалдықсан, шаңырағында қайғылы жағдай болып, қаражатсыз қалған адамдар мен отбасылар туралы ақпарат жарияладап, елден көмек сұрайды.

Жоғарыда аталған алаяқтарға алданбау үшін әрбір адам желіде барынша сақ болуы тиіс. «Жаңа таныстарды» мұқият тексермей, ортақ таныстардың бар-жоқтығына көз жеткізбей тұрып олармен сөйлеспе. Жеке өміріңе қатысты суреттер мен бейнежазбаларды өлеуметтік желіге жарияламауға тырыс, сұхбаттарды ізін сұытпай өшіріп отыр.

«Кибербуллинг» дегеніміз не?

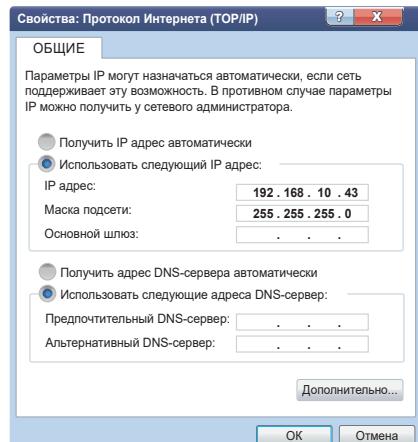
Кибербуллинг (*cyberbullying*) – жас-өспірімдер арасында кең тараған виртуалды лаңкестік (террор). Кибербуллинг (ағыл. *bull* – «бұқа» сөзінің мағынасына жақын үғым) шабуылдау, мәжбүрлеу, айдап салу, қорқыту, ар-намысина тиу, жалған ақпарат тарату, бопсалау сияқты агрессивті үғымдарды білдіреді. Кибербуллинг өте қауіпті, өйткені интернет арқылы әлеуметтік желілерде ар-намысқа тиу, қауесет тарату, қорқыту мен масқаралап беделін түсіру арқылы адамға психологиялық қысым жасалады.



Интернет желісінде жасалған алайқтық, алдау мен кибербуллинг жазасыз қалмайды. Мұндай қылмыс жасағандар заң алдында жауапкершілікке тартылады. Интернетте миллиардтаған пайдаланушы болса да, оларды табу оңай, себебі әрбір компьютер интернетке қосылғанда оған арнайы IP-адрес беріледі.

«IP-адрестеу» деген не?

IP-адрес (*Internet Protocol Address*) – интернетке шыққан компьютердің адресі (мекенжайы). **Провайдер** (*Интернетті пайдалануга руқсат беретін үйім*) әр интернет қолданушыға қайталанбас IP-адресін береді. Интернет желісіне қосылған әрбір пайдаланушының өзіндік қайталанбас физикалық адресі болады (IP-адресі). Адрес нүктемен бөлінген 4 ондық саннан тұрады, олардың әрқайсысы – 0 мен 255 санының арасындағы кез келген сан. Желідегі компьютерді осы сандарға қарап анықтайды. Мысалы: **192.168.10.43** немесе **128.29.15.124** (*1-сүрет*). Осылай адрестеу жүйесі бойынша миллиардтаған компьютерге жеке адрес беруге болады.



1-сүрет. Интернет протоколы



Талдау



Интернетте тағы қандай алайқтықтар мен қауіп-қатерлер бар? Бірлесіп талдандар. Ойларыңды ортаға салындар.



Жинақтау



Кибербуллингтің өсіреле жасөспірімдерге қаншалықты қауіпті екенін білдіңдер. Кибербуллингке қалай қарсы тұруға болады? Кибербуллингке қарсы күресудің ережесін жасандар. Ережелеріңің әрбір пунктіне дәлелді мысал келтіріңдер.



Бағалау



Интернет желісін пайдалану барысында қауіпсіздік шараларын сақтаудың қаншалықты маңызды екенін бағамда.



Ойлануға берілген сұрақтар

1. «Ақпараттық қауіпсіздік» деген не?
2. Ақпараттық қауіпсіздіктің маңыздылығы неде?
3. Ақпараттық қауіпсіздік жүйесінің қандай бағыттары бар?
4. «Желідегі алаяқтық» деген не?
5. Кибербуллингтен қандай қауіп бар?
6. Интернетте алаяқтық жасағандарды анықтауға бола ма?
7. «IP-адрестеу» деген не?



Үй тапсырмасы

№	Зерттеу тақырыптары	Зерттеу сұрақтары мен бағыттары
1	Интернет желісіндегі қауіпсіздік шаралары	<ul style="list-style-type: none"> • Интернет желісінде пайдаланушы білуі тиіс негізгі ережелер • Интернетте ақпараттық қорғаудың жолдары
2	Әлеуметтік желілерді қауіпсіз пайдалану	<ul style="list-style-type: none"> • Әлеуметтік желілер мен олардың ерекшеліктері • Пайдаланышыға әлеуметтік желілерде кездесетін қауіп-қатерлер
3	Интернеттің маңызы	<ul style="list-style-type: none"> • Интернет желісінің пайда болу тарихы • Интернетті пайдаланудың жағымды, жағымсыз аспектілері • Интернетті қауіпсіз пайдалану бойынша ұстанатын қағидалар қандай?

3.1

Статистикалық деректер. Абсолютті және салыстырмалы сілтемелерді қолдану



Статистикалық деректерді өңдеуде абсолютті және салыстырмалы сілтемелерді қалай қолдануға болады?



Ойлан

- «Статистикалық дерек» деген не?
- Статистикалық деректермен қандай мамандар жұмыс істейді?



Жаңа білім

«Статистикалық дерек» дегеніміз не?

Тарихқа үцілсек, ежелгі заманда да мемлекеттер өз халқының саны мен оның өскерге жарайтын бөлігін, мал базы мен жердің көлемін анықтау мақсатында есеп жүргізіп отырған. Бұл деректерді салық жинау, соғысқа дайындық сияқты мақсаттарда пайдаланған. Өркениеттің дамуы нәтижесінде есепке алып отыратын жаңа бағыттар пайда болған. XVII ғасырдың ортасында Ұлыбританияда «Саяси арифметика» деген ғылыми бағыт пайда болды. Бұл қоғамда болып жатқан түрлі өзгерістерді есепке ала отырып, оның дамуына талдау жүргізуге бағытталған. «Саяси арифметика» бағытымен қатар, Германияда «Міндettі статистика» мектептері ашылды. Осындай мектептердің арқасында ғылымда жаңа бағыт – **статистика** пайда болды. **Статистика** (лат. *status* – жай-күй, құбылыстар реті) – үлкен ғылым.

Қоғамның даму барысын зерттеуде жиналған ақпараттың көлемі өте үлкен. Сол ақпаратты жинақтап, реттеп, өндеп өрі талдап отыру қажет. Осы жұмыстарды электронды түрде атқаруда Excel кестелік процессорын пайдалану жолдарын қарастырайық.

Абсолютті және салыстырмалы сілтемелерді қолдану

Сілтеме – формула жазу барысында қолданылатын ұяшық, қатар, бағандардың адресі.



салыстырмалы сілтеме – относительная ссылка – relative reference
абсолютті сілтеме – абсолютная ссылка – absolute reference

Кестелік процессордың ең маңызды қызметі – ұяшықта орналасқан формулаларды басқа ұяшыққа көшіріп өткізу. Біз 7-сыныпта сілтемелердің **салыстырмалы, абсолютті** және **аралас болатыны** жайлы білгенбіз. Енді осы сілтемелерді пайдалану жолдарымен танысайық. Excel програмmasында үнсіздік бойынша салыстырмалы сілтеме қолданылады, яғни формуланы көшіргенде сілтемелдердегі адрестер алғашқы ұяшық пен көшірменің орналасуына сәйкес автоматты түрде өзгереді. Көшірмеледе ұяшық адрестері өзгермеуді үшін абсолютті сілтеме қолданылады, ол жолдың немесе бағанның атауы алдына (аралас сілтемеде) немесе екеуінің алдына да \$ белгісін қоюмен ерекшеленеді.

Салыстырмалы сілтеме. Формулаларды бір ұяшықтан екіншісіне көшіргенде нәтиже қандай болатыны сілтеменің адрестеуіне тікелей тәуелді. Қалыпты жағдайда формуладағы ұяшықтарға сілтеме – салыстырмалы сілтеме. Бұл формулаларды бір ұяшықтан екінші ұяшыққа көшіргенде сілтеменің адресі автоматты түрде өзгереді. Мысалы, B2 ұяшығында одан бір жол төмен орналасқан B3 ұяшығына сілтеме орналассын делік. Егер осы формула көшірілсе, онда сілтеменің салыстырмалы көрсеткіші сақталады. Мысалы: A9, D25, F5 – салыстырмалы сілтеме.

Абсолютті сілтеме. Абсолют адрестеу кезінде формула көшірілгенде сілтеменің адресі өзгермейді, сілтеме көрсетіп тұрған ұяшық тұрақты болып қалады. Формулаларды өңдеу кезінде адрестеу әдісін өзгерту үшін ұяшықтың сілтемесін ерекшелеп алғып, F4 пернесін басу керек. Абсолют адресте ұяшық нөмірінің алдына \$ белгісі қойылады. Мысалы: \$A\$16, \$A9, A\$7. Соңғы екі жағдайда ұяшық нөмірлерінің бірі абсолют, екіншісі салыстырмалы болып есептеледі. Бұл – **аралас сілтеменің мысалдары (1-кесте).**

1-кесте. Сілтемелерді пайдалану мысалдары

Ұяшықтардағы сілтеменің түрлері	Келесі ұяшықтағы сілтеме	
	Оңға қарай толтыру кезінде	Төменге қарай толтыру кезінде
A1 (салыстырмалы)	B1	A2
\$A1 (баған бойынша абсолют) аралас адресті	\$A1	\$A2
A\$1 (жол бойынша абсолют) аралас адресті	B\$1	A\$1
\$A\$1 (абсолют)	\$A\$1	\$A\$1

Енді сілтемелермен жұмыс істеуді бекіту мысалдарын, сандар мен формулалар жазылған электрондық кесте фрагментін қарастырайық. Осы кесте сілтемелермен жұмыс барысында қалай өзгеретінін байқайык.

2-кесте. Абсолютті және салыстырмалы сілтемелерді пайдалану

Абсолютті сілтеме		Салыстырмалы сілтеме		
№	A	B	C	D
1	5	=A1*\$B\$2	=A1*B1	
2		8		=B2*C2
3				

B1 үяшығында $=A1*\$B\2 формуласы орналассын. Формуланы теріп болғаннан кейін Enter пернесін басқанымызда **нәтиже = 40** болды. Бұл жерде A1 үяшықтағы сан B3 үяшыққа көбейтіліп, B1 үяшығына жүктелді. Енді осы формуланы C2 үяшығына көшіріп қой. Қандай өзгеріс болады еken? Бұл сұраққа жауап беру үшін үяшықтардағы сілтемелерді талдайык.

A1 үяшықтағы сілтеме салыстырмалы, демек формуланың орны өзгергенде, бұл сілтеме де өзгереді.

B2 үяшығындағы сілтеме абсолютті, демек формула орны өзгергенмен, бұл үяшық өзгермейді.

Формуланы C2 үяшығына көшіріп қойғанда, ол $=B2*\$B\2 болып ауысты. Байқаған болсаң, A1-дің орнына көшірме алынған B2 үяшықтың сілтемесі жазылды. Enter пернесін басқанда **нәтиже = 64 болды**.

3-кесте. Формула үяшығына көшіріп қою

№	A	B	C	D
1	5	=A1*\$B\$2		
2			8 = B2*\$B\$2	
3				
4				
5				

Себебі не? Формуланы оңға қарай бір үяшыққа жылжытқанда, A1-дің орнына B1 болып ауысады. Enter-ді басқанда нәтиже пайда болады. Егер осы үяшықтағы $=C3*\$B\2 формуланы D3-ке қойсақ, нәтиже = 0, себебі D3 үяшықта сан жоқ, ол нөлге тең болады.

Егер B1 үяшықтағы формуланы A4-ке қойсақ, онда қандай нәтиже аламыз?

Егер В1 үяшықтағы формуланы А4-ке қойсақ, қандай нәтиже аlamыз?

4-кесте. Үяшықты көшіріп қою арқылы формула алу

№	A	B	C	D
1		5 =A1*\$B\$2		
2			8 = B2*\$B\$2	
3				=C3*\$B\$2
4	=A1*\$B2			
5				

Енді практикалық түрғыдан алынған нақты мысалдарды қарастырайық.



Практикалық жұмыс

1-мысал. Жұмысшылардың айлық жалақысы туралы кесте берілген. Айлық жалақы 10%-та артқаннан кейін, жұмысшылардың айлық жалақысы қанша болады?

Айлық жалақы 10%-та артқанда жалақының сомасын 1,1 ($100\% + 10\% = 110\% / 100\%$) коэффициентке көбейту қажет. 1-суретте жалақыны есептеудің формуласы D5 үяшығында берілген. Егер бұл формуланы басқа үяшықтарға көшіріп өткізсек, онда есептеулер дұрыс жүрмейді. Себебі кейінгі үяшыққа өткенде формула сілтемесі өзгеріп, басқа нәтижені көбейтеді (2-сурет).

	A	B	C	D
1		Жұмысшылардың айлық жалақысы		
2		Жалақының көтерілу пайызы 10%		1,1
3	№	Жұмысшылар	Қазіргі жалақысы	Жалақысы оскен соң
4	1	№1 жұмысшы	100000	110000
5	2	№2 жұмысшы	120000	#ЗНАЧ!
6	3	№3 жұмысшы	110000	12100000000
7	4	№4 жұмысшы	150000	#ЗНАЧ!
8	5	№5 жұмысшы	130000	1,573E+15
9	6	№6 жұмысшы	80000	#ЗНАЧ!
10	7	№7 жұмысшы	105000	1,65165E+20

СУММ			
A	B	C	D
1	Жұмысшылардың айлық жалақысы		
2	Жалақының көтерілу пайызы 10%		1,1
3	№	Жұмысшылар	Жалақысы оскен соң
4	1	№1 жұмысшы	100000 =C4*D2
5	2	№2 жұмысшы	120000
6	3	№3 жұмысшы	110000
7	4	№4 жұмысшы	150000
8	5	№5 жұмысшы	130000
9	6	№6 жұмысшы	80000
10	7	№7 жұмысшы	105000

1-сурет. Жұмысшылардың жалақысы

2-сурет. Жалақыны есептеу

Жұмысшылардың айлық жалақысын автоматты түрде дұрыс есептеу үшін, D2 үяшықты абсолютті сілтемеге ауыстырамыз

(3-сүрет). Абсолютті сілтемеден қейін жалақының автоматты тұрде 10%-ға артқан кестесің көресің (4-сүрет).

	СУММ		
A	B	C	D
1	Жұмышшылардың айлық жалақысы		
2	Жалақының көтерілу пайызы 10%		1,1
3 №	Жұмышшылар	Қазіргі жалақысы	Жалақысы оскен соң
7 1	№1 жұмышшы	100000	110000
5 2	№2 жұмышшы	120000	132000
6 3	№3 жұмышшы	110000	121000
7 4	№4 жұмышшы	150000	165000
8 5	№5 жұмышшы	130000	143000
9 6	№6 жұмышшы	80000	88000
10 7	№7 жұмышшы	105000	=C10*\$D\$2

3-сүрет. Ұяшықтардың абсолютті мекенжайы

	СУММ		
A	B	C	D
1	Жұмышшылардың айлық жалақысы		
2	Жалақының көтерілу пайызы 10%		1,1
3 №	Жұмышшылар	Қазіргі жалақысы	Жалақысы оскен соң
4 1	№1 жұмышшы	100000	=C4*\$D\$2
5 2	№2 жұмышшы	120000	
6 3	№3 жұмышшы	110000	
7 4	№4 жұмышшы	150000	
8 5	№5 жұмышшы	130000	
9 6	№6 жұмышшы	80000	
10 7	№7 жұмышшы	105000	

4-сүрет. Жалақыны автоматты тұрде есептеу



Талдау



1, 2-тапсырмалардың орындалу жолдарын талдаңдар әрі салыстырындар. Талдау барысында төмендегі сұрақтарға жауап іздендер.

- Екі тапсырманың бір-бірінен қандай айырмашылығы бар?
- Тапсырмаларды орындау барысында қандай сілтемелер қолданылған?



Ойлануға берілген сұрақтар

- «Статистикалық дерек» деген не?
- Статистикалық деректер не үшін жинақталады?
- «Сілтеме» дегеніміз не?
- «Салыстырмалы сілтеме» деген не?
- Абсолютті сілтеменің салыстырмалы сілтемеден қандай ерекшелігі бар?
- «Аралас сілтеме» деген не?
- Сілтемелердің қандай пайдасы бар?



Тапсырма



1-тапсырма. Берілген ұлгі бойынша есептеу жүргіз. Кесте жаса.

5-кесте. Тауардың жалпы бағасын есептеу

	A	B	C	D	E
1	№	Tауардың аты	Саны	Бағасы	Бағасы
2	1	Дәптер	50	15	=C2*D2
3	2	Қаламсал	80	50	
4	3	Қалам	25	6	
5	4	Өшірғіш	10	0,45	
6	5	Сызғыш	20	8,15	

- a) Е2 үяшығына дәптердің бағасы мен санының көбейту формуласын жаз: =C2 * D2.
- ә) Е бағанындағы қалған үяшықтарға формуланы көшпіріп қой. Кестені пішімдеп, сақта.

2-тапсырма. Қызметкерлердің негізгі жалақысын есептеуде оның жұмыс коэффициентін еліміз бойынша белгіленген базалық жалақы мөлшеріне көбейту керек (*5-сурет*). Кестедегі деректер бойынша тапсырманы аяқта.

A	B	C	D
1			
2			
3			
4		Қызметкерлердің жалақысы қанша?	
5			
6		Базалық жалақы мөлшері =	17697
№	Қызметкер тізімі	Коэффициент	Жалақы
7 1	№1 жұмысшы	4,8	84945,6
8 2	№2 жұмысшы	3,9	
9 3	№3 жұмысшы		
10 4	№4 жұмысшы		
11 5	№5 жұмысшы		

5-сурет. Негізгі жалақыны есептеу

3-тапсырма. 1-ден 10-ға дейінгі сандар арқылы Пифагордың көбейту кестесін жаса (*6-кесте*).

6-кесте. Көбейту кестесі

	1	2	3	4	5	.	.
1	1	2	3	.	.		
2	2	4	6	.	.		
3	3	6	9	.	.		
4		
5			
.							
.							

3.2

Абсолютті және салыстырмалы сілтемелерді практикалық түрғыдан қолдану



Әртүрлі типтерді деректерді өндеуде абсолютті және салыстырмалы сілтемелерді қалай қолдануға болады?



Ойлан

- Excel-де есептерді шешу барысында абсолютті және салыстырмалы сілтемелерді қолданудың артықшылықтары қандай?



Жаңа білім

Абсолютті және салыстырмалы сілтемелермен тек бір файлмен емес, бірнеше файлмен бір уақытта жұмыс істеуге болады. Біріншісі екіншісіне сілтеме жасалған екі файл болсын. Оның біріншісіне жасалған өзгерістің екінші файлдағы деректерге әсері бола ма? Осыны анықтайық.

Мысал қарастырайық. Тауарлардың құны АҚШ долларында жазылған прайс-парағымыз болсын. Доллар бағамының өзгеруіне байланысты тауардың теңгедегі бағасы өзгеріп отырады. 1-файлда тауарлардың доллардағы бағасы мен теңгедегі бағасы берілген (*1-сурет*). Бұл жерде доллардың теңгеге

шаққандағы бағамы жазылған C2 – абсолютті сілтемелі үяшық.

Доллар бағамы өзгерсе, тауарлардың да құны автоматты түрде өзгереді. 2-файлда бір тоқсан бойы сатылған тауарлардың саны берілген (*2-сурет*). Тауардың бағасын 1-файлда көрсетілген доллар бағамының өзгеруіне төуелді ету үшін, төмендегі әрекеттерді орындаімыз.

2-файлдағы №1 тауардың бағасы жазылған C4 үяшығына '=' белгісін қойып, 1-файлдағы №1 тауардың бағасы жазылған

A	B	C	D
1			=C5*\$C\$2
2			
3			
4 №	Тауардың аты	Бағасы \$	Бағасы ₸
5 1	№1 тауар	20	6400
6 2	№2 тауар	40	12800
7 3	№3 тауар	100	32000
8 4	№4 тауар	250	80000
9 5	№5 тауар	300	96000
10 6	№6 тауар	400	128000

1-сурет. Тауарлар бағасының кестесі



статистика – статистика – statistics
сілтеме – ссылка – reference

D4 ұяшығын шертіп, Enter пернесін басқанда 2-файл мен 1-файл арасында байланыс орнайды. С4 ұяшығының сілтемесі өзгереді ‘=[1.xlsx]Лист1!\$D\$4’. Енді 1-файлдағы тауардың бағасы доллар бағамына қарай өзгерсе, 2-файлда да тауар бағасы өзгереді.

	A	B	C	D	E	F	G	H	I	J
1	Тоқсандағы жалпы сатып алғынған тауар құнының есептеу									
2	№	Тауардың аты	Бағасы (т) –	Сатылған тауар құны			Сатылған тауар сомасы			Жалпы сомасы
3				қаңтар	ақпан	наурыз	қаңтар	ақпан	наурыз	
4	1	№1 тауар		20	30	80	0	0	0	0
5	2	№2 тауар		30	20	70	0	0	0	0
6	3	№3 тауар		40	50	90	0	0	0	0
7	4	№4 тауар		60	40	30	0	0	0	0
8	5	№5 тауар		80	50	20	0	0	0	0
9	6	№6 тауар		40	50	70	0	0	0	0

2-сүрет. Бір тоқсанда алғынған тауарлардың жалпы саны

2-файлдың С4 ұяшығындағы формууланы басқа ұяшықтарға көшірсек, басқаларға да сол ұяшықтағы 1-тауардың бағасы ауысады. Бұлай болдырмай үшін сілтемеге өзгеріс жасауымыз қажет (3-сүрет).

	A	B	C	D	E	F	G	H	I	J
1	Тоқсандағы жалпы сатып алғынған тауар құнының есептеу									
2	№	Тауардың аты	Бағасы, т	Сатылған тауар саны			Сатылған тауар сомасы			Жалпы сомасы
3				қаңтар	ақпан	наурыз	қаңтар	ақпан	наурыз	
4	1	№1 тауар	6400	20	30	80	128000	192000	512000	832000
5	2	№2 тауар	6400	30	20	70	192000	128000	448000	768000
6	3	№3 тауар	6400	40	50	90	256000	320000	576000	1152000
7	4	№4 тауар	6400	60	40	30	384000	256000	192000	832000
8	5	№5 тауар	6400	80	50	20	512000	320000	128000	960000
9	6	№6 тауар	6400	40	50	70	256000	320000	448000	1024000

3-сүрет. Формуланы кестеде қолдану

Әр тауардың өз құны жазылуы үшін, С4 ұяшыққа қатысты ‘=[1.xlsx]Лист1!\$D\$4’ формуладағы қатар нөмірінің алдындағы \$ белгісін алып тастаймыз: ‘=[1.xlsx]Лист1!\$D4’ (4-сүрет).

	A	B	C	D	E	F	G	H	I	J
1	Тоқсандағы жалпы сатып алғынған тауар құнының есептеу									
2	№	Тауардың аты	Бағасы, т	Сатылған тауар саны			Сатылған тауар сомасы			Жалпы сомасы
3				қаңтар	ақпан	наурыз	қаңтар	ақпан	наурыз	
4	1	№1 тауар	6400	20	30	80	128000	192000	512000	832000
5	2	№2 тауар	12800	30	20	70	384000	256000	896000	1536000
6	3	№3 тауар	32000	40	50	90	1280000	1600000	2880000	5760000
7	4	№4 тауар	80000	60	40	30	4800000	3200000	2400000	10400000
8	5	№5 тауар	96000	80	50	20	7680000	4800000	1920000	14400000
9	6	№6 тауар	128000	40	50	70	5120000	6400000	8960000	20480000

4-сүрет. Формуланы кестеде қолдану нәтижесі

Міне, абсолютті және салыстырмалы сілтемелерді программа-да қолдану жолдарымен таныстырыңыз. Енді төмендегі есептерді шешу жолдарына абсолютті және салыстырмалы сілтемелер жайлыш алған білімінді қолданып көр.



Практикалық жұмыс

Жеңіл автокөліктің қозғалысы

Бірқалыпты 72 км/сағ жылдамдықпен жүріп келе жатқан жеңіл автокөліктің (*5-сурет*) әрбір 2 секундтағы жүрген жолын табу кестесін құр.

Тапсырма

- Жеңіл автокөліктің жылдамдығын м/с өлшемінде есепте.
- 20 секундта жүрген жолын анықта.
- 72 км/сағат жылдамдықпен 1 км жолды қанша уақытта жүріп өтеді?
- Электрондық кестені төмендегідей құр. Есептеуге еki баған ал. Біріншісінде қозғалыс уақыты, екіншісінде осы уақыт ара-лығында жеңіл автокөліктің қанша жол жүргені сипатталады.

№	A	B
1	Автокөліктің жылдамдығы	
2	Жылдамдық (км/сағат)	72
3	Жылдамдық (м/с)	20
4	Уақыт аралиғы	2

Сен 7-сыныпта физика сабакында км/сағатты м/с-қа айналдыруды үйренгенсің. В3 үяшығына ‘=B2*1000/3600’ формуласын жаз. Абсолютті сілтемені қолданып, В6 үяшығына ‘=\$B\$3*A6’ жаз. Кестеде 20 секунд ішінде автокөліктің жалпы жүрген жолы мен әр 2 секунд сайын қашықтықтың өзгеруі сипатталған.



5-сурет. Есепке иллюстрация

A	B
1	Автокөліктің жылдамдығы
2	Жылдамдық (км/сағат)
3	Жылдамдық (м/с)
4	Уақыт аралиғы (с)
5	Уақыт (с)
6	Жол (м)
7	0
8	2
9	4
10	6
11	8
12	10
13	12
14	14
15	16
16	18
	20
	400

6-сурет. Жеңіл автокөліктің жүрген уақыты мен жылдамдығы

Тапсырманың жауаптары

- Жеңіл автокөліктің жылдамдығы 20 м/с-қа тең.
- 20 секундта жүрген жолы – 400 метр (*б-сурет*).
- 72 км/сағат жылдамдықпен 1 км жолды 50 секундта жүріп өтеді.

Бұл тапсырманың жауабын табуда 1000 м-ге сәйкес келген уақытты таңдаймыз.



Ойлануға берілген сұрақтар

- Абсолютті және салыстырмалы сілтемелерді қолданудың тағы қандай жолдарын білесің?
- Абсолютті және салыстырмалы сілтемелер арқылы екі файлды қалай бір-бірімен байланыстыруға болады?
- Екі файлды сілтемелер арқылы байланыстыру алгоритмі қандай қадамдардан тұрады?



Тапсырма

1-тапсырма. Төменде электрондық кесте бөлігі формула көрсету режимінде берілген.

Егер С1 ұяшықтағы формуланы көрсету режимінен шыққаннан соң С3 ұяшыққа көшіріп қойсақ, С1 мен С3 ұяшықтарында қандай сандар пайда болады?

№	A	B	C
1	10	15	=B1 + \$A\$2
2	20	20	
3	30	25	
4	40	30	

2-тапсырма. С2 ұяшығындағы формуланы С3 ұяшығына көшіріп қойсақ, онда қандай формула пайда болады?

№	A	B	C
1	30		
2	12	4	=\$A\$1*A2 +B2
3	23	5	
4	43	2	

3-тапсырма. Рамазан 5 бөлмелі үлкен үйде тұрады. Рамазан үйдегі бөлмелердің ені мен ұзындығын өлшеп, кестелік процессорға жа-

зып отыр. Ол төмендегідей дерек алды. Рамазанға үйдегі бөлмелердің ауданы мен периметрін есептеуге көмектес.

Nº	A	B	C	D	E	F
1	№	Бөлмелер	Ұзындығы (м)	Ені (м)	Ауданы (м^2)	Периметрі (м)
2	1	Қонақ қүттегін бөлме	5	6		
3	2	Балалар бөлмесі	4	3		
4	3	Асүй	5	3		
5	4	Жатын бөлме	4	4		
6	5	Ойын бөлмесі	3	3		

4-тапсырма. Самат 8-сыныпта оқиды. Ол жазғы демалыста әкесіне шаруашылықта мал бордақылауға көмектесті. Саматтың ең-бекқорлығына әкесі риза болды. Самат бордақылаған малдың бастапқы салмағы әр күн сайын 1%-ға артып отырды. Төменде Саматтың мал бордақылаумен айналысқан алғашқы 10 күні берілген.

Есептің формуласын құрып, есептеулер жүргіз (*7-cypet*).

1. 10 күннен кейін Самат бордақылаған малдың салмағы қанша килограммға жетті?
2. Самат әкесіне көмектескен аралықта ол бордақылаған мал неше килограмм салмақ қосты?

	A	B	C	D	E	F	G	H	I	J	K	L
1												
2	Күнделікті малдың семіру пайызы бастапқы салмақтан =							1% =	0,01			
3												
4	Күн		1	2	3	4	5	6	7	8	9	10
5	Бастапқы салмағы (кг)	280	282,8									

7-cypet



Үй тапсырмасы

5-тапсырма. 1-файлда тауар бағасының әр айдағы өзгеруі, ал 2-файлда сол тауардың әрбір айдағы сатылған көлемі берілген. 2-файлдағы әр айда сатылған тауардың сомасын анықта.

1-файлдағы дерек

Nº	A	B
1	Ай аттары	Тауардың бағасы (тг)
2	Сәуір	100000
3	Мамыр	105000
4	Маусым	103000

2-файлдағы дерек

Nº	A	B	C
1	Ай аттары	Сатылған тауар саны	Сатылған тауар сомасы
2	Сәуір	80	=
3	Мамыр	57	
4	Маусым	68	

3.3

Кірістірілген функциялар



Электронды кестенің көмегімен есептерді шешуде кірістірілген функцияларды қалай пайдалануға болады?



Ойлан

- «Функция» дегеніміз не?
- Функцияның математика мен адам өміріндегі маңызы қандай?
- «Программаға кірістірілген функциялар» деген не?



Жаңа білім

Функциялар. «Функция» үғымына анықтама

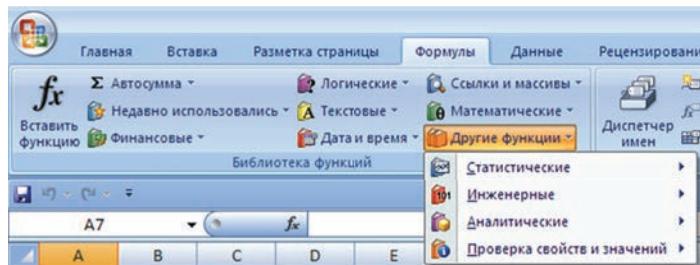
Microsoft Excel программасының функциясы ретінде белгілі бір алгоритм бойынша немесе формулалар бойынша жүргізілетін есептеу операциялары қарастырылады. Әрбір функцияның өзіне тән атауы болады. Excel программасында функциялардың аттары пернелік тақтадан теріліп жазылады, **Формулы** (Формулалар)=>**Библиотека функций** (Функция кітапханасы) деп аталатын команда немесе сайдандар панеліндегі **f_x Вставить функцию** (Функцияны кірістіру) батырмасының көмегімен енгізіледі. **Функция** шебері функцияларға сәйкес формулалардың дайын шаблондарын береді, қолданушы небәрі функцияның аргументтерін ғана жазады.

Excel программасында күрделі мен қарапайым есептеулерге арналған 200-ден аса кірістірілген функция қолдану мүмкіндігі қарастырылған. Сондай-ақ бұл қосымшада қолданушы Excel-дің программалау мүмкіндіктерін (VBA) пайдаланып, өзіне қажет функцияларды да құрып ала алады.

Функция шеберін пайдаланып, функцияларды іздеуді жылдамдату мақсатында олар қаржылық, логикалық, мәтіндік, мерзім мен уақыт, математикалық сілтемелер мен статистикалық деректер, инженерлік, аналитикалық, қасиет және мәнін тексеру санаттары бойынша бөлінген (*1-сурет*).



кірістірілген функциялар – встроенные функции – built-in functions
функциялар – функции – functions



1-сурет. Функция түрлөрі

Функциялардың аргументтері ретінде сандар, санды өрнектер немесе сәйкесінше өрнектер жазылған ұяшық адрестері мен диапазон адрестері пайдаланылады.

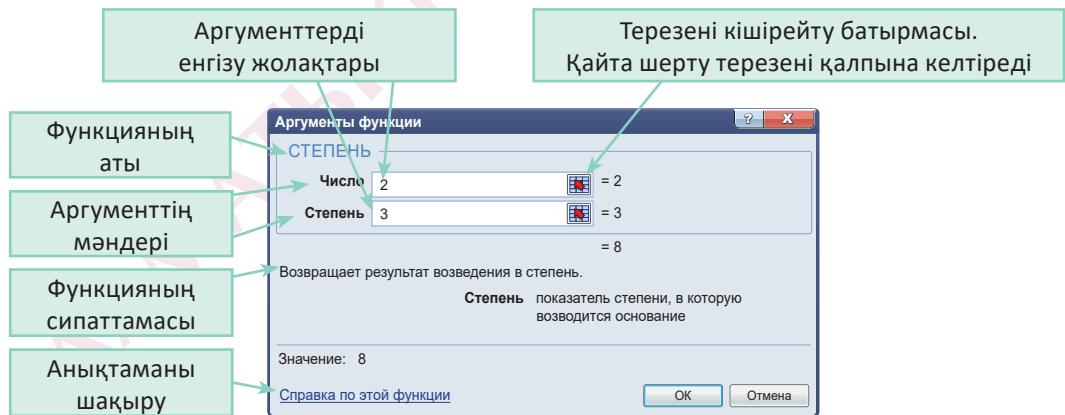
Мысалы:

= СУММ(A5:A9) – мұндағы СУММ функциясы A5, A6, A7, A8, A9 ұяшықтардағы сандар қосындысын (сомасын) есептейді;

= СРЗНАЧ(G4:G6) – мұндағы СРЗНАЧ функциясы G4, G5, G6 ұяшықтардағы сандардың орташа мәнін табады.

Күрделі функциялар әдеттегідей бірінің ішіне бірі жай жақшалар арқылы жазылады: = ОКРУГЛ(СРЗНАЧ(Н4: Н8); 2).

Функцияның аргументтерін функция шеберін пайдаланып енгізу Функция шеберін пайдаланғанда, егер оның аргументі бар болса, онда келесі 2-суреттегідей терезе пайда болады. Бұл терезе функция аргументтерін енгізуге арналған.



2-сурет. Функция шеберінің терезесі

Функция аргументтерін енгізуде келесі тәсілдер қолданылады.

1. Қажет адрестер (ұяшық немесе диапазон адрестері) пернешақтадан теріледі.
2. Қажет ұяшықтар мен диапазондарды жұмысшы беттегі кестеде белгілеу арқылы жазуга болады.

Енді осы функциялардың ішінен ең көп қолданылатын кейбір функцияларға тоқталып өтейік. Бұл функцияларды есеп шешуде қолданамыз.



Математикалық функциялар

ABS (число) – санның модулі;

SIN (число), COS (число), TAN (число) – берілген санның сәйкес тригонометриялық функциялары;

ГРАДУСЫ (угол) – радианды градусқа айналдырады;

ЗНАК (число) – санның таңбасын анықтайды: егер 1 оң сан болса; егер 0 саны 0-ге тең болса; егер 1 теріс сан болса;

КОРЕНЬ (число) – квадрат түбірдің оң мәнін анықтайтын функция;

СУММ (число 1; число 2; ...) – аргументтер тізіміне кіретін барлық санның қосындысын анықтайды;

СТЕПЕНЬ (число; степень) – негізі кез келген нақты болатын санды дәрежелейді.

Статистикалық функциялар

МАКС (аргумент 1; аргумент 2;...) – аргументтердің максимумын анықтайды;

МИН (аргумент 1; аргумент 2;...) – аргументтердің минимумын анықтайды;

СРЗНАЧ (аргумент 1; аргумент 2;...) – аргументтердің орташа мәнін есептейді;

СЧЕТ ЕСЛИ (диапазон; условие) – берілген шартқа сәйкес келетін диапазон аралығындағы аргументтер санын есептейді.

Логикалық функциялар

Логикалық функциялар қандай да бір шарттың орындалу нәтижесіне байланысты қызмет етеді. Бұл шарттарда салыстыру операциялары қолданылады: $=$, $>$, $<$, \neq (тең емес), \geq (үлкен немесе тең), \leq (кіші немесе тең).

ЕСЛИ (ЕГЕР) (логикалық _ұфым; 1-мән; 2-мән) – 1-мән шешімі болса, онда логикалық ұфым ақиқат және керісінше жағдайда 2-мән ақиқат болады. Мысалы: A1 ұяшығына 30 000 саны енгізілген, ал B1 ұяшығына =ЕСЛИ (A1<20000; 12; 15) формуласы енгізілген. Шарттың орындалмауына байланысты нәтижесі 15-ке тең болады.

И (ЖӘНЕ), ИЛИ (НЕМЕСЕ) функциялары күрделі шарттарды құруға арналған.

И (ЖӘНЕ) (логикалық_1 ұфым; логикалық_2 ұфым;...) – егер аргументтердің барлығы АҚИҚАТ мән қабылдаса, онда мән **АҚИҚАТ (ИСТИНА)**, кері жағдайда **ЖАЛҒАН (ЛОЖЬ)** болады.

ИЛИ (НЕМЕСЕ) (логикалық_1 үфім; логикалық_2 үфім;...) – егер аргументтердің жоқ дегенде біреуі АҚИҚАТ мән қабылдаса, онда мән **ИСТИНА (АҚИҚАТ)**, кері жағдайда **ЛОЖЬ (ЖАЛҒАН)** болады.

Мысалы: =ЕСЛИ (И(A1>=20000;A1<40000);15;18) – егер A1 мәні 20000 мен 40000 аралығындағы сандарды қабылдаса, онда мәні 15-ке тең, кері жағдайда 18-ге тең.

Кірістірілген ЕСЛИ (ЕГЕР) функциясының көмегімен күрделі шарттарды құруға болады. Мысалы:

=ЕСЛИ (A1<20000; 12; ЕСЛИ (A1<40000; 15; 18)) – егер A1 мәні 20000-нан кіші болса, есептелген мән 12-ге тең, егер A1 мәні 40000-нан кіші болса, онда мәні 15, кері жағдайда 18 болады.

Егер үяшықтағы формуланы программа **дұрыс анықтай алмаса**, үяшықта төмендегі қателердің бірі бейнеленеді:

Қате бейнесі	Себебі
# Дел/0!	Нөлге бөлу берілген
# Знач!	Қате аргумент немесе оператор енгізілген
# Имя?	Мүмкін емес мән көрсетілген
# Н/Д	Мән берілмеген
# Пусто!	Қызылспектің үяшықтар аймағы көрсетілген
#Сылка!	Қате сілтеме берілген
# Число!	Мәнді алу барысындағы қателік



Практикалық жұмыс

Санды зерттеу

Бестаңбалы сан берілген. Осы санды төмендегі қадамдарды орындаі отырып, функциялар көмегімен зертте. Мысалы: 54289.

- Санның квадратын табу.
- Саннан түбір алу.
- Санның цифрларының қосындысын табу.
- Сан цифрларының арасынан ең кішісін анықтау.
- Санның цифрларының арифметикалық орташасын табу.

Практикалық тапсырманы электрондық кестеде 3-суреттегідей толтыру керек.

МИН		<input checked="" type="radio"/> X ✓ f _x	=СРЗНАЧ(5;4;2;8;9)
A	B	C	D
1			
2 №	54289 санын зерттеу нәтижесі		
3 1	Санның квадратын табу	2947295521	
4 2	Саннан түбір алу	233	
5 3	Санның цифрларының қосындысын табу	28	
6 4	Сан цифрларының арасынан ең кішісін анықтау	2	
7 5	Санның цифрларының арифметикалық орташасын табу	=СРЗНАЧ(5;4;2;8;9)	

3-сурет. Тапсырманы кестеде орындау

Практикалық жұмысты орындауда функция шебері арқылы төмендегі функцияларды пайдаланамыз. Ұяшықтарға осы функциялардың формуласын жазуға болады:

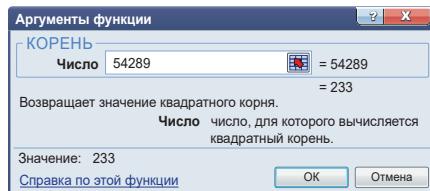
C3 ұяшығы: = СТЕПЕНЬ (54289;2)

C4 ұяшығы: =КОРЕНЬ (54289) (4-сүрет);

C5 ұяшығы: =СУММ (5; 4; 2; 8; 9);

C6 ұяшығы: =МИН (5; 4; 2; 8; 9);

C7 ұяшығы: =СРЗНАЧ (5; 4; 2; 8; 9).



4-сүрет. Квадрат түбір терезесі



Ойлануға берілген сұрақтар

- Функцияның математикалық анықтамасы қалай айттылады?
- Excel программындағы функция қандай қызмет атқарады?
- Excel-де функциялардың қандай санаттары бар?
- Функция аргументтерін енгізуде қандай тәсілдер қолданылады?
- Функцияның жұмыс терезесінің құрылышы қандай?
- Қандай математикалық функциялар бар?
- Логикалық функциялардың қызметі қандай?



Үй тапсырмасы

Қанат 8-сыныпта оқиды. Ол оқитын мектептегі 8-сыныптардың саны 8. Сыныптардағы I тоқсан бойынша білім сапасының қорытындысы шығарылды. Қанат 8-сыныптар бойынша I тоқсандағы білім сапасына қарап, төмендегі жағдайларды анықтауы керек. Электронды кестемен жұмыс істеуді енді ғана үйрене бастаған Қанат сендерден көмек сұрайды.

Электронды кесте бөлігі

№	Тапсырманың мазмұны	Нәтиже жазылатын ұяшық
1	Білім сапасы ең жоғары сыныпты анықтау	C1
2	Білім сапасы ең төмен сыныпты анықтау	C2
3	Сыныптар бойынша орташа білім сапасын анықтау	B9
4	Білім сапасы ең төмен сыныптың білімін орташа деңгейге көтеру үшін қанша пайызға көтеру керегін анықтау	C3
5	Білім сапасы ең жоғары және ең төмен сыныптардың көрсеткіштерінің айырмашылығы	C4

№	A	B	C	D
1	8 «А»	74%		
2	8 «Ә»	72%		
3	8 «Б»	51%		
4	8 «В»	73%		
5	8 «Г»	81%		
6	8 «Д»	37%		
7	8 «Е»	64%		
8	8 «Ж»	68%		

3.4

Кірістірілген функцияларды практикалық түрғыдан қолдану



Функцияларды практикалық түрғыдан қолдану қандай мүмкіндіктерге жол ашады?



Ойлан

- Кірістірілген функциялардың көмегімен құнделікті кездесетін қандай есептерді шешуге болады?
- Кірістірілген функцияларды алгебра пәнінің өзіне таныс қай тарауларына тиімді қолдануға болады деп ойлайсың?



Практикалық жұмыс

Кірістірілген функцияларды практикалық түрғыдан қолдану барысында есептерді кестелік процессор көмегімен шешу жолдарын үйренесің. Сонымен бірге функцияның графигін салу жолдарын, құнделікті өмірмен байланысты есептерді (мысалы, электр энергиясын есептеу) логикалық функциялар көмегімен шешу жолдарын меңгересің. Практикалық жұмыстарды орындаудың толық кезеңі берілген. Берілген ұлгіге сай тапсырмаларды мұқият оқып, компьютерде басқа да тапсырмаларды осы тәрізді құрастырып, өз бетіңше орындайсың.

1-практикалық жұмыс

Мақтау қағазы

Оқу жылышын соңында барлық пәннен үздік шыққан оқушылар сыныпты мақтау қағазымен аяқтайды. Рамазан 8-сыныпта оқиды. Оның оқу жылышы соңындағы жылдық бағаларының қорытындысы электронды кесте түрінде берілген (*1-сурет*).

Тапсырма. Рамазан оқу жылышын соңында үздіктер қатарына кіріп, 8-сыныпты мақтау қағазымен аяқтай ала ма? Анықта.

Тапсырманы орындау идеясы. «Талдау» бағанына «5» деген баға алған пәндер қатарына 1 немесе 0 қоямыз. Егер оқушының барлық пәннен бағасы «5» болса, онда «Талдау» бағанындағы сандар қосындысы пәндердің жалпы санына тең болуы керек. Біздің жағдайда ол 14-ке тең.



сілтемелер және жиындар – ссылки и массивы – references and arrays
математикалық өрнек – математическое выражение – mathematically expression

A	B	C	D
1	Оқушының жылдық қорытынды бағасы		
3	Пәні	Бағасы	Талдау
4	Казақ тілі	5	1
5	Казақ әдебиеті	5	1
6	Орыс тілі	4	0
7	Орыс әдебиеті	5	1
8	Ағылшын тілі	4	0
9	Қазақстан тарихы	5	1
10	Алгебра	4	0
11	Геометрия	5	1
12	Информатика	5	1
13	Физика	5	1
14	Химия	3	0
15	Биология	5	1
16	География	4	0
17	Денешінің тұруы	5	1
18	Орташа бағасы	4,6	9
19	Мақтатай қағазын ала ма?		Жоқ

1-сүрет. Бағалау

Тапсырманы орындау қадамдары

1. Талдау бағанындағы D4

ұяшығына логикалық функция қолданамыз: =ЕСЛИ(C4=5; 1; 0)

2. D4 ұяшығындағы формуланы D17 ұяшығына дейінгі ұяшықтағы

сандардың қосындысын табамыз: =СУММ(D4:D17)

4. D19 ұяшығына логикалық функция қолданамыз:

=ЕСЛИ(D18=14; «Иә»; «Жоқ»)

5. C18 ұяшығына C4 пен C17

мәндерінің арифметикалық орташасын табу формуласын қоямыз: =СРЗНАЧ(C4:C17)

2-практикалық жұмыс

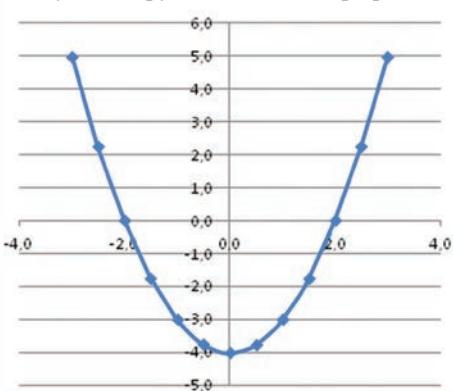
Функция графигі

Тапсырма. $y=x^2-4$ функциясының графигін сал. Функция графигін салу үшін, Excel программасында аргументі x -тің мәніне ординатасы y -тің сәйкес мәндерінен кесте жасаймыз (1-кесте).

1-кесте. y -тің x -ке тәуелділік кестесі

	A	B	C	D	E	F	G	H	I	J	K	L	M	N
1	x	-3,0	-2,5	-2,0	-1,5	-1,0	-0,5	0,0	0,5	1,0	1,5	2,0	2,5	3,0
2	y	5,0	2,3	0,0	-1,8	-3,0	-3,8	-4,0	-3,8	-3,0	-1,8	0,0	2,3	5,0

$y=x^2-4$ функциясының графигі



2-сүрет. График

Тапсырманы орындау қадамдары

1. Кестені жасауда x -тің мәнін, мысалы, -3-тен бастап 0,5 қадаммен алдық.

2. B2 ұяшығына =B1*B1-4 формуласын енгізіп, N2 ұяшығына дейінгі қөшіріп қоямыз.

3. y -тің x -ке тәуелділік кестесін қөшіріп, Вставка (Кірістіру) => Точечная (Нұктелік) => Точечная гладкими кривыми (Тегіс нұктелік иір сызық) командаларын орындасақ, 2-сүреттегі график пайда болады.

Егер кестедегі мәндер өзгерсе, онда диаграмма да өзгеріске ұшырайды.

3-практикалық жұмыс



Жарықты есептеу

Тұтынушыларға 1кВт/сағат электр энергиясының құны мен есептегіш құралдың көрсеткіші берілген. Сонымен бірге тұтынушылардың өткен ай мен осы айда тұтынған электр энергиясының шығыны көрсетілген.

Тапсырма

1. Тұтынушылардың өткен ай мен осы айда тұтынған электр энергиясының жалпы көлемін анықтау.
2. Тұтынушылардың осы айда тұтынған электр энергиясының орташа көрсеткішін табу.
3. Электр энергиясын ең көп тұтынған абонентті анықтау.
4. Электр энергиясын ең аз тұтынған абонентті анықтау.
5. Тұтынушылардың электр энергиясын тұтыну шығынының диаграммасын жасау.

Бұл практикалық тапсырманы орындау барысында сен Excel программасындағы кестелерді жасау мен пішімдеу, ұяшықтарға формула қою, абсолютті сілтеме, функция және диаграмма тақырыптарында алған білімінді бір тапсырма шеңберінде қолданып үйренесің.

Тапсырманы орындау қадамдары

1. З-суреттегі үлгі кестені жасап, ұяшықтардағы деректерді форматтаймыз.
2. Ұяшыққа енгізілетін деректерді қоямыз.

A	B	C	D	E
1	Жарықты есептеу			
2	1кВт/сағат құны 15 теңге			
3	Үй №	Өткен айда есептегіштің көрсеткіші (кВт/сағ)	Ағымдағы айда есептегіштің көрсеткіші (кВт/сағ)	Шығын (кВт/сағ)
4	1-үй	2243	2347	104
5	2-үй	4256	4401	145
6	3-үй	356	499	143
7	4-үй	7411	7789	378
8	5-үй	8685	8888	203
9	6-үй	25	231	206
10	7-үй	485	567	82
11	Жалпы шығын	23461	24722	1261
12	Орташа шығын	3351,6	3531,7	180,1
13	Ең көп тұтыну			378
14	Ең аз тұтыну			82

З-сурет. Электр энергиясын есептеу деректері

- 1-тапсырма бойынша B11, C11, D11, E11 ұяшықтарына қосынды алу функциясын пайдаланып, қосындының мәнін аламыз. Мысалы, B11 ұяшығына = СУММ(B4:B10) формуласын қоямыз.
- Тұтынушылардың ағымдағы айда пайдаланған электр энергиясының орташа көрсеткішін табу үшін, D12 ұяшығына = СРЗНАЧ (D4:D10) формуласын жазамыз.
- Электр энергиясын ең көп пайдаланған абонентті табу үшін, = МАКС(D4:D10) формуласын D13 ұяшығына, ең аз пайдаланған абонентті табу үшін, = МИН(D4:D10) формуласын D14 ұяшығына жазамыз.
- Тұтынушылардың электр энергиясын пайдалану шығынының диаграммасын жасау үшін, D4:D10 аралығындағы ұяшықтарды көшіріп, диаграммаға кірістіру арқылы жаңа диаграмма аламыз (4-сурет).



4-сурет. Диаграмма



Үй тапсырмасы

- Оқушылардың аты-жөні, жасы мен бойы көрсетілген. Волейбол секциясына жасы 14-тен аспаған, бойы 165 см-ден жоғары қанша оқушы қатыса алады?
- Сауда қоймасындағы өнімдердің сақталу мерзімінің өзгеруіне байланысты бағасы да арзандап отырады. Егер өнім қоймада 10 айдан артық сақталса, онда оның бағасы 100 пайызға, егер 6 айдан асып, 10 айға жетпесе, 50 пайызға арзандайды. Сауда қоймасындағы тауарлардың жинақ кестесін жаса. Кестеде тауардың аты, сақталу мерзімі, арзандағанға дейінгі бағасы, арзандағаннан кейінгі бағасы болуы керек.
- $y = -x^2 + 3$ функциясының графигін сал.

Ескерту: Кестеде көрсететін деректердің күнделікті өмірдеңі деректерге сүйене отырып ұсынуымыз керек. Мысалы, оқушының бойы 300 см десек, бұл шындыққа жанаспайды.

3.5

Қолжетімді ақпараттың негізінде деректерді талдау



Excel программасының көмегімен қолда бар ақпарат негізінде қалай болжам жасауға болады?



Ойлан

- «Болжау» дегеніміз не?
- Күнделікті өмірде нендей нәрселерді болжаймыз?
- Фылым мен техникада болжау жасала ма?



Жаңа білім

Статистикада хронологиялық тәртіpte жүйелі турде орналасқан әрі құбылыстың дамуын көрсететін қатарлар **динамикалық қатарлар** деп аталады. Динамикалық қатарда өрбір уақыт кесіндісі үшін екі көрсеткіш – уақыт көрсеткіші t мен қатардың деңгейі y келтіріледі. Динамикалық қатар белгілі бір көрсеткіштерді өлшеу нәтижесінде пайда болады. Бұл көрсеткіштер техникалық жүйенің, табиғаттың, экономика мен өлеуметтік жүйелердің көрсеткіштері болуы мүмкін. Динамикалық қатардың көмегімен объект температурасының өзгерісі, желінің өткізу қабілеті, тауардың белгілі бір уақыт аралығындағы сатылуын талдауды мысал ретінде айтуға болады. Динамикалық қатардың көмегімен белгілі бір көрсеткіштердің негізгі даму тренді немесе бағыты анықталады.

Динамикалық қатарды талдау дегеніміз – динамикалық қатар құрылымын анықтауға әрі болжам жасауға арналған математикалық-статистикалық әдістер жиынтығы.

Болжам «болжау», «аян беру» деген мағынаны білдіреді. **Болжау** – болашақ оқиғаның немесе құбылыстың ғылыми моделі. **Болжамдау** – болжауды жобалау кезеңі.

Болжау әдісіне:

- статистикалық әдіс;
- эксперттік бағалау (дельфи әдісі);
- модельдеу жатады.



деректерді талдау – анализ данных – data analysis
болжам – прогноз, прогнозирование – forecast, prediction

Excel программасында диаграммалар көмегімен деректердің өзгерісін болжайтын **сызықтық трендті** пайдалануға болады. Тренд сызықтары деректердегі трендтерді графикалық түрде көрсетуге әрі болжам мәселелерін талдауға арналған. Мұндай талдау **регрессиялық талдау** деп те аталады. Регрессиялық талдауды пайдалану арқылы болашақ мәндерді болжау үшін, диаграммадағы тренд сызығын шынайы деректен тыс созуға болады.

Excel-дегі тренд сызықтарының түрлері

Excel программасындағы диаграммаға тренд сызығын қосу қажет болғанда, алты түрлі трендті немесе регрессиялық түрлердің кез келгенін таңдауға болады.

Оларға сызықтық, логарифмдік, полиномдық, дәрежелік, жылжымалы орташа мән және экспоненциалдық түрлері жатады. Сызықтық трендпен танысайық.

Сызықтық тренд тұрақты жылдамдықпен өсетін немесе кемитін модельдерді сипаттауға қолайлы болжаудың қарапайым түріне жатады. Трендтің бұл түрі берілген деректерді сипаттайтын нүктелердің өзара орналасуы түзу сызыққа жақын болған жағдайды сипаттау үшін өте қолайлыш. Сызықтық тренд санды деректерді талдау үшін пайдаланылады. Мысалы, екі шаманың өзара байланыс заңдылығын бағалау керек болған жағдайда қажет. Сызықтық трендті құруда екі шаманың арасындағы өзара байланыстың сызықтық функция коэффициенті ең кіші квадраттар әдісімен анықталады.

Сызықтық функция коэффициентінің мәндерін табуға R квадрат коэффициенті қолданылады.

R-квадрат мәні

R-квадратты сенімділік коэффициенті деп атайды. Оның көмегімен болжаудың дәлдігін анықтайтын. R-квадрат мәні 1-ге тең немесе 1-ге жақын болғанда, тренд сызығы түзу сызыққа дәл келеді, яғни болжау дұрыс. Тренд сызығын деректерге сәйкестендіргендеге, Excel программының автоматты түрде оның R-квадрат мәнін есептейді. Қажет болған жағдайда бұл мәнді диаграммада көрсетуге болады.

Excel программасында тренд сызығын қалай қосамыз? Бұл сұраққа жауап беру үшін төмендегі мысалды қарастырайық.

Қанат әлеуметтік желідегі блогерлердің парақшаларын қарау барысында онымен танысушылардың санының артуы парақшада жарияланған мақалалар (пост, репост) санына тәуелді екендігін байқады. Қанаттың болжауынша, парақшаларда

жарияланған пост саны мен танысушылардың санының арасында байланыс бар.

Жарияланған пост саны 100-ге жеткенде, танысушылар саны 228 мыңға жеткен (1-кесте). Енді Қанатты төмендегі сұрақ қызықтырды: «Жарияланған пост саны 120 болғанда, парақшамен қанша адам танысады? Ал 150 болса ше?». Қанат осы екі шаманың өзара байланыс формуласын табу арқылы алдағы уақытта парапада жарияланған пост санына қарай парақшалармен танысушылардың санын қалай болжай алады?

1-кесте. Әлеуметтік желілерде жарияланған мақалалар саны мен танысушылар саны

№	Парақшада жарияланған мақалалар саны	Парақшамен танысушылар саны (мың)
1	30	67
2	42	88
3	57	92
4	69	120
5	75	145
6	87	184
7	95	199
8	100	228
9	120	?
10	150	?

Екі шама арасындағы байланысты анықтау теңдеуін табу үшін белгілеу енгізейік. Мақала санын x , кірушілер санын y деп алайық. 1-кестеге қарап, бұл шамалар арасындағы байланыс сызықты деп қарайық. Онда алгебра курсынан белгілі сызықтық теңдеу формуласын жазамыз: $y = kx + b$.

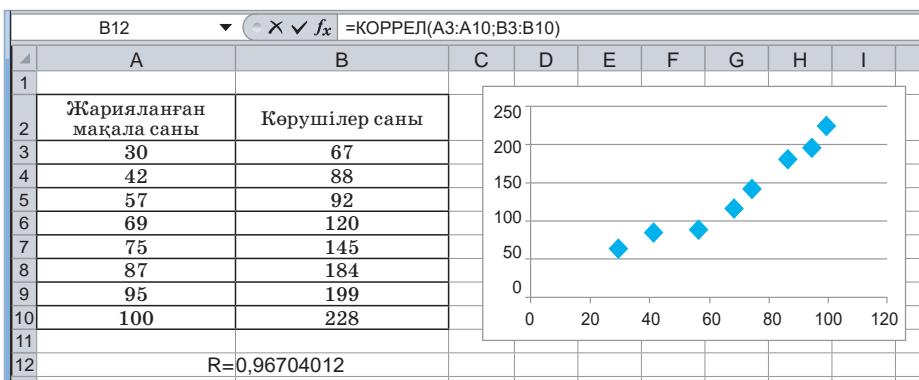
Егер осы теңдеудің k және b коэффициенттерін есептей алсақ, онда x -тің орнына кез келген мән қою арқылы өзімізді қызықтырған сұраққа жауап табамыз.

Кестедегі деректерге қарап, сызықтық графигін жасайық.

Ол үшін кестенің элементтерін ерекшелеп, төмендегі әрекеттерді орындау керек. **Вставка (Кірістіру) => Диаграммы (Диаграммалар) => Точечная (Нұктелік) => Точечная с маркерами (Маркерлі нұктелік)** командаларын орындаپ, **1-суреттегі** диаграмманы аламыз.

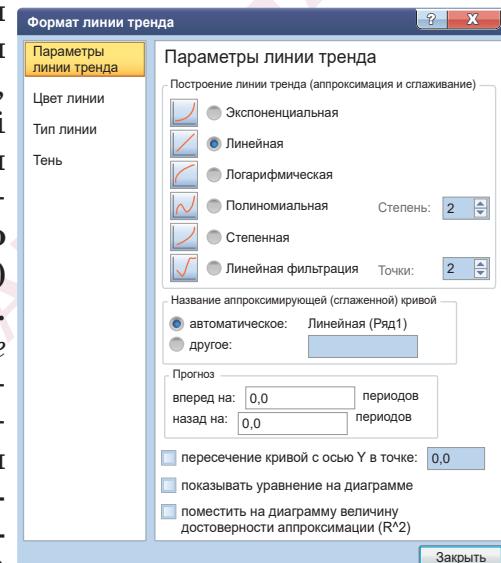
Диаграммада x осі бойынша пост саны, y осі бойынша бетті көру саны (мың есебімен) орналасқан. Диаграммада көріп тұрғандай нұктелер бір тұзу бойымен орналаспай, шашылып кеткен. Нұктелердің бәріне мүмкіндігінше жақын орналасқан тұзу сызық жүргізе аламыз ба? Сұраққа жауап беру үшін бұл нұктелердің корреляциясын тексеру керек. Бір шаманың өзгеруінен екінші шаманың орта мәні өзгеретін статистикалық тәуелділік **корреляциялық тәуелділік** деп аталады. Excel-де **КОРРЕЛ** (корреляция) функциясы бар (**1-сурет**). Корреляция функциясының мәні **R**-мен белгіленеді. **Корреляция** көне лат. *correlatio* – «арақатыс», «өзара байланыс» деген мағына береді. Корреляция функциясының мәні **R > 0** оң мәнге өсетін байланыс болса, **R < 0** болғанда теріс кемуші байланыс болады. Корреляция функциясының мәні 1-ге жақын-

даған сайын байланыс сыйығы түзуге жақын болады. Біздің жағдайда $R = 0,96704012$ -ге тең, яғни сыйықтық байланыс айқын байқалады.

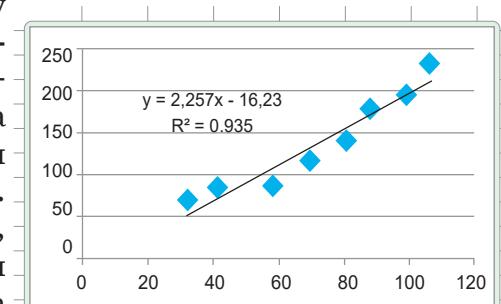


1-сүрет. Корреляция

Диаграммаға тренд сыйығын жүргізу үшін, нұктелер арасынан бір нүктені таңдаймыз. Мысалы, диаграммада 5-ші тұрған нүктені алайық. Оны тінтуірдің оң жағымен басып тұрып, пайда болған контекстік мәзірден **Добавить линию тренда...** (Тренд сыйығын қосу...) командасын орындаімыз (2-сүрет). 6 түрлі трендтің ішінен **2-сүретте** көріп тұрғандай **Линейная** (Сыйықтық) трендін таңдаймыз. Сыйықтық трендтің байланыс тендеуін диаграмма бетіне шығару үшін, **Показывать уравнение на диаграмме** (Тендеуді диаграммада көрсету) және шаманың шынайылық аппроксимациясы R^2 -ты диаграммаға қою үшін, **Поместить на диаграмму величину достоверности аппрокси-мации R^2** (R^2 сенімділік аппрокси-мациясының шамасын диаграммаға енгіз) командаларының жанындағы төртбұрыштарға белгілер соғамыз. **Закрыть (Жабу)** батырмасын бассақ, 3-сүреттегі байланыс тендеуі мен R^2 мәні бетіне орналасқан диаграмма пайда болады.



2-сүрет. Тренд сыйығы



3-сүрет. Байланыс тендеуі

Енді Қанаттың сұрағының жауабын табайық. Ол үшін мына төмөндегі әдістерді пайдалануға болады.

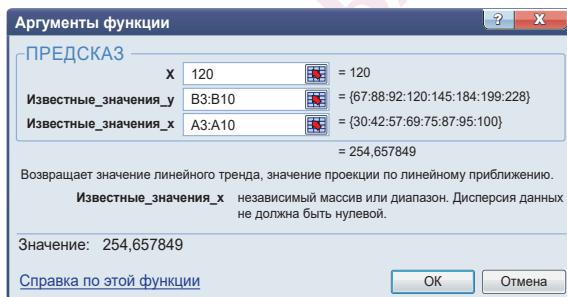
1-әдіс. Диаграммадағы $y = 2,257x - 16,23$ сызықтық теңдеуге қарап, оның қаншалықты өсетінін болжауымыз керек. Ол үшін $x = 120$ мәнін теңдеуге қоямыз. Есептеу нәтижесінде бетті көрушілер саны орта есеппен 254 мың екені анықталды.

Мақала саны 150 болғанда, 322 мың көруші болады деп болжаймыз. $y = 2,257 \cdot 120 - 16,23, y = 254,61$.

2-әдіс. Болжау үшін ПРЕДСКАЗ функциясын қолданамыз. Бұл функция да статистикалық функциялар тобына жатады. ПРЕДСКАЗ функциясының жұмысын *4-суреттегі* терезеден байқауға болады:

= ПРЕДСКАЗ (x ; y -тің белгілі мәндер аралығы; x -тің белгілі мәндер аралығы).

x орнына 1-кестеден болжам жасалатын 120 мәні, y -тің белгілі мәндер аралығына кірушілердің белгілі саны ескеріліп, x -тің белгілі мәндері аралығында болжау жасалады. $x = 120$ -дан жоғары орналасқан мәндер аралығын қоямыз. Болжау нәтижесі алдыңғы бірінші әдістегідей 254,65 мәнімен дәл келді.



4-сурет. Болжау нәтижесі



Талдау



Қанаттың мәселесіне талдау жасаңдар. Талдау барысында болжаудың теориялық және практикалық бөлігін ажыратып көрсетіңдер.



Жинақтау



Күнделікті өмірде кездесетін мәселелер ішінен болжам жасауға арнап 2 мысалдан құрастырыңдар. Құрастырылған мысалдардағы екі шаманың арасындағы байланысты дәлелдендер.



Бағалау



Қолда бар ақпараттың негізінде болжам жасау несімен маңызы? Мұндай болжамдардың қандай пайдасы бар?



Ойлануға берілген сұрақтар

1. «Динамикалық қатарлар» дегеніміз не? Олардың қандай қызметі бар?
2. Excel-де болжам жасауга қандай функциялар қолданылады?
3. КОРРЕЛ функциясының қызметін түсіндір.
4. «Сызықтық тренд» дегеніміз не?
5. Сызықтық трендтердің түрлері қандай?
6. ПРЕДСКАЗ функциясы қандай қызмет атқарады?



Тапсырма

1. Жинақтау барысында құрастырған екі мысалдың бірін компьютерде болжа.
2. Өздерің орналасқан аймақтағы күн райының 10 күн ішіндегі көрсеткіші берілген (*2-кесте*). 11-ші күні ауа температурасының қалай болатынын болжа.

2-кесте. Күн райының 10 күн ішіндегі көрсеткіші

Күндер	Сол күні тіркелген ауаның температурасы
1	+2
2	+3
3	+5
4	+4
5	+6
6	+10
7	+13
8	+9
9	+15
10	+18
11	?

3.6-3.7

Қолданбалы есептерді шешу



Excel программасының көмегімен қолданбалы есептерді шешуде кірістірілген функцияларды, деректердің түрлі типі мен форматтарын қалай пайдалануға және функция графиктерін қалай түрфызуға болады?



Ойлан

- «Қолданбалы есеп» дегеніміз не?
- Қолданбалы есептерді компьютерде және қолмен орындаудың ерекшеліктері неде деп ойлайсың?
- Функциялар көмегімен қандай қолданбалы есептерді шешуге болады?



Практикалық жұмыс

Қолданбалы есептер – адамның белгілі бір саламен байланысты іс-әрекетінен туындайтын есептер. Мысалы: математика, физика, химия, экономика, бухгалтерия т.б. салалардың есептері. Теориялық материалдарды практикалық түрфыдан шешуді қолданбалы есептерді шешу деп атайды. Excel программасында көптеген салалардың есептерін практикалық түрфыдан оқай әрі жылдам шешуге болады. Біз осындағы есептерді шешу жолдарын практикалық түрфыдан үйренеміз.



Математикалық есептерді шешуде электрондық кестені қолдану

Тапсырма. $y = 2x + 2$ және $y = -2x + 6$ функциялары берілген.

Осы функцияның графигін салып, қылышын нүктесін анықта.

Тапсырманы орындау алгоритмі:

1. Кестені жасауда x мәнін **-5**-тен **+5**-ке дейін **1** қадаммен алайық.

2. Берілген x мәнін $y = 2x + 2$ функциясына салып, y мәнін есептеу үшін, функцияға сәйкес $= 2 * B3 + 2$ формуласын **A3** үяшығына енгізіп, **A13** үяшығына дейін көшіріп қоямыз.



электрондық кесте – электронная таблица – spreadsheet
кірістіру – вставить – insert

3. Берілген x мәнін $y = -2x + 6$ функциясына салып, у мәнін есептеу үшін, функцияға сәйкес $= -2 * D3 + 2$ формуласын С3 үшіндеңін **C13** үшіндеңін дейін көшіріп қоямсыз.

4. $y = 2x + 2$ функциясының графигін салу үшін, **B3 : B13** пен **A3 : A13** арасындағы үшіндеңін **Вставка (Кірістіру) => Диаграммы (Диаграммалар) Точечная (Нұктелік) => Точечная с гладкими кривыми и маркерами (Тегіс нұктелік иір сызық және маркерлермен) командаларын орындаңыз, **1-суреттегі** диаграммадағы қызыл түсті сызықтық түзуді сымамыз.**

5. $y = -2x + 6$ функциясының графигін осы диаграммаға кірістіру үшін, **C3 : C13** арасындағы үшіндеңін **Вставить (Кірістіру)** батырмасын басамыз. Диаграммада көк түсті сызықтық түзу пайда болады. Функция графиктерінің қызылдықтарын нұктесін **1-суреттегі** 9-қатарға қарап анықтауға болады. Екі функция $x = 1$ мен $y = 4$ нұктелерінде қызылдықтары сада. Кестенің басқа жерінде мұндай сәйкестік жоқ.



1-сурет. Графиктердің қызылдықтары

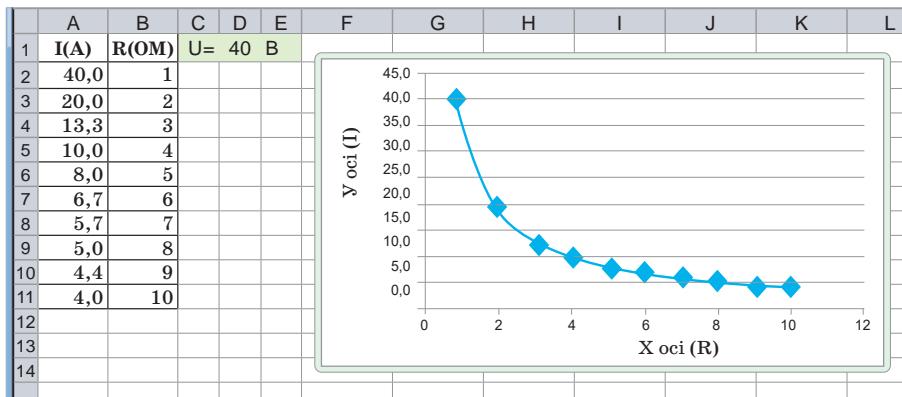
Физикалық есептерді шешуде электрондық кестені қолдану

Тапсырма. Ток күші мен кедергі арасындағы өзара байланыс $I(R)$ графигін құр. Кедергі (R) 1 Ω -мен 10 Ω аралығында 1 Ω қадаммен өзгереді.

Тапсырманы орындау алгоритмі:

Тапсырманы орындау барысында физикадағы тізбек бөлігі үшін Ω заңының формуласын жазамыз. $I = \frac{U}{R}$ формуладан байқалып түрғандай, ток күші мен кедергі арасында кері байланыс бар.

D1 ұяшығына кернеудің түрақты мәнін жазайық. Мысалы, U=40В. D1 ұяшығын абсолюттік ұяшық ретінде пайдаланамыз. В бағанына кедергілердің мәнін қоямыз, ал A2 ұяшығына =\\$D\$1/B2 формуласын жазып, A11-ге дейін көшіріп шығамыз (2-сурет).



2-сурет. График сзызу

Геометриялық есептерді шешуде электрондық кестені қолдану

Тапсырма. Бірнеше үшбұрыштың барлық қабыргасының ұзындығы берілген. Герон формуласын пайдаланып, осы үшбұрыштардың ауданын тап. Үшбұрыштардың ең үлкен және ең кіші аудандарын анықта.

Тапсырманы орындауда алгоритмі

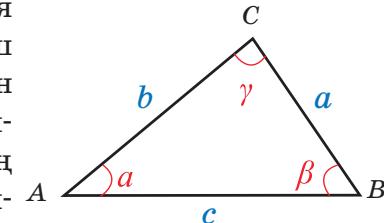
1. Тапсырманы орындауда геометрия курсындағы үшбұрыштың берілген үш қабыргасын пайдаланып, үшбұрыш ауданын есептеудің Герон формуласын – үшбұрыштың ауданын (S) оның қабыргаларының (a, b және c) ұзындықтары арқылы өрнектейтін формуланы қолданамыз:

$$S = \sqrt{p(p - a)(p - b)(p - c)}.$$

Мұндағы p – үшбұрыштың жарты периметрі: $p = \frac{a + b + c}{2}$.

2. Excel программасында 3-суреттегі үлгідегідей кесте жасаймыз. Үшбұрыш қабыргаларының мәндерін ойша беруге болады.

3. Е3 ұяшығына $= (B3+C3+D3)/2$ формуласын жазып, Е4-тен Е12 ұяшығына дейін көшіріп қоямыз.



4. F3 ұяшығына =КОРЕНЬ(E3*(E3-B3)*(E3-C3)*(E3-D3)) болатын Герон формуласын жазып, F12 ұяшығына дейін көшіріп қоямыз.

5. Ең үлкен ұшбұрыштың ауданын табу үшін, = МАКС(F3:F12) формуласын қолданамыз.

6. Ең кіші ұшбұрыштың ауданын табу үшін, = МИН(F3:F12) формуласын қолданамыз.

7. Барлық ұшбұрыштың орташа ауданын табу үшін,
= СРЗНАЧ(F3:F12) формуласын пайдаланып есептейміз.

A15						f_x	Ұшбұрыштардың орташа ауданы
1 2	A Ұшбұрыш №	Ұшбұрыш қабыргалары			Жарты периметрі	Ұшбұрыштың ауданын есептеу	
	B a	C b	D c				
3	1	10	12	13	17,5	57,00	
4	2	8	7	9	12	26,83	
5	3	5	4	3	6	6,00	
6	4	8	5	11	12	18,33	
7	5	4	6	8	9	11,62	
8	6	7	11	13	15,5	38,50	
9	7	14	15	16	22,5	96,56	
10	8	2	4	3	4,5	2,90	
11	9	4	4	7	7,5	6,78	
12	10	3	4	5	6	6,00	
13	Ең үлкен ұшбұрыштың ауданы					96,56	
14	Ең кіші ұшбұрыштың ауданы					2,90	
15	Ұшбұрыштардың орташа ауданы					27,05	

3-сурет. Excel-де есептеу

Экономикалық есептерді шешуде электрондық кестені қолдану

Тапсырма. Телефон компаниясы өз қызметіне ақы төлеудің жаңа жүйесін енгізді. Жаңа ақы төлеу жүйесі бойынша, егер абонент 1 айда A минутқа дейін сөйлессе, онда ол әрбір минут үшін 10 теңге, ал A минуттан артық сөйлессе, онда артық сөйлескен әрбір минут үшін B теңге төлейді. Осы компанияның қызметін алғаш пайдаланған 10 абоненттің төлейтін жалпы сомасын электрондық кесте көмегімен қалай есептеуге болады?

Тапсырманы орында алгоритмі

1. Тапсырманы орындауда 4-суреттегі үлгі бойынша кесте дайында.
2. Логикалық функцияны қолданып, сөйлесу A минуттан артып кеткен және A минутқа жетпей қалған жағдайларды қарастырып, сөйлесу бағасын табамыз. C4 ұяшығына

- =ЕСЛИ (B4>\$C\$1; \$C\$1*\$F\$1; B4*\$F\$1) формуласын жазып, C13 үяшығына дейін көшіріп қоямыз. Формулада сөйлесу уақытының шегі A, бағалар жазылған үяшықтарға абсолютті сілтеме қоямыз.
3. Сөйлесу уақыты А минуттан асып кеткен абоненттердің артық минуттарын D бағанына орналастырамыз. D4 үяшығына =ЕСЛИ(B4>\$C\$1; B4-\$C\$1; 0) формуласын енгізіп, D5-тен D13 үяшығына дейін көшіріп қоямыз.
 4. E4 үяшығына белгіленген мөлшерден артық минуттардың бағасын есептеудің = D4 * \$F\$2 формуласын жазып, E13-ке дейін көшіріп қоямыз.
 5. F бағанасында абоненттің бір айда төлеген жалпы сомасын F4 үяшығында = C4 + E4 формуласының көмегімен есептеп, F13-ке дейін көшіріп қоямыз.

	A	B	C	D	E	F
1		A= 300 минутқа дейін		Бағасы=10		
2		A= 300 минуттан жоғары		Бағасы = 25		
3	Үйлер	Бір айда телефонмен сөйлесуге кеткен уақыт (минутпен)	Сөйлесу бағасы (теңге)	Берілген мөлшерден артық (минут)	Сөйлесу бағасы (теңге)	Телефон байланысының жалпы сомасы (теңге)
4	1-үй	500	3000	200	5000	8000
5	2-үй	173	1730	0	0	1730
6	3-үй	468	3000	168	4200	7200
7	4-үй	392	3000	92	2300	5300
8	5-үй	100	1000	0	0	1000
9	6-үй	500	3000	200	5000	8000
10	7-үй	267	2670	0	0	2670
11	8-үй	308	3000	8	200	3200
12	9-үй	150	1500	0	0	1500
13	10-үй	87	870	0	0	870

4-сурет. Электрондық кесте

Колданбалы есептерді шешудің үлгілерін пайдаланып, тәменде берілген тапсырмаларды орында.



Практикалық жұмыс

1. $y = 3x - 2$ және $y = -x + 4$ функциялары берілген. Осы функцияның графигін салып, қызылысу нүктесін анықта.
2. $y = -x^2 + 2$ функциясының графигін сал.
3. 48, 84, 36, 60, 24 сандарының ЕYOB тап.
4. 21, 32, 7, 6, 12 сандарының ЕКОЕ тап.
5. Ток күші мен кернеу арасындағы өзара байланыс I(U) графигі *вольтамперлік сипаттама* деп аталады. Кернеу (U) 1В мен

- 10В аралығында 1В қадаммен өзгереді. Ток күші мен кернеу арасындағы өзара байланыс графигін сал.
6. Тұрақты 5 м/с жылдамдықпен қозгалатын дененің жүріп от-кен жолының уақытқа тәуелділік $S(t)$ графигін сал. Уақыт $t = 0...20$ секунд аралығында 2 с қадаммен өзгереді.
 7. Дененің кинетикалық энергиясы мен оның жылдамдығы арасындағы $E_k(V)$ байланыс графигін сал. Дене массасы тұ-рақты $m=100$ кг. Дене жылдамдығы $V=0...20$ м/с аралығында 2 қадаммен өзгереді. Байланыс графигі қандай түрге жатады? Себебін түсіндір.
 8. Массасы $m = 10$ кг судың температурасын өзгерту үшін қа-жетті жылу мөлшері Q мен дene температурасының өзгерісі Δt арасындағы байланыс графигін $Q(\Delta t)$ сал. $\Delta t = 0...100^\circ\text{C}$ аралығында 10°C қадаммен өзгереді. $C=4200 \text{Дж}/\text{кг}^\circ\text{C}$.
 9. Кестеде берілген деректерді пайдаланып, белгісіз шамаларды тап.

№	A	B	C
1	Шеңбердің радиусы R (м)	Шеңбердің диаметрі d (см)	Шеңбердің ұзындығы L (дм)
2	20	?	?
3	38	?	
4	15		
5	45		

10. Төмендегі кестеде үшбұрыш төбелерінің координаталары берілген. Мысалы, бір төбесі $A(x_1, y_1)$. Осы координаталарды пайдаланып, үшбұрыш қабыргасының ұзындықтарын тап. Бұл қабыргалардан үшбұрыш құруға бола ма? Анықта. Егер үшбұрыш құруға болса, «Иә» немесе «Жоқ» сөздері әрбір үшбұрыш қабыргаларының жанына жазылсын.

Ескерту: Қабырганың ұзындығын табу мен үшбұрыш құру шартын анықтау үшін, геометрия курсындағы екі нүктे арасындағы қашықтық формуласы мен үшбұрыштарды құру шарттарын пайда-ланамыз.

	A	B	C	D	E	F	G	H	I	J
1	A		B		C		Қабырганың ұзындығы			
2	x_1	y_1	x_2	y_2	x_3	y_3	a	b	c	Үшбұрыш құрай ма? «Иә» немесе «Жоқ»
3	2	3	6	8	17	8				

(Кестенің жалғасы)

4	-4	-3	8	9	-7	1				
5	4	6	9	9	5	7				
6	9	8	90	99	3	3				
7	7	7	11	20	17	26				

11. «Математика» мен «Физика» пәндері бойынша ҮБТ тапсырған 10 оқушының жинаған үпайы жазылған кесте құр. Әр пән бойынша жинауға мүмкін болатын үпай саны – 45. Оқушылардың жинаған үпайларын бағалау шөкілдеріне өткіз. Тест қорытындысы бойынша неше оқушы пәндер бойынша «5», «4», «3» және «2» бағаларын алды? «5» – 81–100%, «4» – 61–80%, «3» – 41–60% аралығында, одан төмен жинағандар «2» деген баға алады.

№	A	B	C	D	E
1	Қатысушылар	Математика	Баға	Физика	Баға
2	№1 оқушы	25		18	
3	№2 оқушы	40		29	
4	№3 оқушы	12		14	
5	№4 оқушы	32		36	
6	№5 оқушы	40		40	
7	№6 оқушы	20		5	
8	№7 оқушы	13		6	
9	№8 оқушы	7		21	
10	№ 9 оқушы	4		18	
11	№ 10 оқушы	35		25	
12	«2» бағасының саны				
13	«3» бағасының саны				
14	«4» бағасының саны				
15	«5» бағасының саны				

12. Қазақстан халқының соңғы 10 жылдағы демографиялық көрсеткіші берілген. 2025 жылы Қазақстан халқының саны қанша болады? Болжам жаса.

№	A	B
1	Жылдар	Халық саны
3	2011	16 559 000
4	2012	16 900 000
5	2013	17 035 000
6	2014	17 289 000
7	2015	17 557 000
8	2016	17 818 000
9	2017	18 014 000
10	2018	18 137 000
11	2019	18 632 500
12	2020	19 180 920
13	2021	?
14	2022	?

Шығармашылық тапсырмалар

1-тапсырма. Электрондық кестеде берілген үлгі бойынша атмосфералық қысымның апталық өзгеруіне баға беретін кесте жаса. Тапсырманы орындауда абсолютті сілтемелер мен логикалық функция қызметін пайдалануға болады.

Қалыпты атмосфералық қысым =760 мм сынап бағаны.

Егер:

- 755–765 мм сынап бағаны аралығында болса, «Талапқа сай».
- 720–754 мм сынап бағаны аралығында болса, «Талаптан төмен».
- 766–780 мм сынап бағаны аралығында болса, «Талаптан жоғары».

№	A	B	C	D	E
1	№	Апта күндері	Атмосфералық қысым (мм сынап бағанасы)	Қалыпты жағдайдан ауытқу	Қалыпты жағдайдан ауытқуды бағалау
2	1	Дүйсенбі	746	16	Төмен
3	2	Сейсенбі	764	4	Қалыпты
4	3	Сәрсенбі	755	5	Қалыпты
5	4	Бейсенбі	778	32	Жоғары
6	5	Жұма	719	41	Төмен
7	6	Сенбі	735	25	Төмен
8	7	Жексебі	758	2	Қалыпты

Орындаушыға арналған дескрипторлар:

- кестені толық форматтау;
- ұяшықтарда абсолютті сілтемені қоя білу;
- логикалық функцияны пайдаланып, ұяшықтарға өрнекті дұрыс жазу.

2-тапсырма. Оқушылардың дene бітімі

8-сынып оқушыларының дene бітімін анықтау керек.

Салмақ нормасы = (Бойы – Салмағы) • 1,1

№	A	B	C	D	E
1	Оқушылар	Бойының ұзындығы (см)	Салмағы (кг)	Салмақ нормасы (кг)	Нормадан ауытқу (%)
2	1-оқушы	161	49		
3	2-оқушы	157	50		
4	3-оқушы	149	55		
5	4-оқушы	172	60		
6	5-оқушы	165	50		
	Орта мәндер	?	?		

Орындаушыға арналған дескрипторлар:

- кестені толық форматтау;
- ұяшықтарда формууланы пайдалану;
- ұяшықтарда деректердің типін пайдалану;
- ұяшықтарда функцияны қолдану.

3.8**Excel программасында жобалық жұмыстар орындау**

Ақпаратты электронды кестелерде өңдеу барысында алған білімді өмірде қалай пайдалануға болады?

**Ойлан**

- Электронды кестелерді өңдеу ортасында алған теориялық және практикалық білімді қалай жүйелеуге болады?

Шығармашылық-практикалық тапсырмалар

Төменде берілген жоба тақырыптарының ішінен бір тақырыпты таңдап алып, жеке немесе жүппен орында. Жобаны орындау барысында теориялық материалдарды ұсынумен қатар жобаның программалық өнімдерін жасау өте маңызды.

1-жоба тақырыбы***Excel программасын түрмисста қолданудың тиімді жолдары*****Қысқаша мазмұны**

Excel программасын қарастырып, күнделікті өмірде бұл программа қай салаларда қолданылып жатқанына тоқталу керек. Күнделікті түрмисқа қажет есептеудің бірінің Excel программасындағы жобасын ұсыну.

2-жоба тақырыбы***Бізді қоршаган диаграммалар*****Қысқаша мазмұны**

Excel программасында деректерді диаграмма тәсілінде ұсынуға тоқталу керек. Диаграммаларды пайдалану тарихын, олардың түрлері мен қажеттілігін зерттей отырып, күнделікті өмірден алған мысалдармен дәлелдер келтіру. Нәкты бір тақырыпты таңдап, ондағы деректерді диаграмма арқылы сипаттау жолдарын ұсыну. Мысалы, мектеп оқушылары арасында йод тапшылығын зерттеу нәтижелерін диаграмма арқылы көрсету.



электрондық кесте – электронная таблица – spreadsheet
сызықтық функция графигі – график линейной функции – graph linear functions

3-жоба тақырыбы

Сызықтық, параболалық және гиперболалық функциялардың графигін Excel программасының көмегімен зерттеу

Қысқаша мазмұны

Математикалық функция түрлеріне тоқтала отырып, Excel программасында деректерден сызықтық, параболалық және гиперболалық функциялардың графиктерін құру жолдарын ұсынып, жоба дайындау. Мысалы, $y = ax^2 + b$.

4-жоба тақырыбы

Excel программасында сызықтық тендеулер жүйесін шешу жолдары

Қысқаша мазмұны

Excel программасында сызықтық тендеулер жүйесін шешу жолдарының алгоритмін жасақтауға зерттеу жүргізу. Есептеуге көмектесетін Excel программасында автоматтандырылған үлгілерді жасау және ұсыну жолдарын зерттеу. Программа өнімін ұсыну. Таныстырылым жасау.

5-жоба тақырыбы

Excel программасында тест жасау

Қысқаша мазмұны

Excel программасында тест жасау жолдарын зерттеу. Жасаған тест үлгілерінің ерекшеліктерін түсіндіру. Артықшылықтары мен кемшіліктеріне тоқталып, программа өнімін ұсыну. Таныстырылым жасау.

6-жоба тақырыбы

Білім сапасын бағалауда Excel программасын тиімді қолдану

Қысқаша мазмұны

Мектеп өкімшілігі мен мұғалімдері үшін оқушылардың тоқсан сайын білім сапасын бағалауға көмектесетін Excel программасында автоматтандырылған үлгілерді жасау және ұсыну жолдары жайлы әңгімелу. Программа өнімін ұсыну. Таныстырылым жасау.

7-жоба тақырыбы

Excel программасының көмегімен «Физика» пәнінен зертханалық жұмыстарды орындауда есептеу жұмыстарының дәлдігін арттыру

Қысқаша мазмұны

Мектеп программасындағы «Физика» курсында зертханалық жұмыстарды қолмен есептеу қыын. Excel программасы көмегімен зертханалық жұмыстардағы есептеулерді автоматты түрде орындаітын, абсолюттік пен салыстырмалық қателерді дәл табу формулалары енгізілген программа үлгілерін жасау және оны ұсыну. Программа өнімін көрсету. Таныстырылым жасау.

8-жоба тақырыбы

Қолданбалы математика есептерінде Excel программасын пайдалану

Қысқаша мазмұны

Қолданбалы математика, яғни математиканы өмірде қолдану жағдайларына байланысты болатын есептерді шешуде Excel программасын пайдалану жайлыштырған.

Жобаны қорғау кезінде:

- таңдаған тақырыбыңмен таныстыруға;
- тақырыпты не себепті таңдағаныңды түсіндіруге;
- мақсат пен міндеттеріңді айқындаңыз беруге;
- пайдаланған құралдардың атап шығуға;
- тақырып бойынша қысқаша шолу жасауға;
- жобаны орындауда пайда болған ойыңды ортаға салуға;
- жоба барысында үйренген жаңа білім мен дағды жайлыштырғандағы;
- жоба қорытындысымен таныстыруға;
- тыңдаушылардың пікірі мен бағасын тыңдаңыз, қабылдай білуге дайын бол.

Жобаны бағалау барысында:

- жобаны өз бетінше орындауды;
- тақырыптың өзектілігі;
- тақырыпты аша білуі;
- мәселені ұсынудың өзіндік жолы;
- қорғаудағы өртістік қабілеті мен өзін үстай білуі;
- техникалық құралдар мен көрнекіліктегі қолдана білуі;
- қойылған сұрақтарға дұрыс әрі нақты жауап береді.

Жоба соңында есебін жаз.

- Мен неге бұл тақырыпты таңдадым?
- Жаңадан нені білдім, нені үйрендім?
- Керекті ақпараттарды қалай іздеп таптым?
- Ақпараттарды қалай өндедім?
- Теория мен практиканы қалай ұштастырудым?
- Қай нәрсе ұтымды шықты?
- Нені орында алмадым?
- Не оңай болды?
- Не қиын болды?
- Жобаны қайта орындасам, тағы да қалай толықтыраңым?
- Менің жобаны орындаудан алған өсерім қандай?

4.1

For циклі



Python программалау тілінде параметрлі цикл операторы қалай қолданылады?



Ойлан

- Цикл жайлы не білесің?
- Бұл үғымның біздің өмірімізben қандай байланысы бар?
- Табиғат пен адам өмірінде циклдің маңызы қандай?
- Математикада циклдерге не жатады?



Жаңа білім

Циклдің табиғат пен адам өміріндегі маңызы зор. Табиғатта, адам өмірі де белгілі бір циклден тұрады. Цикл – тұрақты қайталанып тұратын оқиғалар жиынтығы. Мысалы, жыл мезгілдері, күн, апта мен ай – бұлардың бәрі табиғатта қайталанып тұратын циклдер. Циклдердің түрі көп. Оларға экономикалық, тарихи, программалау, математикалық, тіршілік циклдері т.б. жатады. Программалауда циклдік алгоритмдер өте маңызды. Олардың көмегімен табиғатта кездесетін цикл түрлерінің алгоритмін құрастырып, компьютерде модельдеуге болады.

Бірнеше рет қайталанып орындалатын белгілі бір іс-әрекеттерді сипаттайтын алгоритмдер тобы **циклдік алгоритмдер** деп аталады.

Циклдік алгоритмдерді пайдалануда программалау тілінің операторы түрінде оны қысқаша жазу мүмкіндігі болады. Циклдік алгоритмдерді программалау тіліне аударатын арнайы операторлар бар. Циклдік операторлар қолдануына қарай екі түрге жіктеледі (*1-схема*).



1-схема. Программаудағы циклдер түрлері

Егер алгоритмді құру кезінде қайталанудың параметрлері алдын ала белгілі процесті үйімдастыру қажет болса, онда параметрлі цикл операторы қолданылады. Мұндай операторға **for** операторы жатады. Python-да **for** операторы белгілі бір есептеуде көрсетілген мәнге жеткенге дейін жұмыс істейді. Параметрлі цикл жұмысын үйімдастыру үшін, **for** операторын **range()** функциясымен бірге қолдану керек.

Жазылуы:

for i in range(*k, n, x*):

<оператор немесе өрнек>;

range (*k, n, x*) функциясы *k*-дан *n* – 1-ге дейін *x* қадаммен сандар тізімін жасайды.

Мысалы: *k* = 1, *n* = 11 және *x* = 2 болса, онда **for i in range(1, 11, 2)** цикл қатары *i* шамасының мәні 1-ден 10 аралығында 2 қадаммен өзгегерді деп оқылуы керек.

for операторында range () функциясын қолдану түрлері

for i in range(<i>n</i>)	for i in range(<i>k, n</i>)	for i in range(<i>k, n, x</i>)
for i in range(6): print(i)	for i in range (6, 11): print(i)	for i in range(1, 11, 2): print(i)
Нәтиже:	Нәтиже:	Нәтиже:
0	6	1
1	7	3
2	8	5
3	9	7
4	10	9
5		

For операторының қолданылуына екі түрлі мысал келтіруге болады.

1. Цикл параметрінің мәні өсу ретімен (**1-мысал**).

2. Цикл параметрінің мәні кему ретімен (**2-мысал**).

Цикл жұмысын түсіну үшін қарапайым мысалдар қарастырайық.

1-мысал. 1 мен 100 аралығындағы сандардың қосындысын табатын программа жаз. Мысалдың программа коды *1-curretme* берілген. Программа коды орындалғанда экранға 5050 шығады.

```
i=s=0
for i in range (1,101):
    s+=i
print (s)
```

1-curretme. 1-мысалдың
программа коды



цикл – цикл – cycle
диапазон – диапазон – range

2-мысал. 1 мен 10 аралығындағы сандарды көрі ретпен экранға шығар.

Мысалдың программа коды 2-сүретте берілген. Программа коды орындалғанда, экранда 10 9 8 7 6 5 4 3 2 1 сандары пайда болады.

```
i=0  
for i in range (10,0,-1):  
    print(i)
```

Ln:4 Col:0

2-сүрет. 2-мысалдың программа коды



Практикалық жұмыс

8 «А» сыныбында N ($0 < N < 30$) оқушы бар. Оқушылардың бойлары көрсетілген. Егер сыныптағы оқушының бойы 140 смден қысқа болса, ол аласа бойлы оқушыға жатады. Сыныпта неше оқушының бойы аласа екенін анықтайдын программа кодын жаз.

Практикалық жұмыстың программа кодына (3-сүрет) түсініктеме беріп өтейік. Берілген есептің программа кодын жазуда циклдік

```
N=int(input('Oqushy sany ='))  
i=x=z=0  
for i in range(1,N+1):  
    print(str(i),end='')  
    x=int(input('-Oqushynyn boiy='))  
    if x<140:  
        z+=1  
print('Boiy alasa oqushylar sany =',z)
```

Ln:9 Col:0

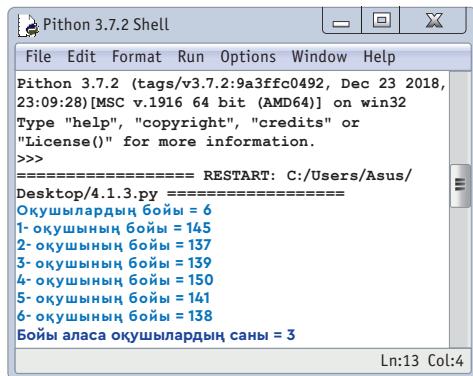
3-сүрет. Программа коды

және тармақталу операторларын бірге қолданамыз. Барлық оқушының бойын N рет енгізу үшін, енгізу операторын циклдің ішіне орналастырып, әр цикл қадамында оқушының бойының ұзындығын x айнымалысына оқытамыз. Одан кейінгі қадамда оқушы бойының ұзындығын 140-пен салыстырып тексереміз. Көріп түрғандай, цикл денесінде бір емес, үш оператор жұмыс істейді. Үш операторды бір циклдің ішінде пайдалану үшін, **for** операторынан кейінгі жолдарда 4 бос орын (пробел) қою керек. Мұнда $z = 1$ z мәнін 1-ге арттыратын $z = z + 1$ өрнегінің қысқа жазылу түрін көрсетеді. Цикл ішіндегі **print (str (i), end="")** жолы оқушының нөмірі бойынша енгізу керек-

тігін көрсету үшін қолданылған. 4-суретте программа жұмысының нәтижесі берілген.

Шексіз цикл

Кейбір программаларда циклді басқару мүмкін болмай қалады. Бұл – программалау кезінде жіберген қателерден туындастын жағдай. Мұндай циклдер шексіз циклдер деп аталады. Шексіз цикл кезінде программаны тоқтату үшін, *Ctrl + C* перенелерінің комбинациясын бір мезгілде басу керек.



The screenshot shows the Python 3.7.2 Shell window. The code is as follows:

```
>>> ===== RESTART: C:/Users/Asus/Desktop/4.1.3.py =====
Оқушылардың бойы = 6
1-окушының бойы = 145
2-окушының бойы = 137
3-окушының бойы = 139
4-окушының бойы = 150
5-окушының бойы = 141
6-окушының бойы = 138
Бойы аласа оқушылардың саны = 3
```

Ln:13 Col:4

4-сурет. Программа жұмысының нәтижесі

Талдау

Практикалық жұмыста орындалған программа кодын (3-сурет) талдаңдар. Егер есеп шартын «оқушылардың бойы 140 пен 155 см аралығында» деп өзгертсек, программа кодына қандай өзгеріс жасау керек? Себебін түсіндіріңдер.

Жинақтау

Параметрлі циклдер көмегімен орындалатын қарапайым бір мысал құрындар. Құрган мысалдарыңың Python тіліндегі программа кодының жобасын дайындаپ, ұсынындар.

Бағалау

Циклдік операторларды программалауда қолданудың маңызына баға бер.

Ойлануға берілген сұрақтар

1. Циклді қалай түсінесің?
2. Циклдің адамның өміріне өсері қандай?
3. Программалауда циклдердің қандай түрлері бар?
Олардың айырмашылығы қандай?
4. Цикл алгоритмінің тармақталу алгоритмінен қандай ерекшеліктері бар?



Тапсырма

*

1-тапсырма. K мен N ($1 \leq K \leq N \leq 100$) аралығындағы барлық жүп санды экранға шығаратын программа құр.

Мысалы	Нәтиже
23 35	24 26 28 30 32 34

*

2-тапсырма. N ($10 \leq N \leq 99$) санына дейінгі екітаңбалы сандар арасында екі цифры да бірдей сандарды экранға шығаратын программа құр. Екітаңбалы сан 10-нан басталады.

Мысалы	Нәтиже
34	11 22 33

**

3-тапсырма. 8-сынып оқушыларының арасында баскетбол үйірмесіне қатысатындарға бойының ұзындығы бойынша іріктеу жүріп жатыр. Іріктеуге барлығы N ($130 \leq$ оқушы бойы ≤ 200) оқушы қатысқан өрі бойларының ұзындығы K сантиметрден биіктегі ғана қабылданған. Үйірмеге барлығы неше оқушы қабылданған? Ең ұзын оқушы нешінші орында тұр? Мысалда берілген оқушылардың бойы тігінен енгізіледі.

Мысалы	Нәтиже	Түсініктеме
7 170 165 180 171 135 143 175 169	3 2	Бойы 170 см 3 оқушы бар. Ең ұзын оқушы 2-орында.



Үй тапсырмасы

Программа жұмысын аяқтаған кезде программада кездесетін `print()` операторы экранға қандай мәндерді шығарады?

1)

```
N = 5
i = x = z = 0
for i in range(1, N + 1):
    x = int(input())
    if x % 10 == 4:
        z += 1
print(z)
```

2)

```
N = 5
i = x = s = 0
for i in range(1, N + 1):
    x = int(input())
    if x % 2 == 1:
        print(x % 10)
        s += x % 10
print(s)
```

4.2

Кірістірілген циклдерді үйымдастыру



Python программалау тілінде кірістірілген циклдерді қалай үйымдастыруға болады?



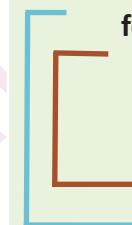
Ойлан

- «Кірістірілген цикл» дегеніміз не?
- Екі **for** циклін бір-біріне кірістіріп пайдалансақ, қандай нәтиже алуымыз мүмкін?
- Кірістірілген циклдерді қандай есептерді шешуде қолдануға болады деп ойлайсың?



Жаңа білім

Бірінің ішіне екіншісін кірістіріп пайдаланған бірнеше қарапайым циклдер **кірістірілген циклдер** деп аталаады. Кірістірілген циклдерде ішкі цикл сыртқысына толық енуі керек. Бір-біріне кірістірілген екі қарапайым циклдің жазылу түрі *1-схемада* берілген. Кірістірілген циклдерді күрделі цикл деп те атайды.



```
for i in range(1, n):
    for j in range(1, n):
        <1-оператор>
        <2-оператор>
        ...
        ...
```

1-схема. Екі қарапайым циклдің жазба түрі

Циклде қолданылған n саны бүтін сандар типіне жатады (*1-схема*). Егер программада бір цикл қолданылса, онда ол $n-1$ рет жұмыс істейді. Егер программада кірістірілген 2 цикл қолданылса, онда сыртқы цикл n рет орындалғанда ішкі цикл $n \cdot n$ рет жұмыс істейді. Мысалы, төменде берілген циклде сыртқы цикл 10 рет, ішкі цикл 100 рет, жалпы 110 рет жұмыс істейді.

1) **for i in range(1, 11):**
 for j in range(1, 11):

2) **for i in range(1, 101):**
 for j in range(1, 101):
 for k in range(1, 101):

Кірістірілген циклдер көмегімен барлық нұсқа тексерілетін жағдайларды қарастырылатын есептерді шығаруға қолданған тиімді. Бірақ циклдерді бір-біріне 2-3-тен артық кірістіру тиімсіз. Ол про-



кірістірілген цикл – вложенный цикл – embedded loop

циклды орындау – выполнение цикла – loop execution

граммалың жұмыс істеу уақытын ұзартады. Егер программада үш қарапайым циклді бір-біріне кірістіріп қолдансақ, онда программалың жұмыс істеу уақыты шамамен n^3 -ге тең болады. Бұл шамалың өте үлкен екенін мына мысалдан да білеміз. Мысалы, $n = 100$ болса, онда кірістірілген 3 цикл 1010100 рет жұмыс істейді. Егер циклдің 1 000 000 рет орындалуына шамамен 1с уақыт кететінін ескерсек, онда $n = 1000$ болғанда, цикл шамамен 10^9 рет, яғни млрд рет артық жұмыс істейді. Бұған шамамен 1000 секунд, яғни 17 минуттай уақыт кетеді. Сонда компьютерде бір есептің нәтижесін құтуге 17 минуттай қажет болады. Сол себепті де мұндай есептерді қарастырганда, тиімді алгоритмдер арқылы цикл жұмысын азайту қарастырылады.

Кірістірілген циклдер көмегімен сандық және символдық деректерді тіктөрбұрышты кесте түрінде экранға шығаруға болады. Сандық деректерді тіктөртбұрышты кесте түрінде берсе, түсінуге оңай. Деректерді тіктөртбұрыш түрінде экранға шығаруға тәмендегі цикл қатарларының коды жазылады. Мысалы, өлшемі 5x5 тіктөртбұрышты көбейту кестесін құру үшін тәмендегі программа кодын жазады (*1-сурет*), ал программа орындалғандағы нәтиже *2-суретте* берілген.

```
i=j=0
for i in range(1,6):
    for j in range(1,6):
        print(i*j,end="\t")
    print()

Ln:6 Col:0
```

```
File Edit Format Run Options Window Help
Python 3.7.4 (tags/v3.7.4:e09359112, Jul 10 2019, 21:42:51) [MSC v.1916 64 bit (AMD64)] on win32 Type "help", "copyright", "credits" or "license" for more information.
>>> == RESTART: C:\Users\User\AppData\Local\Temp\4.2.1.py
1 2 3 4 5
2 4 6 8 10
3 6 9 12 15
4 8 12 16 20
5 10 15 20 25
>>>
```

1-сурет. Программа коды

2-сурет. Программаның нәтижесі

Программа кодынан көріп түрғанымыздай, сыртқы цикл 1 рет орындалғанда, ішкі цикл 5 рет орындалады. Деректерді тіктөртбұрыш пішінде берген кезде **print()** операторын дұрыс орналастыру маңызды. Бұл жерде "\t" деректердің арасын табуляция арқылы бірдей қашықтыққа жылжытады. Кірістірілген циклдерді қолдану барысында тәменде көрсетілген қателер жиі кездеседі.

Кірістірілген циклдерді қолдануда жиі кездесетін қателер:

- ішкі және сыртқы циклдердің цикл параметріне бірдей айнымалы жазу;
- кірістірілген ішкі, сыртқы циклдерге қатысты операторларды орналастыруда бос орындардың санынан қателесуден оператор немесе өрнектердің циклге енбей қалу жағдайы;
- цикл параметрлеріне нақты сандарды қолдану.



Практикалық жұмыс

Тапсырма. Жай сан. Ұзындығы N болатын сандар тізбегі берілген ($N < 100$). Сандар тізбегіндегі жай сандарды анықтап, экранға шығар. Жай сан – өзіне және 1-ге ғана бөлінетін сан.

Есепті шешу алгоритмі. Есепті орындау үшін кірістірілген циклдерді пайдаланаңыз. Санның жай сан екенін тексеру үшін, санның квадрат түбірін бүтін мәнге жуықтаймыз. Тексерілуі керек сандарды сыртқы циклді оқуға пайдаланамыз. Ішкі циклді берілген сандарды жай сан шартына тексеру үшін пайдаланамыз. Ішкі цикл параметрі 2-ден басталып, бүтін мәнге жуықталған санның түбіріне дейін 1 қадаммен өзгереді. Осы сандар аралығында тексерілетін санның бөлгіші болмаса, ол жай санға жатады. Есептің программа коды *2-суретте*, ал программаның орындалу нәтижесі *3-суретте* берілген.

Мысалы	Нәтиже
5	5 197 47
5	
45	
197	
47	
18	

```

4.2.2.py - C:\Users\User\AppData\Local...
File Edit Format Run Options Window Help
import math
N=int (input())
i=j=x=0
for i in range (1,N+1):
    x=int(input())
    z=0
    for j in range (2,round(math.sqrt(x))):
        if x%j==0:
            z=1
    if z==0:
        print(x)
Ln:5 Col:17

```

2-сурет. Есептің
программа коды

```

Python 3.7.4 Shell
File Edit Format Run Options Window Help
Python 3.7.4 (tags/v3.7.4:9b2fa0c, Oct 11 2018, 27 54:52) [MSC v.1916 64 bit (AMD64)]
Type "help", "copyright", "credits" or "license" for more information
>>> === RESTART: C:\Users\User\AppData\Local...
5
45
197
47
18
5
197
47
>>>

```

3-сурет. Программаның орындалу
нәтижесі

Python-да математикалық функцияларды қолданатын арнайы модульдер бар. Программада математикалық модульді іске қосу үшін **import** командасын қолданамыз. Мысалы: **math.sqrt(x)** – x салының квадрат түбірін береді, онда **round (math.sqrt(18))= 4** болады.

Талдау



Практикалық тапсырмадағы «жай сандарды анықтау» программа кодына талдау жасай отырып, программаның өрбір жолына түсініктеме жазыңдар.



Ойлануға берілген сұрақтар

- Кірістірілген циклдердің жай циклдерден қандай айырмашылығы бар?
- Кірістірілген циклдердің орындалу уақытын қалай есептеуге болады?
- Бірнеше циклді кірістіріп пайдаланудың программа жұмысына кері әсері қандай?
- Деректерді тіктөртбұрышты кесте түрінде беру программасын жазуда қай оператордың жұмысы маңызды?



Тапсырма

1-тапсырма. Кірістірілген циклдер көмегімен деректерді $n=5$ жағдайы үшін тіктөртбұрыш түрінде экранға шығар. Тапсырманы орындауда шарт тексеру операторымен циклдерді кірістіріп пайдаланған тиімді.

1)	2)	3)	4)	5)
1 1 1 1 1	1 1 1 1 1	1 1 1 1 1	1 0 0 0 0	0 0 0 0 1
1 1 1 1 1	0 0 0 0 0	2 2 2 2 2	0 1 0 0 0	0 0 0 1 0
1 1 0 1 1	1 1 1 1 1	3 3 3 3 3	0 0 1 0 0	0 0 1 0 0
1 1 1 1 1	0 0 0 0 0	4 4 4 4 4	0 0 0 1 0	0 1 0 0 0
1 1 1 1 1	1 1 1 1 1	5 5 5 5 5	0 0 0 0 1	1 0 0 0 0

2-тапсырма. Программаның орындалуы аяқталған соң, x айнымалысы қандай мәнге ие болады?

1) $i = j = x = 0$
 $\text{for } i \text{ in range}(1, 6):$
 $\quad \text{for } j \text{ in range}(i + 1, 6):$
 $\quad \quad x += 1$

2) $i = j = x = 0$
 $\text{for } i \text{ in range}(1, 6):$
 $\quad \text{for } j \text{ in range}(i + 1, 6 - i):$
 $\quad \quad x += 1$

3) $i = k = x = 0$
 $\text{for } i \text{ in range}(1, 5):$
 $\quad \text{for } k \text{ in range}(i - 1, i + 1):$
 $\quad \quad x += 2$

4) $i = k = 0; x = 10$
 $\text{for } i \text{ in range}(1, 4):$
 $\quad \text{for } k \text{ in range}(0, i):$
 $\quad \quad x = x + (k - 1)$



Үй тапсырмасы

«Кірістірілген циклдер» тақырыбына арналған бір есептің мәтінін құрастыр. Құрастырган есебіңің шешу жолы мен программасын ұсын.

4.3

While циклі



Python программалау тілінде шарт циклі операторын қалай қолданамыз?



Ойлан

- «Циклдік процесс» дегеніміз не?
- Циклдік процестерге құнделікті өмірден қандай мысалдар келтіре аласың?
- Программалауда циклдердің қандай түрлері бар?
Есіне түсір.
- «Шарт» пен «цикл» ұғымын қалай байланыстыраң едің?
Мысал келтір.



Жаңа білім

Өткен тақырыптарда **for** параметрлі циклінің жұмысымен таныстық. Енді циклдің екінші бір түрі – шарт циклі **while**-дың жұмысымен танысадайық. Шарт циклі **while**-дың жұмысы істей реті 1-схемада берілген. Шарт циклі **while** (өзірге) резервтік сөзі мен цикл шарты <шарт> логикалық типтегі өрнектен тұрады. Цикл шарты логикалық өрнек түрінде жазылады. Логикалық өрнектің нәтижесі бульдік типтегі нәтиже болады. Шарт циклінің логикалық өрнегі **true** (ақиқат) немесе **false** (жалған) мәнін ғана қабылдайды. Цикл ішіндегі операторлар орындалуды баста-мас бұрын цикл шартының мәні тексеріледі. Шарт «акиқат» мәнін қабылдаса, цикл өз жұмысын жалғастырады. Егер өрнектің мәні «жалған» болса, онда цикл жұмысын аяқтайды. Басқа операторды немесе өрнекті цикл ішінде орналастыру үшін, **while** мен цикл шарты жазылғаннан кейінгі жолдарда 4 бос орын тастан жазу қажет. Цикл ішіне орналасқан операторлар мен өрнектер цикл шарты жалған мәнін қабылдағанша ретімен орындала береді. Егер цикл шарты

while <шарт>:

цикл денесі
<1-оператор не өрнек>
<2-оператор не өрнек>
...
< n – оператор не өрнек>

1-схема. while циклінің жазба түрі



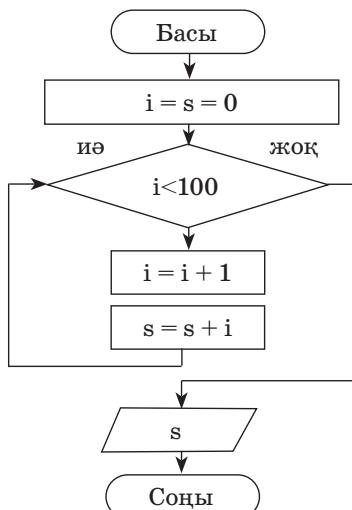
әзірге циклі – цикл пока – **while loop**

цикл шарты – условие цикла – **loop condition**

жалған болса, онда цикл денесі орындауды бастамай-ақ өз жұмысын тоқтатады.

While операторының жұмыс істеу негіздерін түсіну үшін мысалдар қарастырайық. Ол үшін «параметрлі циклдер» тақырыбында алғынған 1-мысалды тағы да қайтадан қарастырамыз. Екі цикл операторының жұмысымен бірдей мысал арқылы танысу олардың жұмыс ерекшеліктерін түсінуді оңайлатады.

Мысалы: 1 мен 100 аралығындағы сандардың қосындысын табатын программа жаз.



1-сурет. Блок-схема

Мысалдың шарты бойынша 1-ден 100-ге дейінгі сандарды қосуды шартты цикл көмегімен орындауға болады (1-сурет). Шарт тексеру циклі «*i* шамасының мәні 100-ден кіші ме?» деген сұрағын әр жолы қайталап отырады. Циклдік алгоритмдердің блок-схемасы 1-суретте берілген. Бұл цикл жұмысы *i* шамасының мәні 100-ден үлкен болғанда барып тоқтайды. Цикл шарты жалған болғанға дейін *s* қосқышына *i*-дің мәнін қосып отырады (2-сурет).

```
i=s=0
while i<100:
    i+=1
    s+=i
print(s)
```

2-сурет.
Программа коды

Шарт циклі while-дың ерекшелігі:

- циклдің қайталану шарты цикл денесі жұмыс істемей түрып тексеріледі;
- шарт циклі цикл саны белгісіз болған жағдайларды қарастырғанда қолданылады.



Практикалық жұмыс

Практикалық есеп. Берілген *N* натураган санды цифрларының қосындысының жұп немесе тақ екенін анықта. Егер қосынды тақ болса, «1» немесе «2» санын экранға шығар.

Мысалы	Нәтиже
4856	1
7517	2

Есептің программа кодын құру жолында бір ғана мәселе туын-дайды. Санның цифрларын қалай жеке-жеке қарастыруға болады? Математикада N санының цифрларын жеке-жеке қарастыру үшін заңдылықтарға жүгінеміз.

Сондай заңдылықтардың бірі: «Кез келген санды 10-ға бөлгендеге қалатын қалдық санның соңғы цифрына тең». Егер $N = 148$ болса, $148 : 10 = 14$ және қалдық 8-ге тең. Санның соңғы цифры да 8-ге тең. Есептің алгоритмдік шешімі ретінде осы заңдылықты программа құруда пайдаланамыз (*3-сурет*).

```
s=k=0
N=int(input())
while N!=0:
    k=N%10
    s+=k
    N//=10
if s% 2==1:
    print(1)
else:
    print(2)
```

Ln:11 Col:0

3-сурет. Программа коды



Талдау



Практикалық жұмыста орындалған есептің программа кодын (*3-сурет*) талдандар. Программаның әрбір қатарының атқаратын қызметіне түсініктеме жазындар.



Жинақтау



Практикалық жұмыста берілген есептің шартын өзгертіп, берілген санның құрамындағы нөлдерді анықтау керек болса (*3-сурет*), онда программа кодына қандай өзгеріс жасауға болады? Программа кодын өзгертуге ұсыныс жасандар. Ұсыныстарынды дәлелдендер.



Бағалау



Шікірталас

Шарт циклі **while** мен **for** параметрлі циклінің қызметінде-гі негізгі үқастық пен айырмашылықтарды салыстырып, бағала. Екі оператордың орнына біреуін қалдырып, екіншін алышп тастауға бола ма? Егер цикл операторларының санын қысқартсақ, қайсысын алышп тастар едің?



Ойлануға берілген сұрақтар

1. «Шарт циклі» дегеніміз не?
2. Шарт циклі **while** операторының жұмыс негіздері қандай?
3. Шарт циклі қандай жағдайда тоқтайды?
4. Шарт цикліне қандай есептерді мысал ретінде келтіре аласыңың?



Тапсырма

*

1-есеп. N натурал саны берілген. Сол санның цифрларының көбейтіндісін табатын блок-схема құрастырып, Есептің программа кодын жаз.

Мысалы	Нәтиже
4582	320

**

2-есеп. N натурал саны берілген. Сол санның неше таңбалы сан екенін анықтайтын блок-схема құрып, программа кодын жаз.

Мысалы	Нәтиже
8361270	7

3-есеп. N натурал саны берілген. Сол санның цифрларының арасындағы ең үлкенін табатын блок-схема құр. Есептің программа кодын жаз.

Мысалы	Нәтиже
70856954	9

4-есеп. Берілген программа кодын талда. Программа жұмысын аяқтағанда экранға қандай нәтиже береді? Программа нәтижесіне қарап, есептің шартын анықта.

1) $x = 0$
 $x = \text{in}(\text{input}())$
while $x \neq 0:$
 $k = x \% 10$
 $\text{print}(k, \text{end}="")$
 $x //-= 10$

2) $x = 0$
 $x = \text{in}(\text{input}())$
while $x \neq 0:$
 $k = x \% 2$
 $\text{print}(k, \text{end}="")$
 $x //-= 2$



Үй тапсырмасы

N натурал саны берілген. Сол санда неше тақ цифр бар? Блок-схема құрып, программа кодын жаз.

Мысалы	Нәтиже
3857455	5

4.4

Continue циклін басқару



Python тілінде программалауда **continue** циклін басқару нұсқаулығын қалай тиімді қолдануға болады?



Ойлан

- Күнделікті өмірде жүріп жатқан белгілі бір циклдік процеске араласуға, оны басқаруға бола ма? Ойыңды ортаға сал. Мысалдар келтір.
- Программаның орындалуы кезінде кейбір цикл қадамдарын орындамай, келесі қадамға өту үшін не істеуге болады?



Жаңа білім

Программа орындалуы барысында цикл жұмысын басқарып отыруға, оның жұмысына араласуға тұра келеді. Программалау кезінде ағымдағы цикл қадамы (итерация) жұмысын шарт қою арқылы тоқтатып, бірден келесі цикл қадамына (итерация) өтуіңе болады. Бұлай істеуге **continue** нұсқаулығы көмектеседі.

Continue (жалғастыру) – белгілі бір қойылған шарттың көмегімен ағымдағы цикл итерациясын тоқтатып, келесі цикл итерациясына ауыстыратын цикл нұсқаулығы.

Continue нұсқаулығы циклдің басына бірден ауысады қамтамасыз етеді. Бұл нұсқаулық арқылы кей жағдайларда циклдерді кіріктіріп пайдаланудан сақтайтын. **Continue** нұсқаулығын циклдің екі түрі: **for** параметрлі циклі мен **while** шарт циклінде қолдануға болады (*1, 2-схемалар*).

```
for i in range(k, n, x):
    <1-оператор не өрнек>
    <2-оператор не өрнек>
    if < шарт >:
        continue
    ...
    < n – оператор не өрнек >
```

1-схема. for цикліне қолдану

```
while <шарт>:
    <1-оператор не өрнек>
    <2-оператор не өрнек>
    if < шарт >:
        continue
    ...
    < n – оператор не өрнек >
```

2-схема. while цикліне қолдану



циклді басқару – управление циклом – cycle control

циклдің басына бару – переход к началу цикла – go to the beginning of the cycle

Continue нұсқаулығының циклдің екі түрінде қалай жұмыс істейтінін түсіну үшін мысалдар қарастырайық.

1-мысал. N мен M бүтін сандар аралығындағы барлық тақ санды экранға шығаратын программаның кодын құрастыр.

Мысалдың шарты бойынша N мен M аралығында цикл жүргізіп, цикл параметрінің жүп немесе тақ екенін тексереміз. Цикл параметрі жүп болған жағдайда бірден цикл басына қайтып, кейінгі цикл қадамын орындауға өтеміз. Мысалдың программа коды *1-сүретте*, ал нәтижесі *2-сүретте* берілген.

Continue нұсқаулығын for параметрлі циклде қолдану



```
File Edit Format Run Options Window Help
N,M=map(int,input().split())
for i in range (N,M+1):
    if i % 2==0:
        continue
    print(i,end=' ')
Ln:6 Col:0
```

1-сүрет. Программа коды (for)



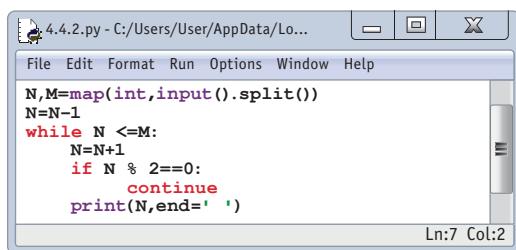
```
Python 3.7.4 (tags/v3.7.4:e09359112 t
(AMD64) on win32 Tipe "help" "copyright",
"credits" >>>
== RESTART: C:/Users/User/AppData\14 25
15 17 19 21 23 25
>>>
Ln:7 Col:0
```

2-сүрет. Программа нәтижесі

Continue нұсқаулығын while шарт циклінде қолдану

3-сүретте **continue** нұсқаулығын **while** цикліне қолдану программасы берілген. Екі циклде де қолдануда үлкен айырмашылық жок. Программа коды орындалғанда дәл *2-сүреттегі* нәтижесіні аламыз. Егер дәл осы типтес есептерді орындау керек болса, онда екі циклдің кез келгенін қолдануға болады.

Continue нұсқаулығын тиімді қолдану программада цикл құрылымын оқайлатып құруға көмектеседі. Бұл нұсқаулық кейбір есептердің программасын қарапайым етіп құруға ыңғайлы. Оны төмендегі мысалдан байқауға болады.



```
File Edit Format Run Options Window Help
N,M=map(int,input().split())
N=N-1
while N <=M:
    N=N+1
    if N % 2==0:
        continue
    print(N,end=' ')
Ln:7 Col:2
```

3-сүрет. Программа коды (while)

2-мысал. Латын әріптерінен құралған S жол берілген. Осы жолдағы барлық «а» әріптерін өшіріп, жолды экранға шығар.

Бұл есепте **for** параметрлі цикл көмегімен **continue** нұсқаулығын пайдаланып, одай программа құрымызызға болады. Есептің программа коды *4-сүретте*, нәтижесі *5-сүретте* берілген.

```

4.4.3.py - C:/Users/User/AppData/Lo...
File Edit Format Run Options Window Help
s=input('s=')
for i in S:
    if i=='A' or i=='a':
        continue
    print(i,end=' ')
Ln:5 Col:15

```

4-сүрет. Программа коды (2-мысал)

Мысалы	Нәтиже
Aduyaaaakeraasd	duykersd

```

Python 3.7.4 Shell
File Edit Format Run Options Window Help
Python 3.7.4 (tags/v3.7.4:e09359112 t (AMD
on win32 Tipe "help" "copyright", "credi
>>
== RESTART: C:\Users\User\AppData\Local\Temp\4.4.3.py
S=aduyaaaakeraasd
duykersd
>>

```

5-сүрет. Программа нәтижесі

for циклімен жұмыс істеу **while** цикліне қарағанда күрделірек болғанымен, **Python**-да бұл цикл – әмбебап. Циклді сандар тізбегі, жолдар мен тізімдер сияқты объектілердің кез келгеніне қолдануға болады. 2-мысалда біз **for** циклін жолдық объектілерге қолдану әдісімен таныстырық.



Ойлануға берілген сұрақтар

1. **Continue** нұсқаулығының қызметі қандай?
2. **Continue** нұсқаулығын программа құруда пайдалану қандай қажеттілікten туындаған?
3. **Continue** нұсқаулығын қолдануға мысалдар келтір.



Талдау

2-мысалдың *4-сүреттегі* программа кодына талдау жасаңдар. Осы мысалды **for** циклінің орнына **while** циклін қолданып шығаруға бола ма? Егер болса, *4-сүреттегі* программа кодында қандай өзгерістер жасау керек? Ойларыңды ортаға салыңдар. Себебін түсіндіріңдер.



Жинақтау

Өздеріңе таныс циклдік есептердің бірін таңдаңдар. **Continue** нұсқаулығын пайдаланып, шығару жолын ұсыныңдар.



Тапсырма

*

1-есеп. K мен P аралығындағы соңғы цифры t -ға тең сандарды экранға шығаратын программа кодын құр.

Мысалы	Нәтиже
10 39 7	17 27 37

**

2-есеп. N натурал саны берілген. Сол санда көздесетін барлық P цифрын өшіріп таста. Санды кері ретпен экранға шығаратын программа кодын құр.

Мысалы	Нәтиже
2547585 5	8742

**

3-есеп. Ағылшынның кіші әріптерінен тұратын S жол берілген. Осы жолдан дауысты дыбысты белгілейтін әріптерді өшіріп, дауыссыздарын экранға шығаратын программа кодын құр. Ағылшын тілінде алты дауысты дыбыс бар. Олар: a, e, i, o, u, y.

**

4-есеп. Наурыз айында Қазақстанның барлық өңірінде ауа райы құбылмалы келеді. Қоқтем шығып, күн жылынса да, кей күндері ауа температурасы нөлден төмендейді. Еліміз бойынша наурыз айының A мен B ($1 \leq A \leq B \leq 31$) күндері аралығындағы орташа ауа температурасы берілген. **Continue**-ді пайдаланып, температура нөлден жоғары болған күндердің санын анықтайтын программа құрастыр. Деректерді енгізу тігінен орындалады.

Мысалы	Нәтиже
10 19 -5 3 4 -2 3 -1 4 7 10 7	7



Үй тапсырмасы

N мен M бүтін сандары берілген. N -нен бастап әрбір 3-ші түрған санды қосатын программа кодын құрастыр.

Мысалы	Нәтиже	Түсініктеме
21 34	135	$21 + 24 + 27 + 30 + 33 = 135$

4.5

Break циклін басқару



Python тілінде программалауда **break** цикл басқару нұсқаулығын қандай жағдайда қолдануға болады?



Ойлан

- Құнделікті өмірде жүріп жатқан белгілі бір циклдік процесті бірден тоқтата салуға бола ма? Қандай мысал келтіре аласың?
- Программаның орындалуы кезінде тиісті нәтижеге жеткен-нен кейін цикл жұмысын тоқтату үшін не істеу керек?



Жаңа білім

Өткен тақырыпта программаны орындау барысында кейбір цикл қадамдарын орындаамай келесі қадамға өтүге болатынын білдік. Программалау кезінде цикл жұмысы мақсат етілген нәтижеге жеткен уақытта оның жұмысын тоқтатуға болады. Ол үшін **break** нұсқаулығын пайдаланамыз.

Break (үзілу) – белгілі бір қойылған шарт орындалуымен ағымдағы цикл жұмысын бірден тоқтатып, программа жұмысын циклден шығарып, әрі қарай жалғастыратын цикл нұсқаулығы.

Python-да **break** нұсқаулығы цикл жұмысына әсер ететін сыртқы фактор пайда болған кезде циклді бұзуға мүмкіндік береді. **Break** нұсқаулығын **for** параметрлі циклі мен **while** шарт циклінде қолдануға болады (*1, 2-схемалар*). Схемаларда көрсетілгендей, **break**-тің программада жазылуында **continue** нұсқау-

```
for i in range(k, n, x):
    <1-оператор не өрнек>
    <2-оператор не өрнек>
    if < шарт >:
        break
    ...
    < n-оператор не өрнек>
```

1-схема. for цикліне қолдану

```
while < шарт >:
    <1-оператор не өрнек>
    <2-оператор не өрнек>
    if < шарт >:
        break
    ...
    < n-оператор не өрнек>
```

2-схема. while цикліне қолдану



циклдің үзілі – прерывание цикла – **conditional break**
циклдің нұсқаулығы – инструкция цикла – **loop instruction**

лығынан айырмашылығы жоқ. Тек екеуінің атқаратын қызметі екі түрлі, бірі циклден шығаруды қамтамасыз етсе, екіншісі циклдің белгілі бір қадамдарын орындамай өтуге мүмкіндік береді.

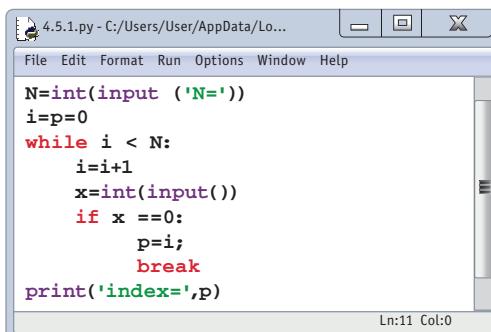
Break нұсқаулығы циклдің екі түрінде қалай жұмыс істейтінін түсіну үшін бір мысал қарастырайық.

Мысалы. Ұзындығы N ($1 \leq N \leq 100$) болған сандар тізбегі берілген. Тізбектегі сандардың арасында тек біреуі нөлге тең. Нөл саны орналасқан орынды табатын программа жаз.

Break пен **while** көмегімен берілген мысалдың программасы құрылған (*1-сүрет*). Программа коды орындалғанда *2-сүреттегі* нәтижені аламыз. Программа сандар тізбегіне нөлді енгізгенде, оның орнын сақтап, **break** циклі арқылы бірден тоқтатады (*2-сүрет*).

Мысалы	Нәтиже
$N=5$	
4	
8	
0	
7	
4	

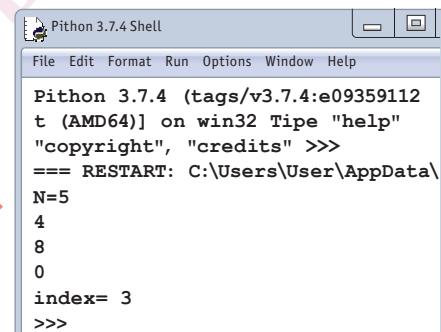
Break нұсқаулығын while шарт циклінде қолдану



```
N=int(input ('N='))
i=p=0
while i < N:
    i=i+1
    x=int(input())
    if x ==0:
        p=i;
        break
print('index=',p)
```



1-сүрет. Программа коды (while)

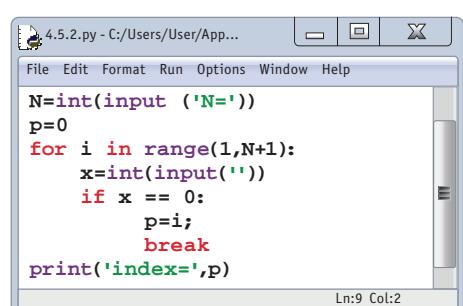


```
Pthon 3.7.4 Shell
File Edit Format Run Options Window Help
Pthon 3.7.4 (tags/v3.7.4:e09359112
t (AMD64) on win32 Tipe "help"
"copyright", "credits" >>>
== RESTART: C:\Users\User\AppData\Local\Temp\4.5.1.py
N=5
4
8
0
7
4
index= 3
>>>
```

2-сүрет. Программа нәтижесі

Break нұсқаулығын for параметрлі циклінде қолдану

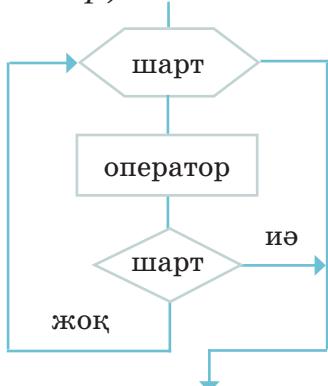
3-сүретте **break** нұсқаулығын **for** параметрлі циклінде қолдану программасы берілген. **For** циклінің программасы **while** цикліне **break**-ті қолданудан айырмашылығы өте аз екені көрініп тұр. Программа коды орындалғанда *2-сүреттегі* нәтижені аламыз.



```
N=int(input ('N='))
p=0
for i in range(1,N+1):
    x=int(input())
    if x == 0:
        p=i;
        break
print('index=',p)
```

3-сүрет. Программа коды (for)

Жоғарыда айтып өткендей, **break** нұсқаулығы программаны орындау барысында нақты ізделген нәтижеге жеткенде циклден шығу үшін қолданылады. Енді **break** пен **continue** нұсқаулықтарының графикалық блок-схемасы қалай өрнектелетінімен танысадайық (3, 4-схемалар).



3-схема. Break нұсқаулығының блок-схемасы



4-схема. Continue нұсқаулығының блок-схемасы



Талдау



Break пен **continue** нұсқаулықтарының графикалық блок-схемасына, қолданылған программа кодтарына талдау жасандар. Нұсқаулықтардың графикалық блок-схемаларын программамен салыстырындар. Блок-схемаларды тағы да басқаша қалай құруға болады?



Жинақтау



Жоғарыда берілген мысалдағы тапсырманың толық блок-схемасын құрып, ұсыныңдар.



Бағалау



Break пен **continue** нұсқаулықтары қызметіндегі негізгі үқсастық пен айырмашылықтарды салыстыра отырып, бағала. Бұл нұсқаулықтардың программа құруды жеңілдетуге қандай пайдасы бар? Екі нұсқаулықтың тек біреуін ғана қолданып нәтиже алуға бола ма? Себебін түсіндір.



Ойлануға берілген сұрақтар

1. **Break** нұсқаулығы цикл жұмысына қалай әсер етеді?
2. **for i in range(1,10):**

```
if i ==4 :
    break?
```

циклі неше рет жұмыс істейді?
3. **Break**-ті циклде шартсыз қолдансақ, қандай нәтиже аламыз?
4. Циклде бірнеше **break** нұсқаулығын қолдануға бола ма?

Себебін түсіндір.



Тапсырма

*

1-есеп. N натурал саны берілген. Сол санның цифрларының арасында нөл болса, цикл жұмысын тоқтатып, «Error» сөзін, әйтпесе санның өзін кері қарай экранға шыгаратын программа құр.

Мысалы	Нәтиже
570256	Error
4589	9854

**

2-есеп. Латын әріптерімен «,» үтірі бар S жол берілген. Осы жолдағы үтірге дейінгі әріптер мен олардың санын экранға шыгаратын алгоритмнің блок-схемасын және программасын құр.

Мысалы	Нәтиже
Alma, almurt	Alma 4

3-есеп. N натурал саны берілген. Сол санның цифрларының арасындағы ең кішісін табатын блок-схема құр. Есептің программа кодын жаз.

Мысалы	Нәтиже
78856954	4



Үй тапсырмасы

Төменде берілген 1-ші және 2-программа кодтарын талда. Программа жұмысын аяқтағанда экранға қандай нәтиже шығады?

1) **s = 0**

```
for i in range(11, 21):
    s = s + i % 10
    if s>15:
        break
print(s)
```

2) **x = 10; s = 0**

```
while x>5:
    x = x-2
    s = s+x
    if s>20:
        break
print(x)
```

4.6

Else циклін басқару



Python тілінде программалауда **else** цикл басқару нұсқаулығын қандай жағдайда қолдануға болады?



Ойлан

- Break пен continue цикл басқару нұсқаулықтарын программа жазуда қолдана білу қаншалықты маңызды?
- Шарт тексеру операторының **else** командасы қандай қызмет атқарады? Есіне түсір.



Жаңа білім

Откен тақырыптарда **break** пен **continue** цикл нұсқаулықтарын циклдің екі түрінде қолдану жолдарымен таныстық. Бұл нұсқаулықтарды циклдің ішінде шарт тексеру операторымен бірге қолдандық. Осындай нұсқаулықтың тағы бірі – **else** цикл басқару нұсқаулығы. Else нұсқаулығының алдыңғы екеуінен айырмашылығы – **for** және **while** циклдерінің сыртында орналасады. Else нұсқаулығы цикл барлық қадамды толық орындалады, қалыпты жағдайда өз жұмысын аяқтағанын немесе **break** арқылы бірден тоқтағанын көрсетеді. Else нұсқаулығы цикл тек **break**-ті пайдаланбай, өз жұмысын қалыпты аяқтағанда ғана орындалады.

```
for i in range (k, n, x):
    <1-оператор не өрнек>
    <2-оператор не өрнек>
    ...
    < n – оператор не өрнек>
else:
    < оператор не өрнек>
```

1-схема. for циклінің толық құрылымы

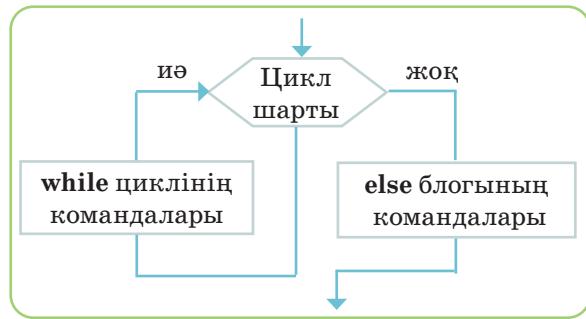
```
while < шарт >:
    <1-оператор не өрнек>
    <2-оператор не өрнек>
    ...
    < n – оператор не өрнек>
else:
    < оператор не өрнек>
```

2-схема. while циклінің толық құрылымы

1-схемада for циклінің толық құрылымы берілген. Егер цикл денесінде **break** нұсқаулығы болмаса, онда цикл өз жұмысында барлық қадамды толық аяқтайды. Содан кейін ғана **else** блогындағы оператор не өрнекті орындейды. Ал енді *2-схемадагы while* циклінің

эйтпесе – иначе – else
цикл денесі – тело цикла – loop body

толық құрылымына назар аударайық. Бұл цикл жұмысын цикл шартын тексеруден бастайды. Егер шарт «ақықат» болса, онда цикл жұмысын жалғастырады. Ал егер шарт «жалған» болса, онда **else** блогындағы командалар орындалып барып, цикл операторы жұмысын аяқтайды. З-схемада **while** циклінің алгоритмдік блок-схемасы берілген. **For** циклінің алгоритмдік блок-схемасы да осы құрылымға ұқсас болады. Циклдердің **else** блогымен бірге қолданған жағдайдағы жұмысын түсіну үшін практикалық тапсырмалар қарастырайық.



З-схема. **while** циклінің алгоритмдік блок-схемасы



Практикалық жұмыс

1-практикалық тапсырма. Ұзындығы $N(1 \leq N \leq 20)$ бүтін сандар тізбегі берілген. Осы тізбек тек оң сандардан тұрса, онда тізбектің қосындысын, әйтпесе «Tizbekte teris san bar» жауабы экранға шығатын программа құрастырыңыз.

1) Мысалы	Нәтиже	2) Мысалы	Нәтиже
$N=5$	$S=23$	$N=5$	Tizbekte teris san bar
2		2	
3		3	
5		5	
6		4	
7		-3	

Түсініктеме. Тапсырманың программасын құру үшін, **for** циклін **else** блогымен бірге қолданайыңыз. Тізбек мүшелерін енгізуді цикл деңесіне орналастырамыз. Тапсырманың программа коды 1-суретте берілген. Программа коды орындалғанда 2-суреттегі нәтижені аламыз.

```

4.6.2.py - C:/Users/User/AppData/Local/Pro...
File Edit Format Run Options Window Help
N=int(input ('N='))
s=0
for i in range(1,N+1):
    x=int(input())
    s+=x
    if x <=0:
        print("Tizbekte teris san bar")
        break
else:
    print('s=',s)
Ln:6 Col:10
  
```



1-сурет. Программа коды

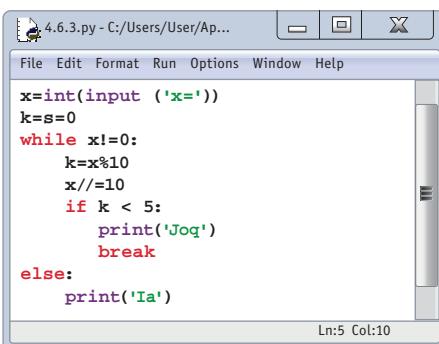
```

Python 3.7.4 Shell
File Edit Format Run Options Wind
Python 3.7.4 (tags/v3.7.4:e093591, May 27 2019, 16:57:46)
Type "help", "copyright", "credits" or "license" for more information
>>>
==== RESTART: C:/Users/User/AppData/Local/Programs/Python/Python37-32/4.6.2.py ====
N=5
2
3
5
6
7
S = 23
>>>
  
```

2-сурет. Программа нәтижесі

2-практикалық тапсырма. Санның цифрларының құрамында 5-тен кіші цифр болмаса, онда мұндай санды «мықты сан» деп атайды. X натураган саны берілген. Осы сан «мықты сан» болса, онда «Іа», әйтпесе «Joq» жауаптарын экранға шыгаратын программа құрастыр.

Түсініктеме: Тапсырманың программасын құру үшін, while циклін else блогымен бірге қолданамыз. Санның цифрларын қалдықты анықтау амалымен бөліп алып, 5-пен салыстырып шықсақ жеткілікті. Тапсырманың программа коды 3-суретте, нәтижесі 4-суретте берілген.



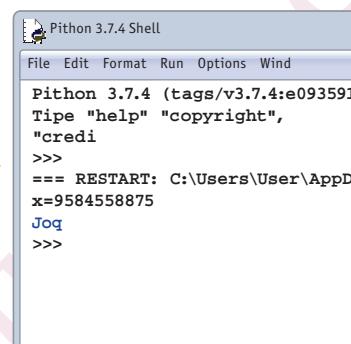
```

4.6.3.py - C:/Users/User/App...
File Edit Format Run Options Window Help
x=int(input ('x='))
k=s=0
while x!=0:
    k=x%10
    x//=10
    if k < 5:
        print('Joq')
        break
    else:
        print('Ia')
Ln:5 Col:10

```



3-сурет. Программа коды



```

Python 3.7.4 Shell
File Edit Format Run Options Wind
Python 3.7.4 (tags/v3.7.4:e093591, Dec 27 2019, 16:57:44)
[PyQt5: "help", "copyright", "credits"]
>>>
==== RESTART: C:\Users\User\AppData\Local\Temp\4.6.3.py ====
9584558875
Joq
>>>

```

4-сурет. Программа нәтижесі

Талдау

1, 2-практикалық тапсырмалардың программа кодын талданадар (1, 3-суреттер). Екі программадағы else блогының қызыметіне түсініктеме беріңдер.



Жинақтау

For циклінің алгоритмдік блок-схемасын құрып, оны ұсынышдар.



Бағалау

Break, continue, else цикл нұсқаулықтарының негізгі үқсастықтары мен айырмашылықтарын салыстыра отырып бағала. Бұл нұсқаулықтардың программа құрудағы маңызы қандай? Программа құруды қаншалықты жеңілдетеді? Ойынды ортаға сал.

Мысалы	Нәтиже
59867	Іа
977479	Joq



Тапсырма

*

1-есеп. N натурагал саны берілген. Сол санның цифрларының арасында «7» цифры болса, цикл жұмысын тоқтатып, «Error», әйтпесе санның цифрларының қосындысын экранға шығаратын программа құр.

Мысалы	Нәтиже
92576	Error
123456	21

**

2-есеп. Латын әріптерінен құралған L жол берілген. Осы жолда $'k'$ әрпі кездесе ме? Анықта. Егер іздеген әріп кездессе, онда «bar», ал кездеспесе, «joq» жауабын экранға шығаратын программа құр.

Мысалы	Нәтиже
uhuyhukgggff	bar

**

3-есеп. Ұзындығы $N(1 \leq N \leq 20)$ бүтін сандар мен бір ғана бөлшек саннан тұратын тізбек берілген. Бөлшек сан кей жағдайда тізбекте кездеспеуді де мүмкін. Осы тізбекте бөлшек болса, онда санның орнын, әйтпесе «joq» жауабын экранға шығаратын программа құр. Программада деректер тігінен енгізіледі.

Мысалы	Нәтиже
6 45 69 327 32.5 87 23	Index = 4



Үй тапсырмасы

Төменде берілген тақырып бойынша жоба жұмысын орында.

Жоба тақырыбы	Зерттеу сұрақтары мен бағыттары
Python программалау тілінде циклдерді басқару	<ul style="list-style-type: none"> Циклдің қызметі мен түрлері Циклдік алгоритмдермен жұмыс істеудің ерекшеліктері Циклдің басқару нұсқаулығын қолдану жолдары: <code>continue</code>, <code>break</code>, <code>else</code>

4.7

Алгоритмнің трассировкасы



Python тілінде программа жұмысын біртіндеп тексеріп, қатесін қалай жөндеуге болады?



Ойлан

- Программалау үшін алгоритмнің маңызы қандай?
- Программада жіберген қателерінді қалай анықтап, қалай жөндейсің? Тәжірибеңе сүйеніп өңгімелे.
- Программа жолдарын біртіндеп орындаудың қажеттілігі неде?



Жаңа білім

Программа құрушы программалау тілінің операторлары мен оларды қолдану ерекшеліктерін қаншалықты жақсы білсе, соншалықты қатесі аз болады. Дегенмен қанша жақсы біліп, тәжірибелі болса да, қате кетуі мүмкін.

Кез келген программалау тілінде жазылған программаны дер кезінде реттеп, **түзетуге** (отладка) болады. Программалау тілінде қателерді түзететін өз құралдары бар. Программалаушылар өз тілінде программада жіберілген қателерді **багтар** (ағыл. *bug* – қате), программа кодын түзетуді **дебаггер** (ағыл. *debugger* – түзетуші) деп атайды. **Python** тілінде де қатені жөндейп, программаны түзететін, реттейтін өз құралдары бар.

Программалаушылар программа жазуда 3 түрлі қате жіберуі мүмкін. Олар:

1. Синтаксистік;
2. Программаның орындау уақытына байланысты;
3. Алгоритмдік қателер.

Программадағы қателермен жұмыс істеу кезінде екі түрлі сұрақ туындаиды. Олар:

- Программада осы қателерді қалай анықтауға болады?
- Программада осы қателерді біртіндеп орындаі отырып, қалай жөндеуге болады?



алгоритмнің трассировкасы – трассировка алгоритма – tracing algorithm
түзету – отладка – debugging

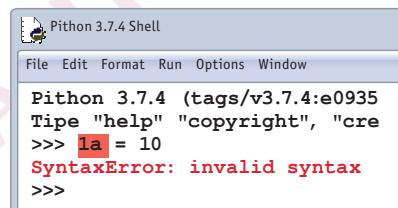
Программаның жұмыс атқарып тұрған уақыты, оның командалары компьютер процессорының көмегімен бірінен соң бірі ретімен орындалады. Программалаушы программаның орындалуы кезінде дәл қай уақытта қай команданың орындалып жатқанын анықтай алмайды. Программаның дұрыс жұмыс істемеу себебін білу үшін, командалардың орындалуының нақты тәртібін білу керек. Оны программада **трассировка** деп атайды.

Трассировка алгоритмі кездесетін қатемен жұмыс істеу ретін білдіреді. Программаны біртіндеп, қадамdap (step-by-step) орында процесі **трассировка** деп аталады.

Трассировка уақытында программалаушы команда беру арқылы программаның тізімде тұрған қатарларын орындаі алады.

Синтаксистік қате

1-суреттегі қате айнымалы саннан басталған. Мұндай қателерді программа интерпретация кезінде бірден көрсетеді. Егер қате дер кезінде жөнделмесе, онда программа жұмысын тоқтатады.



```
Python 3.7.4 Shell
File Edit Format Run Options Window
Python 3.7.4 (tags/v3.7.4-e0935
Type "help", "copyright", "cre
>>> la = 10
SyntaxError: invalid syntax
>>>
```

1-сурет. Синтаксистік қате

Программаның орындалу уақытына байланысты қате

Қателердің тағы бір түрі программаның орындалу уақытының өтіп кетуімен байланысты (run-time errors). Бұл программаны іске қосу немесе тестілеу кезінде пайда болады. Мысалы, бұл қатенің туындау себебі – программа компьютерде жоқ **input.txt** мәтіндік файлын оқуға әрекет жасайды. Бұл файлды таппаған соң, «қате» деп көрсетеді. Программада қате болса, хабарлама беріледі де, тоқтайады.

Алгоритмдік қателер

Тестілеу – программаның жұмыс қабилетін тексеруге арналған, арнайы дайындалған деректерді компьютерге енгізу. Егер программа тестілеу кезінде қарастырылған бір жағдайды дұрыс орындалап, басқасын қате орындаса, онда бұл – есептің алгоритмін құрудан кеткен қате. Мысалы, «+» орнына «-» таңбасын немесе «*» белгісін шатастырып қою. Қатенің бұл түрі интерпретация кезінде көрінбейді және программа дұрыс іске қосылады. Бірақ нәтижедегі жауап қате болады. Ол үшін программа кодын қайта қарап, жөндеу керек. Міне, осы процесс **түзету жасау кезі (отладка)** деп аталады. Қарапайым мысалды қарастырайық.

Екітаңбалы сан берілген. Сол санның цифрларының қосындысы жұп немесе тақ екенін анықтайтын жоба жасайық.

Программа коды интерпретациялағып, жұмыс істеп тұр. Бірақ бұл программа коды арқылы тек «жұп» санды ғана анықтауға болады. Программа кодында «тақ» сан болу жағдайы қарастырылмаған. Міне, алгоритмдік қате деген осы (*2-сүрет*). Программаны қайта қарап, шарт орындалмаған жағдайға «else: ('taq')» жолын жазсақ, программа коды жөнделіп, «тақ сан» берілгенде де нәтижені дұрыс табады. Сонымен бірге программалау барысында жиі кездесетін бірнеше түрлі қате бар. Соларға тоқталып өтейік (*1-кесте*).

```

4.8.1.py - C:/Users/User/AppData...
File Edit Format Run Options Window Help
N=int(input ())
if ((N//10)+(N %10)) %2==0:
    print('jup')

```

Ln:1 Col:0

2-сүрет. Алгоритмдік қате

1-кесте

№	Программа коды	Қате түрі	Программа орындалғанда экранға шығатын хабарлама
1	s = input() if s % 2==0: print('Jup')	Шаманың типіне сәйкес емес амал қолдану	TypeError: not all arguments converted during string formatting
2	k=int(input()) p=0;t=k/p print(t)	Санды нөлге бөлуға болмайды. Айнымалыны нөлге бөлу	ZeroDivisionError: division by zero
3	a = 5 print (a + b)	b шамасын таныстырмая	NameError: name 'b' is not defined
4	int("Hi")	Жолды санға айналдыру	ValueError: invalid literal for int() with base 10: 'Hi'
5	k=int(input()) t=input() print(k+t)	Бүтін сан мен жолды қосу. Мысалы: 45+ '85'	TypeError: unsupported operand type(s) for +: 'int' and 'str'

Программаның қатесін жөндеу үшін, трассировкалау кезінде `quit()` командасын пайдаланған жөн. Цикл деңесінің жұмысын бақылау үшін шарт тексеру операторымен бірге қолдану жақсы нәтиже береді.

Берілген n санының арасынан жұп сандардың санын табатын программа құр. Программаны трассировкалау арқылы нәтижесін анықта. Мысалдың программа коды 3-суретте, трассировка 2-кестеде берілген.

```
n = int(input())
s = 0
for i in range(0, n):
    t = int(input())
    if t % 2 == 0:
        s += 1
print(s)
```

3-сурет. Мысалдың программа коды

2-кесте. Программаны трассировкалау

Айнымалылар		Цикл шарты	Айнымалы	Шарт тексеру	Айнымалы
n	i	i < n	t	t % 2 == 0	s
3	0	0 < 3 (иә)	5	5 % 2 == 0 (жок)	0
	1	1 < 3 (иә)	16	16 % 2 == 0 (иә)	1
	2	2 < 3 (иә)	4	4 % 2 == 0 (иә)	2
				Нәтиже	2



Талдау



Алдыңғы программа құрудағы тәжірибелеріңе сүйеніп, өздеңдіңде программа құрастырғанда жіберген қателерді топтаңдар. Оны 1-кестедегі қате түрлерімен салыстыра отырып, талдаңдар.

Программа трассировкалау қалай орындалады? Ол үшін төмендегі мысалды қарастырайық.



Ойлануға берілген сұрақтар

Мысал келтір.

1. Программалау барысында өз тәжірибенде кездескен қандай қателерді айта аласың?
2. «Алгоритмді трассировкалау» дегеніміз не?
3. Программа трассировкасы қалай жүзеге асырылады?
4. Программа құру барысында жиі кездесетін қателердің түрі қандай?
5. Программаны трассировкалау неліктен өте маңызды?



Тапсырма

*

1-тапсырма. k мен p аралығындағы сандардың цифрларының қосындысын табатын программа құр. Тапсырманы орындау барысында қандай қате жибердің? Назар аудар. Цикл денесінде орналасқан операторлар жұмысын трассировкалау арқылы тексер.

Мысалы	Нәтиже
97 103	61

**

2-тапсырма. Төменде берілген программа кодтарында кеткен қателерді анықта.

1)

```
a = input('a=')
s = 0
for i in range(1, 10):
    s = s + i
    a= a * i
print(s/a)
```

2)

```
p = int(input ('p='))
s = 0; i = 0
while i < p:
    i = i + 1
    s = s + i
t = s * k
print(t)
```

3-тапсырма. $N(100 \leq N \leq 999)$ санына дейінгі үштаңбалы сандар арасындағы барлық цифры әртүрлі сандарды экранға шығар.

Мысалы	Нәтиже
107	102 103 104 105 106 107

Ескерту: Үштаңбалы сандар 100-ден басталады.



Үй тапсырмасы

for параметрлі циклі мен **while** шарт цикліндегі тақырыптардың бірін таңдал, есеп құрастыр. Компьютерде құрастырыған есебіңің программасын жаз. Жазған программаңың трассировкасын жаса.

4.8-4.9

Программауды үйренейік. Практикалық тапсырмалар



Python-да есептердің программасын құруда циклдерді қалай тиімді қолдануға болады?



Ойлан

- **while** цикл операторын қолданудың қандай ерекшеліктері бар?
- **for** цикл операторын пайдалануда қандай қателер жиі кездеседі?
- **Continue, break, else** циклдерін басқару нұсқаулықтарын қолдану программа құрудың тиімділігін қаншалықты арттырады?
- Программаны біртіндеп орында отырып, тексерудің қандай тиімді жақтары барын анықта.



Практикалық жұмыс

Программауда жолдарын түсінуге арналған үлгі тапсырмалар

**

1-тапсырма. Тіктөртбұрышты кесте. Тәменде көрсетілген өлшемі $N \cdot N$ тіктөртбұрышты кестені экранға шығаратын программа құр N ($1 < N < 20$). Тапсырманы орындаудың үлгісі *1-сүретте* берілген. Программаның шығару жолын талдаң, блок-схемасын құр.

Мысалы	Нәтиже			
4	1	2	3	4
	5	6	7	8
	9	10	11	12
	13	14	15	16

```

4.8.1.py - C:/Users/User/App...
File Edit Format Run Options Window Help
N=int(input())
p=0
for i in range(1,N+1):
    for i in range(1,N+1):
        p+=1
        print(p,end="\t")
    print()
Ln:6 Col:22

```

1-сүрет. Программа коды



палиндром сан – палиндром числа – number palindrome
программа әзірлеу – разработка программы – program development
сандар тізбегі – последовательность чисел – sequence of numbers
әртүрлі цифрлары бар үштаңбалы сан – трехзначное число с разными цифрами – three-digit number with different digits



2-тапсырма. Палиндром сандар

A-дан *B*-ға ($1 \leq A, B \leq 10^{18}$) дейінгі палиндром сандарды тауып, экранға шығаратын программа құрастыр. Егер сол аралықта палиндром сан болмаса, онда экранға «-1»-ді шығар.

Солға да, онға да бірдей оқылатын сан **палиндром** деп аталауды. Тапсырманы орындаудың үлгісі *2-суретте* берілген. Программаның шығару жолын талда.

№	Мысалы	Нәтиже
1	1800 2020	1881 1991 2002
2	1800 1880	-1



3-тапсырма. Цифрлары әртүрлі үштаңбалы сандар

Цифрлары әртүрлі үштаңбалы сандардың жалпы санын тап (кірістірілген циклдерді қолдан). Тапсырманы орындаудың үлгісі *3-суретте* берілген. Программаның шығару жолын талда.

```

4.8.2.py - C:/Users/User/App...
File Edit Format Run Options Window Help
A, B=map(int,input().split())
t=0
for i in range(A,B+1):
    n=i; s=0
    while n >0:
        k=n%10; n//=10
        s=s*10+k
    if s==i:
        t+=1; print(s, end=' ')
    else:
        if t==0:
            print(-1)

```

2-сурет. 2-тапсырманың
программа коды

```

4.8.2.py - C:/Users/User/AppData/L...
File Edit Format Run Options Window Help
p=0
for i in range(1,10):
    for j in range(0,10):
        for k in range(0,10):
            if i!=j and j!=k and i!=k:
                p+=1
print(p)

```

3-сурет. 3-тапсырманың
программа коды

Жоғарыда шешімі берілген тапсырмалардың үлгі программаларын талдай отырып, төмендегі тапсырмаларды орында.



Тапсырма



1-тапсырма. Бірдей сандар

K-дан *N*-ге дейінгі екітаңбалы сандар арасында екі цифры да тақ сандарды экранға шығар.
($10 \leq K, N \leq 99$)

Мысалы	Нәтиже
15 32	15 17 19 31

**

2-тапсырма. Барлығы әртүрлі

M -нен N -ге дейінгі төрттаңбалы сандар арасындағы ($1000 \leq M, N \leq 9999$) цифrlары әртүрлі сандардың нешеу екенін экранға шығар. Егер сол аралықта ондай сан болмаса, онда экранға «-1»-ді шығар.

№	Мысалы	Нәтиже
1	1245 1256	9
2	1221 1229	-1

**

3-тапсырма. Ең үлкен аудан

N тіктөртбұрыш қабырғаларының ұзындығы a ($1 \leq a \leq 20000$) және ені b ($1 \leq b \leq 20000$), N ($1 \leq N \leq 20$). Осы тіктөртбұрыштардың арасындағы ауданы ең үлкенінің мәні мен орнын экранға бер.

Мысалы	Нәтиже
3	30
4 3	2
6 5	
9 2	

**

4-тапсырма.

Тіктөртбұрышты кесте

Төменде көрсетілген өлшемі $N \cdot N$ тіктөртбұрышты кестені құр. N ($1 < N < 20$)

Мысалы	Нәтиже
4	0 1 1 1
	1 0 1 1
	1 1 0 1
	1 1 1 0

**

5-тапсырма.

Үшбұрышты кесте

Төменде көрсетілген өдіспен сандарды экранға шығар. N ($1 < N < 20$)

Мысалы	Нәтиже
5	2 3 4 5
	3 4 5
	4 5
	5

**

6-тапсырма. 4 цифры

K -дан N -ге дейінгі сандардың цифrlарының арасында 4 цифрының жалпы кездесу санын тап. Егер 4 цифры кездеспесе, онда «-1» жауабын экранға шығар. Есептің блок-схемасын құрып, программа кодын жаз.

№	Мысалы	Нәтиже
1	4444 4450	21
2	1055 1063	-1

Ескерту. Кірістірілген циклге else нұсқаулығын қолдан.

7-тапсырма. 2 мен 5 цифры

N ($1 \leq N \leq 10^9$) санының арасынан кез келген ретпен орналасқан, тек қана 2 мен 5 цифрынан тұратын сан құр. Егер бұл цифrlар кездеспесе, онда «0» жауабын экранға шығар. Есептің блок-схемасын құрып, программа кодын жаз.

Ескерту. Continue нұсқаулығын қолдан.

№	Мысалы	Нәтиже
1	54522158	52255
2	14789631	0

8-тапсырма. Ортақ сандар

N ($1 \leq N \leq 10^9$) және M ($1 \leq N \leq 10^9$) сандарының екеуінде де кездесетін цифrlарды тап. Егер екеуінде де кездесетін ортақ цифр болмаса, экранға «-1»-ді шығар. Есептің программа кодын жаз.

Мысалы	Нәтиже
125360	
267894	62

Ескерту: кірістірілген циклді қолдан.

9-тапсырма. Палиндром

N ($1 \leq N \leq 10^{18}$) саны палиндром ба? Анықта. Программасын құр. Егер сан палиндром болса, экранға «Ia», әйтпесе «Joq» жауаптарын шығар.

№	Мысалы	Нәтиже
1	2332	Ia
2	12324	Joq

10-тапсырма. Сандар тізбегі

K -дан N -ге дейінгі сандар ($1 < K$, $N < 10^9$) берілген. Осы сандардың арасында цифrlарының қосындысы жұп болған сандарды экранға шығар.

Мысалы	Нәтиже
30 40	31 33 35 37 39 40

Тапсырманы орындау алгоритмі:

- тапсырманы мүқият оқып, шартын түсіну;
- тапсырманың математикалық моделін жасау;
- алгоритмдік блок-схема құру;
- программа құру мен трассировкалау.

4.10

Программалау бойынша жоба жұмыстарын орындау



Python программалау тілі бойынша жиналған теориялық және практикалық білімді жобаларды орындауда қалай тиімді пайдалануға болады?



Ойлан

- Жобалау жұмыстарын орындау ережелері қандай? Есіңе түсір.
- Программалау арқылы күнделікті өмірде өз қажетімізге жарайтын қандай жобалар жасауға болады?



Тапсырма

Төменде берілген жоба тақырыптарының арасынан бір тақырыпты таңдап алып, жеке немесе жұппен орында. Жобаны орындау барысында теориялық материалдарды ұсынумен бірге жобаның программалық өнімдерін жасау да өте маңызды.



1-жоба тақырыбы

Python программалау тілінде программалаудың маңызы

Қысқаша мазмұны. Python программалау тілінің пайда болу тарихына, оның басқа программалау тілдерінен ерекшелігіне тоқталу керек. Python программалау тілінің мүмкіндіктеріне, бүгінгі таңдағы нұсқалары мен оны қолданудың маңызына назар аудару керек.

2-жоба тақырыбы

Python программалау тіліндегі циклдердің түрі мен оларды программалау

Қысқаша мазмұны. Циклдік алгоритмдерге, олардың программалаудағы ерекшелігіне тоқталу керек. Басқа программалау тілдерінде циклдер қалай қолданылады? Олардың бір-бірінен айырмашылығы қандай?

3-жоба тақырыбы

Python программалау тілінде қарапайым тест программасын жасау

Қысқаша мазмұны. Python-да тест жасау жолдарын зерттеу. Жасаған тест үлгілерінің ерекшеліктерін түсіндіру. Артықшылықтары мен кемшіліктеріне тоқтала отырып, программа өнімін ұсыну. Таныстырылым жасау.

4-жоба тақырыбы

Программалау көмегімен қолданбалы есептерді шешу

Қысқаша мазмұны. Қолданбалы математика, яғни математиканы өмірде қолдану жағдайларына байланысты есептерді программалау әдісі арқылы шешу жолдарын зерттеу. Дайындаған шағын программалық өнімдерін ұсыну. Таныстырылым жасау.

5-жоба тақырыбы

Программалау тілдері көмегімен электрондық оқулықтар жасау

Қысқаша мазмұны. Бұгінгі таңда электрондық оқулықтарды жасайтын программалар өте көп. Программалау тілдері көмегімен оқулықтарды жасау әдістерін теориялық түрғыдан зерттеу.

6-жоба тақырыбы

Python тілінде «Физикадан» есептер мен лабораториялық жұмыстарды программалау

Қысқаша мазмұны. Мектепте «Физика» пәнінен 7, 8-сыныптарда оқыған тақырыптар бойынша есептердің формуласын қолданып, берілген физикалық шамаларды автоматтарты түрде есептейтін программа жасап, ұсыну. Таныстырылым жасау.

7-жоба тақырыбы

Геометриялық есептерді программалау

Қысқаша мазмұны. «Геометрия» пәні бойынша өткен тақырыптарынды, аудан, периметр, фигураның қолемін табуға берілген есептерді программалау жолдарын зерттеу. Геометриялық фигура-лардың әртүрлі параметрлерін программалау арқылы шешу жолдарын зерттеу. Дайындаған шағын программалық өнімдерін ұсыну. Таныстырылым жасау.

Жобаны қорғау кезінде:

- таңдаған тақырыбынды тиянақты түрде таныстыруға;
- тақырыпты не себепті таңдағаныңды түсіндіруге;
- мақсат пен міндеттерінді айқындалап беруге;
- пайдаланған құралдарынды нақтылап атап шығуға;
- тақырып бойынша қысқаша шолу жасауға;
- жобаны орындауда пайда болған ойынды ортаға салуға;
- жоба барысында үйренген жаңа білім мен дағды жайлышты өндірілеуге;
- жоба қорытындысымен таныстыруға;
- тыңдаушылардың пікірі мен бағасын тыңдалап, түсіністікпен қабылдай білуғе дайын бол.

Жобаны бағалау барысында:

- жобаны өз бетінше орындауын;
- тақырыптың өзектілігі;
- тақырыпты аша білуін;
- мәселені ұсынудың өзіндік жолы;
- қорғаудағы әртістік қабілетің мен өзінді үстай білуін;
- техникалық құралдар мен көрнекіліктерді пайдалана білуін;
- қойылған сұрақтарға дұрыс өрі нақты жауап берес алудың бағаланады.

Жоба соңында есебін жаз.

- Мен неге бұл тақырыпты таңдадым?
- Жаңадан не білдім, не үйрендім?
- Керекті ақпаратты қалай таптым?
- Деректерді қалай өндедім?
- Теория мен практиканы қалай ұштастырудым?
- Қай нәрсе ұтымды шықты?
- Нені орындаі алмадым?
- Не оңай болды?
- Не қызын болды?
- Жобаны қайта орындауда, тағы да қалай толықтырадар едім?
- Жобаны орындаудан алған әсерім қандай?

5.1 Мәселе қою (қалыптастыру)



Python тілінде есептердің моделдерін құру кезінде мәселені қою қалай жүзеге асады?



Ойлан

- Циклдер жайлышы не білесің? Бұл үғымның біздің өмірімізбен қандай байланысы бар?
- Табиғатта, адам өмірінде циклдердің маңызы қандай? Мысал келтір.
- Программалау кезінде мәселенің қойылу кезеңін дұрыс анықтау мен тұжырымдау қаншалықты маңызды? Есіңе түсір.



Жаңа білім

7-сыныпта есептерді шешу кезеңдерімен танысқансың. Мәселені қою кезеңі осы кезеңдердің арасындағы ең маңыздысы, яғни есепті шешудің іргетасы деп санауға болады. Есептің берілгені қате тұжырымдалса, онда кейінгі қадамдар қалай орындалғанына қарамастан, есеп қате шығатыны анық.

Берілген есепті **Python** программалау тілінде орындау кезеңдерімен төменгі сыныпта толық танысқанбыз. Есептің программасын орындау кезеңдері практикада қалай жүзеге асатынын көрейік. Ол үшін есепті шешу кезеңдерін практикалық түрғыда пайдаланмыз. **Python**-да «Ағашты тең бөліктеге бөлу» есебінің жобасын қарастырайық. Жоба деп қарау себебіміз, бұл тапсырманың программа коды басқаша құрылса да, ол дұрыс нәтиже беруі мүмкін. Сол үшін бұл есепті толық шешу алгоритмін авторлық жоба ретінде қарастырамыз. Жобаны толық орындау, программалауды жүзеге асыру кезеңдері практикалық жұмыс ретінде берілген. Осы жобаны орындау кезеңін мүқият талдау, компьютерде орындасаң, программа құрудың математикалық түрлерін байланысты екенін түсінуге болады.



жобаның математикалық моделі – математическая модель проекта –
mathematical model of the project
деректерді тестілеу – тестирование данных – data testing



Практикалық жұмыс

«Ағашты тең бөліктеге бөлу» жобасы

Ұзындығы N ($2 \leq N \leq 100$) метрге тең ағашты M ($2 \leq M < 20$) бірдей тең бөліктеге бөлуге бола ма? Анықта. Егер тең бөліктеге бөлінбесе, айырмашылығы 1-ден артық болмайтын бөліктеге бөл.



№	Мысалы	Нәтиже
1	24 6	4 4 4 4 4 4
2	13 4	3 3 3 4
3	37 8	4 4 4 5 5 5 5 5

1. Мәселенің қойылуы

Берілген есептің шартын дұрыс түсіну керек. Есеп бойынша екі шарт туындайды:

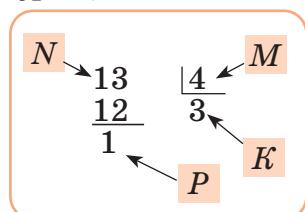
1-шарт. Ұзындығы N метр ағашты M бірдей бөлікке бөлу.

2-шарт. Бірдей бөлікке бөлу мүмкін болмаса, бөліктедің ұзындық айырмашылығы 1-ден артық болмайтын бүтін бөліктеге бөлу. Есепте басқа жағдай туындаиды, ейткені ағашты бөлікке бөлуде міндетті түрде осы екі шарттың бірі орындалады.

2. Жобаның математикалық моделін құрастыру

Берілген мысалдың 1-тестінен көріп түргандай, N саны M -ге қалдықсыз бөлінсө ғана ағашты тең бөліктеге бөлуге болады. 1-шарттың орындалуы үшін $N \% M = 0$ шарты орындалуы керек. Егер бұл шарт орындалмаса, онда есептің нәтижесі 2-шарт бойынша алынады. Берілген тестілерге қарасақ, есептің орындалуында өзіндік зандаудың бар екені байқалады. Мысалы, 2-тест жағдайын қарастырайық. 13 санын 4-ке бөлгенде бөлінді 3-ке, қалдық 1-ге тең болды. Есептің жауабы ретінде K -ны ($K = 3$) $M - P$ рет, яғни 3 рет және $K + 1$ -ді P рет жауап ретінде шығару керек (1-сурет). Бұл зандаудың басқа да жағдайларда тексеруге болады.

Мысалы, 37 және 8 болған жағдайды қарастырайық. 37-ні 8-ге бөлгенде бөлінді 4, қалдық 5-ке тең. Бөліндін 8-5 рет жауапқа шығарамыз. 4 4 4 тізбегі пайда болады. Бұдан кейін бөліндігे 1-ді қосқанда $4 + 1$, қалдық сан қаншаға тең болса, сонша рет жауапқа шығарамыз. 5 5 5 5 тізбегі пайда болады.



1-сурет. Жобаның математикалық моделі

Есептің 1-шарты бойынша ағашты тең бөлікке бөлу мүмкін болмағанмен, 2-шарты бойынша бөліктегінде айырмашылығы 1-ден артық емес бүтін бөліктеге бөлуге болады.

3. Алгоритмін құрастыру. Есептегі негізгі заңдылықты тауып, математикалық моделін жасап алдық. Бұл есептің программасын жасау үшін оны жүйелеп, қадамдарға бөлуіміз қажет. Әрбір қадамда орындалатын командалар тізімін жоспарлаймыз.

Есептің алгоритмі:

1. Енгізу мен нәтижені шығару операторларын таңдау. Біздің жағдайда **input** пен **print** операторларын пайдаланамыз.
2. Есептегі басты шартты $N \% M = 0$ тексеру үшін, шарт тексеру операторын қолданамыз.
3. Сандар тізбегін шығару үшін, параметрлі цикл операторын пайдаланамыз.

4. Алгоритмді программа тіліне аудару

Жоспарланған алгоритм бойынша жобаның программасын құрайық (*2-сурет*). Программаны құру барысында жобаның программасында қолданылған әрбір оператордың бір-бірімен үйлесімді жұмыс істеуін үнемі назарда ұстау керек.

```
5.1.1.py - C:/Users/User/AppData/Local/Programs/Python...  
File Edit Format Run Options Window Help  
n,m=map(int,input().split())  
t=n // m;  
if n % m==0:  
    print(s, end='Agash ten bolikke bolinedi')  
    for i in range (1, (m-(n%m)+1)):  
        print(t, end=' ')  
else:  
    print('Agash ten bolikke bolinbeysi')  
    for i in range (1, (m-(n%m)+1)):  
        print(t, end=' ')  
    for i in range (1, (n%m)+1):  
        print(t+1, end=' ')  
Ln:12 Col:28
```

2-сурет. Программа коды

5. Деректерді енгізу мен тестілеу

Жобаның программасы дайын болғанда оны іске қосамыз. Жоба іске қосылғанда *2-суреттегі* жоба пішіні пайда болады. Деректерді енгізу барысында назар аударатын жайт – есептің шартында берілген шектеулерден тыс деректер енгізбеу керек. Егер программаға 0 және 2 немесе 10 және 16 деректерін енгізсек, программа қате жауап береді. Себебі бірінші жағдайда ағаштың ұзындығы 0-ге тең болуы мүмкін емес. Мұндай ағаш бөліктеге бөлінбейді. Ал екінші жағдайда есеп шарты бойынша ағашты бүтін бөліктеге бөлу шарты орындалмайды.

6. Есептің нәтижесін алу әрі талдау

Тестілеу нәтижесінде программа алдын ала дайындалған тестілерге дұрыс жауап бергенде ғана есепті 100 пайыз орындалды деп айтуда болады. Егер тестілеу нәтижесінде алынған нәтижелер есептің шартына сәйкес келмесе, онда программа құрылымында кет-

кен логикалық, алгоритмдік қателерді алдыңғы тақырыптарда көрсетілген трассировкалау әдісі арқылы қайта түзетуге тура келеді. *2-суретте* көріп тұрғандай, З-мысалда берілген деректер программаға енгізілгенде, программа дұрыс жауап беріп тұр. Тек бұл деректерге ғана емес, есепке арнап басқа тестілеу деректерін құрып, программа жұмысын тексеріп көруге болады.



Талдау



Математика, физика, информатика бойынша есептер шешу кезеңдерін салыстырыңдар. Ұқсастығы мен айырмашылықтарын анықтаңдар.



Жинақтау



Математикадан өздеріңе таныс есептердің бірін таңдаңдар. Таңдағандарың бойынша есеп шешу кезеңдерін пайдаланып, программа жобасын құрастырып, оны ұсыныңдар. Жобаға атау беріңдер.



Бағалау



Компьютерде есеп шешу кезеңдерін сипаттаудың бұдан басқа қандай түрлері бар? Зертте. Біз танысып шыққан түрі мен басқа түрлері арасында қандай айырмашылықтар бар? Анықта.



Ойлануға берілген сұрақтар

1. Программалау ортасында берілген есепті орындау қандай кезеңдерден тұрады?
2. Жобаның компонентін таңдауда қандай мәселелерге назар аудару керек?
3. Есептің тестін құрастыруда және тестілеу барысында қандай жағдайларды назарда ұстау керек?
4. Есептерді компьютерде шешудің қандай өзіндік ерекшеліктерін атай аласың? Мысал келтір.



Тапсырма

Python программалау тілінде есепті орындау кезеңдерін пайдалана отырып, тәмендеғі тапсырмалардың программасын құр.

* Қарапайым есеп

A, B сандары мен олармен орындалуы керек амалдардың таңбасы берілген. Мысалы: $+, -, /, *$. Берілген амалдар түріне қарап, *A, B* сандарымен осы амалдардың бірін орындаитын программа құр.

Мысалы	Нәтиже
7 9 +	16

** Оң сандардың арифметикалық ортасы

N ($0 < N \leq 100$) нақты сандар тізбегі берілген. Осы тізбектегі оң сандардың арифметикалық ортасын тап. Келесі жолда абсолюттік мәні 100-ден аспайтын *N* нақты сан берілген. Оң сандардың арифметикалық ортасын жүзден бір дәлдікпен экранға шығар. Егер оң сандар болмаса, «*Joq*» хабарламасын көрсет.

Мысалы	Нәтиже
4 3 -9 1 2.0	2.00
5 -1 -7 -19 -20 -40.0	Joq

** Ағаш отырғызу

Еліміздегі экологиялық ахуалды жақсарту, жастардың бойында патриотизмді қалыптастыру және тұған жерге деген сүйіспеншілікті ояту мақсатында, республикада «Жасыл Ел» бағдарламасы жұмыс істейді. «Тұған жерге ағаш сыйла» республикалық акциясы кезінде Астана қаласындағы мектеп



Мысалы	Нәтиже
$N = 5$	
200	4
157	
362	
965	
145	

оқушылары ата-аналарымен және қоғам белсенділерімен бірге әр көктемде ағаш отырғызады. Биыл да 8-сынып оқушылары N топқа бөлініп, ағаш отырғызу жұмыстарына белсенді кірісті ($1 \leq N \leq 20$). Әр топтың отырғызған ағаштарының саны белгілі деп есептеп, ағаш отырғызудан жеңімпаз топтың нөмірін анықтайтын программа жаз.

**

Баскетбол ойыны

Қазақстан Республикасында денешынықтыру мен спортты дамытудың мемлекеттік бағдарламасын іске асыру шеңберінде, еліміздің қалаларында мектеп оқушылары арасында баскетболдан турнирлер үнемі өткізіледі. Баскетбол ойыны 4 кезеңнен тұрады. Енгізу деректері ретінде әр кезеңде командалар жинаған ұпай санын көрсететін сан мен b бүтін саны берілген ($0 \leq a, b \leq 100$). Екі команда арасындағы ойынның нәтижесіне қарай жеңімпазды анықтайтын программа құр. Ойын нәтижесі ретінде «1-komanda», «2-komanda» немесе «ten oyun» сөздері жауапқа беріледі.



Мысалы	Нәтиже
26 17	1-komanda
13 15	
19 11	
14 16	

Ізделінген цифрдың санын анықтау
Бірінші жолда N ($0 < N \leq 10^{18}$) бір бүтін сан берілген. N саны тек одан болады. Екінші жолда K цифры берілген. N санындағы K цифрының санын табатын программа құрастыр.

**

Нәтижені анықта

Келесі программа коды орындалғаннан кейінгі нәтижесін анықта.

```
p=1; i=3
while i <= 9:
    p+=(i / 2)
    i = i + 1
print(p)
```

```
S=0; i=7
while i > 1:
    S+=i % 2
    i-= 1
print(S)
```

```
S=100; K=0
for i in range (1,6):
    K+=i*i
    S-=K
print(p)
```

5.2

Алгоритмді өзірлеу

 Python-да циклдік алгоритмдерді пайдаланып, есептердің қарапайым алгоритмдерін қалай құруға болады?

 Ойлан

- «Цикл» тақырыбындағы есептердің алгоритмдерін құрудың қандай қыншылықтары бар?
- Циклдік алгоритмдердің көмегімен құрылған қандай танымал алгоритмдерді білесің?

 Жаңа білім

Біз 7-сыныпта тармақталу алгоритмдерін өзірлеу жолдарымен танысқанбыз. Бұл тақырыпта біз циклдік алгоритмдер мен шығарылатын есептерді программалау кезінде оның алгоритмдерін дайындау мәселелерін қарастырамыз. Сызықтық және тармақталу алгоритмдерімен салыстырғанда, циклдік алгоритм біршама күрделі. Есептің программасын құру үшін есептің шартына сай, мәселенің қойылуын анықтап алушмен өткен тақырыпта танысқан болатынбыз.

Ендігі кезеңде мәселенің қойылуындағы шарттармен нақты жұмыс істей кезеңі басталады. Жалпы алғанда есептің алгоритмін өзірлеу дегенді біз тек оның графикалық блок-схемасын құру деп туғынбейіміз керек. Әңгіме сол блок-схеманың ішіне не жазатынымызда. Бұл кезеңде есептің шарты бойынша программаны құру қанша, қандай қадамдардан (алгоритм) тұруы керек деген мәселе туындейды. Сондай-ақ есептің неше қадамнан кейін, алгоритмнің қай түрін қолданып мақсатқа жететінімізді нақты жоспарлай алуымыз керек. Сонымен бірге бұл кезеңде есептің математикалық моделі маңызды рөл атқарады.

Есепті программалауда айналысу үшін «Математика» пәнінде алған білімдерінді қолдана білу өте маңызды. Кез келген есептің математикалық моделін құрып, оны математикалық тілге ауыстыра



алгоритм құру – создание алгоритма – algorithm creation
тиімді алгоритм – эффективный алгоритм – efficient algorithm

алсақ, онда оның программасын дұрыс құратынымыз анық. Сол үшін программалауда жиі қолданылатын алгоритмдерді түсініп, жатқа білу программалау кезінде үлкен көмегін тигізеді.

Жиі қолданылатын дайын алгоритмдерді білу неге маңызды?

Программа жазу процесінде берілген есептерді бірнеше кіші тапсырмаларға бөліп алып қарастырудың қолайлы екендігін жоғарыда айтып өттік. Программалау барысында берілетін тапсырмалардың басым бөлігі тиімділікті (оптимизация) анықтауға арналған есептер түрінде ұсынылады. Мұндай есептерде міндетті түрде ең үлкен, ең кіші, ең ауыр, ең жеңілін т.б. анықтау керек. Міне, осындағы жағдайларда алдын ала тексерілген алгоритмдерді қолданған тиімді. Ол біріншіден, уақытты ұнемдеуге, екіншіден, есептегі басты мәселені шешуге мүмкіндік береді.

Программалаушыларға таныс стандартты алгоритмдердің кейбіреулеріне тоқталайық. Бұл алгоритмдердің қызметін біз өткен тақырыптарда пайдаландық. Мұндай алгоритмдер қандай жағдайда қажет?

Алгоритмнің ең маңызды аспектілерінің бірі – алгоритм жылдамдығы. Программалау процесінде программалаушы алдында ұнемі программаның жұмыс уақытын ұнемдеу мәселесі тұрады. Мысалы, «Кірістірілген циклдер» тақырыбында сандар тізбегіндегі жай сандарды анықтап, экранға шығару тапсырmasын орындау. Жай сандарды анықтау мақсатында **for** циклін тексерілетін санының квадрат түбіріне дейін орындау. Неге? Себебіне тоқталайық. Мысалы, тексерілетін сан ретінде 50-ді алайық. Егер осы санының жай немесе құрама сан екенін анықтауымыз керек болса, бұл санды 1 мен 50 арасындағы барлық санға бөлуіміз керек. Онда 50 санын тексеру үшін цикл 50 рет жұмыс істеуі керек. Егер есептің алгоритмін осылай құрсақ, онда 1 000 000-нан үлкен бір санды тексеру үшін цикл денесі миллион рет тексерілуі қажет. Циклдің миллион рет айналып жұмыс істеуі үшін шамамен 1 секунд уақыт керек екенін ескерсек, өрине, бұл өте көп.

for j in range(1, n):

Демек, құрастырған алгоритм тиімсіз. Басқа тиімді алгоритм ұсынуымыз керек. Егер осы есепте 50 санының 1 мен 50-ге міндетті түрде бөлінетінін және кез келген санының бөлгіштері санының өзін есептемегендеге сол санының жартысына дейін кездесетінін ескерген жағдайда (мысалы 50-дің бөлгіштері 50-ді есептемегендеге 25-ке дейін кездеседі), санды 50-ге дейін бөлудің қажеттілігі қалмайды. Бұл жағдай-

да цикл жұмысы 2 есе қысқарып, 50 санын жай санға тексеру үшін 25 рет жұмыс істейді.

for j in range(1, n//2+1):

Егер сандар теориясын пайдалансақ, цикл жұмысын бұдан да азайтуға болады екен.

Теорема. Егер p жай саны a құрама санының ең кіші бөлгіші болса, онда $P \leq \sqrt{a}$ шартты орындалады.

Мысалы, 59 саны – жай сан, өйткені ол $\sqrt{59}$ -дан ($\sqrt{59} \approx 7,6$) кіші 2, 3, 5, 7 жай сандарының ешқайсысына бөлінбейді. Олай болса, 50 санын тексеру үшін де 2...7 арасындағы циклдің 6 рет жұмыс істеуі жеткілікті ($\sqrt{50} \approx 7$). Бұл бастапқы жағдайдан 8 есе аз. Ал тексерілетін сан 1000000 болса, цикл 1000 рет жұмыс істейді.

for j in range(2, round(math.sqrt(x))):

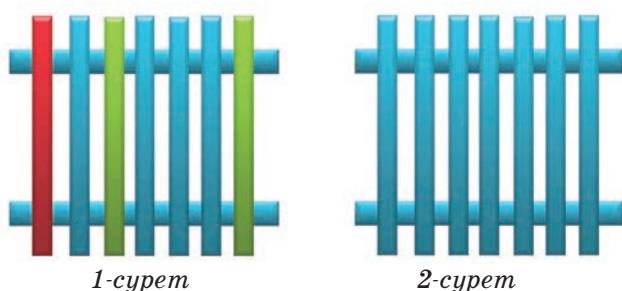
Бір мысалға талдау жүргізе отырып, программа жазу барысында тиімді алгоритмдерді білудің қаншалықты маңызды екенін көріп шықтық. Алгоритмдерді практикалық түрғыдан қолдануды үйрену үшін, «Шарбақты бояу» есебін компьютерде орында.



Практикалық жұмыс

Шарбақты бояу. Қанат үйінің айналасындағы шарбақты бояуға 5 түрлі бояу алды. Ол шарбақты қандай түске бояу керектігін анықтау үшін, шарбаққа әр түсті бояудан жағып шықты (*1-сүрет*). Нәтижесінде шарбақ түрлі түсті болды. Қанат шарбақтағы әртүрлі түстерді мүмкіндігінше аз күш жұмсап, бірдей түске бояғысы келеді. Қанатқа көмектес. Оның шарбағындағы тақтайшалардың саны $N(N < 100)$ және 1...5 арасындағы 5 түстің біріне боялған шарбақ тақтайшаларының түстері берілген. Есептің жауабы ретінде ең аз боялатын шарбақтар саны шығуы керек.

№	Мысалы	Нәтиже
1	$N=7$	3
	3	
	1	
	5	
	1	
	1	
	1	
	5	



Тағы бір назар аударатын мәселе – есептің мәтіні. Көптеген жағдайда программалауға берілген есептер өмірде кездесетін оқиғалар желісінен алынады. Ең күрделісі – осындай есептердің мәтінін түсініп, оны программалау.

Мысалы, жоғарыда берілген есепті математикалық тілде өрнектесек, қарапайым ғана бір қатар мәтіннен құралған болар еді.

«Тізбектегі ең көп қайталанатын сан мен қайталану санын тап».

Бұл жерде 1 ... 5 сандарын енгізе отырып, сол сандардың әрбірінің неше реттен кездескенін санаймыз. Солардың арасынан ең көп қайталанғанын тауып, енгізілген сандардың жалпы санынан алып тастаймыз.



Талдау



«Шарбақты бояу» есебінің программасына (*3-сурет*) талдау жасаңдар. Талдау барысында түрлі жағдайларды қамтитын тест тапсырмаларын дайындал, программа жұмысын жан-жакты тексеріндер.



Жинақтау



Төменде берілген есептің мәтінін күнделікті өмірмен байланысты есеп мәтініне аудыстырындар. Үлгі ретінде практикалық жұмыстағы есепті алындар.

Есеп. N мүшесі бар тізбек берілген. Осы тізбекте P -шы орынға дейінгі тізбек мүшелерінің қосындысын табатын программа құрындар ($1 \leq P, N < 100$).



Үй тапсырмасы

N мүшесі бар тізбек берілген. Осы тізбек мүшелерінің K -ға қалдықсыз бөлінетіндерінің қосындысын тап: ($1 \leq K, N < 100$).

```
File Edit Format Run Options Window Help
z=0; z1=0; z2=0; z3=0; z4=0
n=int(input())
for i in range(1,n+1):
    x=int(input())
    if x==1:
        z+=1
    elif x==2:
        z1+=1
    elif x==3:
        z2+=1
    elif x==4:
        z3+=1
    elif x==5:
        z4+=1
max=z
if z1>max:
    max=z1
if z2>max:
    max=z2
if z3>max:
    max=z3
if z4>max:
    max=z4
print(n-max)
```

3-сурет. Программа коды

5.3-5.4

Алгоритмді программалау. Практикалық жұмыс

 Python-да циклдік алгоритмдерді пайдаланып, есептердің программасын қалай құруға болады?

 Ойлан

- Циклдік алгоритмнің басқа алгоритм түрлерінен қандай айырмашылығы бар?
- Циклдік алгоритмдерді программалауда қандай қызыншылықтар туындауы мүмкін?

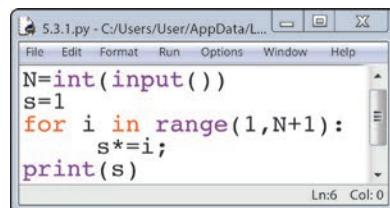
 Жаңа білім

Есептерді программалау барысында олардың алгоритмдік блок-схемасын жасау жолдарымен тәменгі сыныптарда танысқанбыз. Енді цикл тақырыбына берілген есептердің блок-схемасын құрастырып, оны программа тіліне ауыстыруды практикалық түрғыдан үйренуге болады. Ол үшін бірнеше есептің блок-схемасын құру және шығару жолымен танысайық.

1-есеп. N ($1 \leq N \leq 20$) факториалды есептеу блок-схемасын құрып, оны программаға айналдыр.

Есептің математикалық моделі. 1 мен N арасындағы сандардың көбейтіндісі N санының факториалы деп аталады. Факториал белгісі ретінде «!» символы қолданылады. Мысалы, $5! = 1 \cdot 2 \cdot 3 \cdot 4 \cdot 5 = 120$. Басқаша айтқанда, 1-ден N -ге дейінгі сандардың көбейтіндісін табу керек.

Есептің блок-схемасы (1-схема) берілген. Осы алгоритмді программалау тіліне ауыстырамыз (1-сурет).

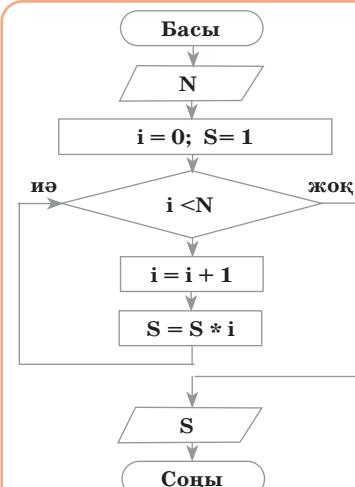


```

5.3.1.py - C:/Users/User/AppData/L...
File Edit Format Run Options Window Help
N=int(input())
S=1
for i in range(1,N+1):
    S*=i;
print(S)
Ln:6 Col: 0
  
```



1-сурет.
Программа коды



1-схема. Блок-схема

2-есеп. (n ($1 \leq n \leq 10^{12}$) саны берілген. Осы санның ең үлкен цифры мен ең кіші цифрын экранға шығар. Есептің блок-схемасын құрып (2-схема), оны программа кодына ауыстырыр (2-сурет).

Мысалы	Нәтиже
85486569	max=9 min=4

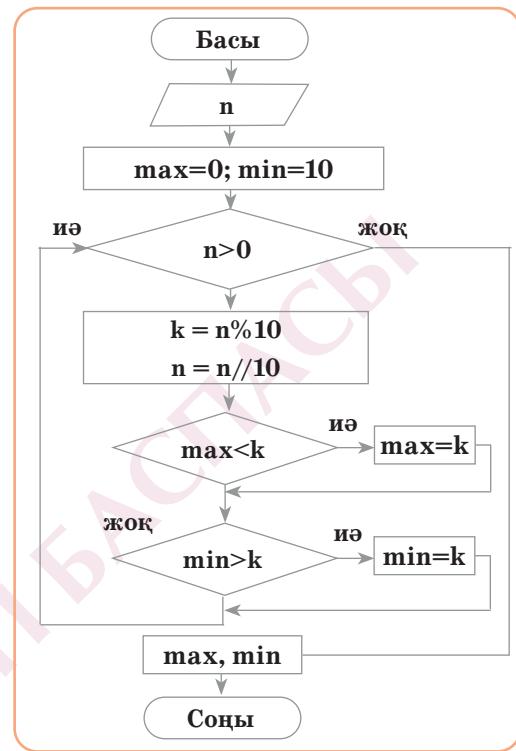
```

5.3.2.py - C:/Users/...
File Edit Format Run Options
Window Help
n=int(input('n='))
s=1; max=0; min=10;
while n>0:
    k=n%10
    n=n//10
    if k>max:
        max=k
    if k<min:
        min=k
print('max=',max)
print('min=',min)
Ln:11 Col:17

```



2-сурет. 2-есептің программасы



2-схема. 2-есептің блок-схемасы

Есептің математикалық моделі. Бұл есептің математикалық моделі оңай, тек н санының цифрларын ажыратып алғып, үлкен мен кішіні анықтау алгоритміне тексерсек жеткілікті. Есепті шешу үшін шарт тексеру циклін қолданамыз.



Талдау



2-есептің программалық кодын талдандар.

Программаға:

1) $n = 495\ 582$

2) $n = 385\ 845$ мәндері енгізілгенде қандай нәтижелер алынады?

Алынған нәтижелерді талдаап, түсініктеме беріңдер.



есептің шарты – условия задачи – conditions of the problem
программа кодын тексеру – проверка кода программы – checking program code

блок-схемалар – блок-схемы – flowchart

есептеу – вычисление – calculating



Тапсырма

* **1-тапсырма.** Төменде берілген программалардың блок-схемаларын құр.

1)

```
x = int(input())
a,b = 0, 0
while x>0:
    a = a + 1
    b = b + x%10
    x = x//100
print(a)
print(b)
```

2)

```
s = 0
for k in range(1, 51):
    if k %5=0:
        s = 2 * k + s
print(s)
```

* **2-тапсырма.** Төменде берілген программа кодтары жұмыс істегендегі, экранда қандай нәтиже аламыз? Программа кодын компьютерге салып тексермей, жауабын тап.

1)

```
s = 0
n = 0
while s < 51:
    s = s + 11
    n = n * 2
print(n)
```

2)

```
s = 175
n = 0
while s + n < 300:
    s = s - 5
    n = n + 25
print(n)
```

* **3-тапсырма.** Төменде берілген программа кодтарын жазуда қателер кеткен. Сол қателерді анықта.

1)

```
s = input()
k = 7
while s < 30
    s = s + 2 * k
    k:=k + 2
print(k)
```

2)

```
s = 0
for k in range(1, 50):
    if k %7 = 0:
        s = s + k
print('2 * s')
```

**

4-тапсырма. Жұп цифрлар қосындысы

Ұзындығы N -ге тең сандар тізбегі берілген ($1 \leq N \leq 1\ 000$). Тізбектегі жұп сандардың соңғы цифрларының қосындысын тап. Тізбек мүшелері – 10^9 -дан үлкен емес натурал сандар. Деректерді енгізу тігінен орындалады.

Мысалы	Нәтиже	Түсініктеме
5 425 4 658 12 35	14	$4 + 8 + 2 = 14$

**

5-тапсырма. Санның бөлшегін ажырату

Ұзындығы N -ге тең нақты сандар тізбегі берілген ($1 \leq N \leq 1000$). Осы сандардың тек бөлшегін 0,001 дәлдікпен жауапқа шығар. Тізбек мүшелері – 10^5 -тен артық емес нақты сандар. Төменде берілген программалардың блок-схемасын құр. Деректерді енгізу тігінен орындалады.

Мысалы	Нәтиже
5 4.12312 3 45.01 2.1 7.00005	0.123 0.000 0.010 0.100 0.000

**

6-тапсырма. Іздеу

Х саны берілген ($1 \leq X \leq 10000$). Осы санға ең жақын жай санды тап. Егер берілген санның өзі жай сан болса, өзін жауапқа шығар. Егер ондай сан екеу болса, кішісін шығар.

№	Мысалы	Нәтиже	Түсініктеме
1	31	31	2-тест: 15-ке жақын жай сандар – 13 және 17.
2	15	13	Есеп шарты бойынша жауап 13 болады.

**

7-тапсырма. Тізбек мүшесін тап

Фибоначчи тізбегі дегеніміз – өрбір келесі мүшесі алдыңғы екі мүшесінің қосындысына тең болатын $1, 2, 3, 5, 8,$

$13, \dots$ сандарының тізбегі. Фибоначчи

тізбегінің формуласы: $F_0 = 0, F_1 = 1, F_n = F_{n-1} + F_{n-2}, n \geq 0$. Осы тізбектің X -шы орнында қандай сан тұрғанын тап ($1 \leq X \leq 50$). Тізбек: $0, 1, 1, 2, 3, 5, 8, 13$.

№	Мысалы	Нәтиже
1	4	3
2	11	89

**

Үй тапсырмасы

N ($1 \leq N \leq 10^{12}$) саны берілген. Осы санда неше 7 цифры барын анықтайтын программа құр.

№	Мысалы	Нәтиже
1	14785747	3
2	786	1

5.5-5.6

Программаны тестілеу



Python-да программа жұмысының тиімділігін қалай тексеріп, қалай синақтан өткізуге болады?



Ойлан

- «Программаны тестілеу» дегеніміз не?
- Программаны тексеруге арналған тестілердің қандай түрлерімен таныссың?



Жаңа білім



7-сыныпта тармақталу алгоритмдеріне жазылған программаларды тестілеу әдістерімен танысқанбыз. Программаны тестілеу процесі программа алгоритмінің сзызықтық, тармақталу немесе циклдік түріне қарамайды. Тестілеу процесі барлық алгоритмде бірдей кезеңдерден тұрады.

Программаны тестілеу процесі үш кезеңге бөлінеді. Олар:

- Қалыпты жағдайларға тексеру.
- Төтенше (шегаралық) жағдайларға тексеру.
- Ерекше жағдайларға тексеру.

Берілген есептің программасын құрып болған соң, оны орындауға көшеміз. Программаны орындау барысында оны алдын ала дайындаған үлгі тестілер арқылы тексеру жолдарын 7-сыныпта үйрендік. Программаны тестілеу кезінде үлгі тестілердің толық өтпеу себебі программа құруда кеткен қателерден туындайды.

Программалауда кездесетін қателер

Компьютерде есеп шығарудың барлық кезеңінде, атап айтқанда, есептің қойылуынан бастап дайын жобаға айналғанға дейін мүқият орындалмаған жағдайда қателер кетеді. Есептің дұрыс қойылмауы, есепті шығару үшін құрылған алгоритмнің дұрыс болмауы, логи-

синтаксистік қателер – синтаксические ошибки – syntax errors

логикалық қателер – логические ошибки – logical errors

шектік(шегаралық) шарттарды тексеру – проверка граничных условий – check boundary conditions

ерекше жағдайлар үшін тексеру – проверка в исключительных условиях – check in exceptional conditions



калық қателер кететін жағдайларды дұрыс есепке алмау, семантикалық және синтаксистік қателер, операцияларды орындау кезінде кететін қателер, деректерді беруде жіберілетін қателер мен енгізу-шығару кезінде кететін қателер бұған мысал бола алады.

Әдетте синтаксистік қателер программаны орындау басталған кезде байқалады. Ал алгоритмдік қателер программалаушының түпкі ойын білмегендіктен, транслятор таба алмайды.

Сондықтан компьютердің синтаксистік қателер туралы хабарының болмауы программаның дұрыстығының жеткілікті шарты емес.

Синтаксистік қателердің мысалдары:

- тыныс белгілерінің қалып кетуі;
- жақшалардың сәйкес келмеуі;
- оператордың дұрыс жазылмауы;
- қызметші сөздің дұрыс жазылмауы;
- айнымалы атының дұрыс жазылмауы;
- циклдің аяқталу шартының болмауы;
- массив сипаттамасының болмауы т.с.с.

Транслятор таба алмайтын қателер

Программада пайдаланылатын операторлар дұрыс жазылғанымен, транслятор таба алмайтын көптеген қателер болады. Осындай қателерден мысалдар келтірейік.

Логикалық қателер:

- шартты тексеруді дұрыс көрсеткенімен, кейін алгоритмнің қай тармағын орындауға өту жолын дұрыс көрсетпеу;
- мүмкін болатын шарттарды толық есептемеу;
- программада алгоритм блоктарын орындауға қажет болатын бір немесе бірнеше шаманы қалдырып кету.

Циклдерде кездесетін қателер:

- циклдің басын дұрыс көрсетпеу;
- циклдің аяқталу шарттарын дұрыс көрсетпеу;
- итерация санын дұрыс көрсетпеу;
- шексіз цикл.

Енгізу-шығару қателері мен деректермен жұмыс істеудегі қателер:

- деректердің түрін дұрыс бермеу;
- талап етілген аз немесе көп деректерді оқуды үйымдастыру;
- деректерді дұрыс нәтижеге шығармау.

Айнымалыларды пайдалануда кететін қателер:

- айнымалыларды олардың бастапқы мәндерін көрсетпей, типін таныстырмай пайдалану;
- бір айнымалыны басқа айнымалының орнына қате көрсету.

Массивтермен жұмыс істеуде кететін қателер:

- алдын ала нөлге келтірілмеген массивтер;
- дұрыс сипатталмаған массивтер;
- индекстерінің берілу реттілігі қате массивтер.

Арифметикалық операцияларды орындауда кететін қателер:

- айнымалының түрін дұрыс көрсетпей (мысалы, нақты айнымалының орнына бүтін айнымалыны көрсету);
- әрекеттер ретін дұрыс анықтамау;
- нөлге бөлу;
- теріс санның квадрат түбірін табу;
- санның мәнді разрядын жоғалту.

Осы қателердің бөрін программаны тестілеу барысында анықтауға болады. Программаны тестілеу процесін практикалық түрғыдан орындаібыз.

Практикалық жұмыс

Программаны тестілеу процесімен жақын танысу үшін, «Белгіштер» есебін («Информатика» пәнінен аудандық олимпиада есебі) қарастырайыңыз.

Белгіштер

Берілген b натурал саны a санына қалдықсыз бөлінсе, онда a саны b санының белгіші болады. Мысалы, 15 санының 4 белгіші бар: 1, 3, 5, 15. Әрбір берілген b саны үшін оның белгіштер саны жұп немесе тақ сан болатынын анықта. N ($1 < N \leq 10^5$) және b саны берілген. Әрбір берілген b ($1 \leq b \leq 10^{18}$) үшін белгішінің саны жұп сан болса 0-ді, белгіш саны тақ сан болса, 1 санын шығар. Программаның әрбір тестіге жауап беру уақыты 2 секундтан аспайды.

Мысалы	Нәтиже	Түсініктеме
5 5 8 6 11 9	0 0 1 0 1	8-дің белгіштері 4-еу, яғни жұп сандар: 1, 2, 4, 8. 9-дың белгіштері 3-еу, яғни тақ сандар: 1, 3, 9, 16.

Есепке берілген үлгі тестілерге тоқталайық (*1-кесте*). Бұл тестілер есептерді шығарудың алдында дайындалып, ұсынылады. Есептің жалпы саны 10 тест болғанымен, біз негізгі 5 тестіге тоқталамыз. Себебі бірдей типтегі тестілер 2 реттен қайталанған.

1-кесте. Есепке берілген үлгі тестілер

№	Кіру тестілері	Шығу жауаптары
1	10 1 2 49 77 49 55 80 94 49 77	1 0 1 0 1 0 0 0 1 0
2	10 830 585 868 529 683 740 967 557 828 1147	0 0 0 1 0 0 0 0 0
3	10 501139 786769 976039 635209 902903 977269 557993 635209 654481 876831	0 1 0 1 0 0 0 1 1 0
4	10 5000207 1524071 50000214792889 50000348753931 50001431026237 50001060703878 50001009723014 50000483494043 50000950188029 50000119016331 50000702267865	0 1 0 0 0 0 0 0 0
5	10 500000001283061852 500000001394837861 499999937510787169 500000000548349143 500000000709382267 499999937510787169 500000002022608063 500000000107748979 500000000076465733 500000000770135069	0 0 1 0 0 1 0 0 0

1-кестеде берілген тестілерге тоқталайық.

1-тест. Қарапайым жағдай үшін.

2-тест. Сандардың ішінде жалғыз бөлгіші тақ болатын сан кездесетін тест.

3-тест. Шегі 10^6 болған жағдайлар үшін.

4, 5-тестілер. b мәні 10^{18} дәрежесіндегі сандарға тексеру тестілері.

Көріп тұргандай, қалыпты, төтенше және экстремалды жағдайларға тестілер берілген.

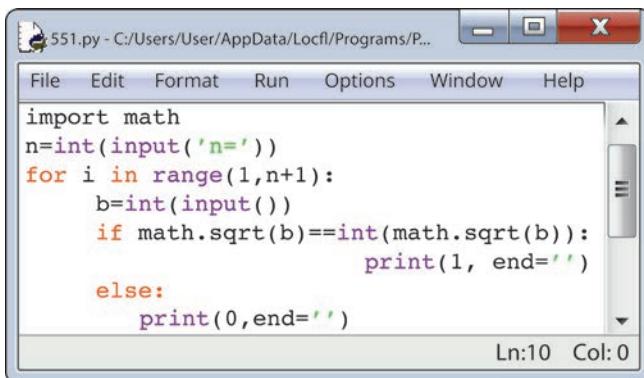
Есептің тестілері неге бұлай берілген? Есепті шешуде қандай қиыншылық кездеседі?

Есепті шешу барысында бір санның бөлгішін табу үшін бізге бір параметрлі цикл керек, ал сандар тізбегінің бөлгішін табу үшін бір цикл қосамыз. Сонда бұл есепті орындауда кірістірілген 2 циклді пайдалануымыз керек болады. Ал берілетін сандар тізбегінің ұзындығы 100 000 екенін ескерсек, онда программа әр тестіге 2 секундта жауап беруі мүмкін емес. Бұл жерде туындастын 2 мәселе: b санының шегі – 10^{18} . Біз санның бөлгішін анықтауда санды сол саның жартысына дейінгі сандарға бөліп шығамыз. Олай болса, біз бұл саның

жартысына дейін циклді айналдыруымыз тіпті де мүмкін емес. Өзіміз білетін алгоритммен шығару есепті 20 пайызға ғана орындауға мүмкіндік береді деген сөз. Демек, бұл есепті шешудің басқа тиімді алгоритмін қарастыруымыз қажет. Бұл есепке қолданатын тиімді алгоритм ретінде берілген саннан квадрат түбір алу қажет. Егер сан квадрат түбірден бүтін шығатын болса, онда бұл саның бөлгіштері – тақ сан, әйтпесе жұп сан болады.

Мысалы: 36 квадрат түбірден 6 шығады. Оның бөлгіштер саны – тоғыз. Олар: 1, 2, 3, 4, 6, 9, 12, 18, 36. Ал 40-тан шығатын түбір: 6, 324... Оның бөлгіштер саны – сегіз. Олар: 1, 2, 4, 5, 8, 10, 20, 40.

Онда есептің программасын (*1-сүрет*) қарапайым түрде ұсынуға болады.



```
551.py - C:/Users/User/AppData/Local/Programs/P...
File Edit Format Run Options Window Help
import math
n=int(input('n='))
for i in range(1,n+1):
    b=int(input())
    if math.sqrt(b)==int(math.sqrt(b)):
        print(1, end=' ')
    else:
        print(0,end=' ')
Ln:10 Col: 0
```



1-сүрет. «Бөлгіштер» есебінің программа коды

Талдау

Практикалық жұмыста берілген «Бөлгіштер» есебінің программа кодына арнап жасалған төмендегі З үлгі тестілерді талдаңдар. Төмендегі тестілер программага салғанда қандай жауаптарды экранға шығарады?

- | | | |
|---------------------------|-------------------------|------------------------|
| 1) 5
16 81 256 100 318 | 2) 5
11 8 26 101 910 | 3) 5
64 3 17 10 121 |
|---------------------------|-------------------------|------------------------|



Ойлануға берілген сұрақтар

1. Программаны тестілеу қандай кезеңдерден тұрады?
2. Тестілеу барысында қандай қате түрлерін анықтауға болады?
3. Программаны тестілеуге қандай тестілер жинағы беріледі?
4. Тестілеудің программа жұмысына қандай әсері бар?
5. Программаны тестілеу неліктен маңызды саналады?



Тапсырма

*

1-тапсырма. Берілген программа кодын (*2-сүрет*) компьютерге енгізбей-ақ үлгі тестілер бойынша тестілегенде қандай нәтижелер алғынады?

```
s = int(input())
n = 0
while 2 * s < 150:
    s = s + 5
    n = n + 2
print(n)
```

2-сүрет. Программа коды

№	Үлгі тест	Нәтиже
1	0	
2	5	
3	15	
4	7	
5	100	

**

2-тапсырма. 0 мен 1-ден тұратын тізбек берілген. Осы тізбектегі ең ұзын бірлер тізбегін тап. Тестілеу кезеңдерін пайдаланып, есептің дұрыстығын анықтайтын, 5 түрлі жағдайды қамтитын тест дайында.

Мысалы	Нәтиже
01001111011	4

**

3-тапсырма. Бөлшек дайындау

Жұмысшылар 5 сағатта N бөлшек дайындауы керек. Олар жұмысты бастаған алғашқы сағатта X бөлшек дайындайды, қалған бөлшектердің әрқайсысын дайындауға T минут уақыт жібереді.

№	Мысалы	Нәтиже
1	7 12 45	12
2	3 10 60	7

X, N және $T(X \leq N \leq 15, T \leq 240)$. Жұмысшылар 5 сағатта қанша бөлшек дайындал үлгереді? Анықта. Есепке түрлі жағдайды қамтитын 5 тест дайында. Программасын жаз.



Үй тапсырмасы

*

Кітаптың бағасы

Кітаптардың саны $N (N \leq 20)$ және бір кітаптың бағасы K берілген. Бір кітаптың бағасы K теңге, 2 кітаптың бағасы $2 * K$ теңге ... т.с.с. экранга шығар.

Мысалы	Нәтиже
3 500	1 кітап – 500 теңге 2 кітап – 1000 теңге 3 кітап – 1500 теңге

5.7

Олимпиадаға дайындық. Практикалық тапсырмалар



Python-да есептердің моделін құру, жобасын дайындау және программа кодын жазу



Ойлан

- Күнделікті өмірде кездесетін қандай процестерді программалау арқылы қызықты етіп орындауға болады?
- Пайдаланушының программа жазуда жіберетін негізгі қателері қандай?
- Есепті шешу жолының алгоритмін табуда қандай білімдерді үштастыра білу керек?



Практикалық жұмыс



1-тапсырма. Салыстыру

А мен В сандарының қайсысының цифрларының көбейтіндісі кіші?

Мысалы	Нәтиже
121 43	121



2-тапсырма. Нөлге тең

К мен Р сандарының арасында цифrlарының ең болмағанда біреуі нөлге тең неше сан барын анықта. ($1 \leq K \leq P \leq 1000$)

Мысалы	Нәтиже
8 22	2



3-тапсырма. Шарлар

Шарлар үшбұрыш тәрізді орналасқан: бірінші қатарда – 1, екінші қатарда – 2, үшінші қатарда – 3 т.с.с. Егер N шар берілген болса, осы шарлардың көмегімен қанша қатар құруға болады? *Kіру файлы:* бірінші қатарда бір ғана N натурал саны берілген ($1 < N < 999$). *Шығу файлы:* құруға болатын қатар саны түрады.

Input.txt	Output.txt
15	5



информатика пәнінен олимпиада есептері – задачи олимпиады по информатике – olympiad problems in computer science
нөлге тең – равно нулю – is equal to zero

Есепті шешуге берілген түсініктеме. Егер берілген шар саны қатарды толық құруға жетпесе, онда қатар толық құрылмаған деп есептейміз. Мысалы: егер 18 шар берілсе, онда есептің жауабы ретінде 5 шығуы керек, ейткені қатарды толық жабуға 3 шар жетпейді.

1
2 3
4 5 6
7 8 9 10
11 12 13 14 15
16 17 18 ...

* 4-тапсырма. Спортшы

Спортшы B күннен кейін өтетін жүгіру жарысына дайындықты 1-күні K метр қашықтыққа жүгіруден бастады. Ол әр күні жүгіру қашықтығын M пайызға артырып отырса, ең соңғы дайындық күні ол қандай қашықтыққа жүгіреді? Жауапты бүтін мәнге дейін жуықтап бер. Сандар B , K , M ретімен енгізіледі.

Мысалы	Нәтиже
11 1000 10	2593

**

5-тапсырма. Сағаттың соғуы

Ескі қабырға сағаттарының «тілі» нешені көрсетсе, сонша рет соғады. Мысалы, сағат 3 болса, 3 рет соғады. Ал сағат тілі 3 сағат 30 минутты (үш жарымды) көрсетсе, 1 рет соғады (*1-сурет*). Күндізгі сағат бірде және түнгі сағат бірде 1 рет, күн ортасында және түн ортасында 12 рет соғады. Бір тәулік ішіндегі берілген уақыт аралықтарына қарап, осы уақыт ішінде сағат барлығы неше рет соққанын анықтайтын программаның жобасын ұсын. Енгізу үшін a , b және c , d сандары берілген. a – бастапқы сағат, b – бастапқы минут, c – соңғы сағат, d – соңғы минут ($0 \leq a, c \leq 23$, $0 \leq b, d \leq 59$, $b \neq 30$, $d \neq 30$). Жоба үлгісі *1-суретте* берілген.



1-сурет. 5-тапсырмада сурет

Мысалы	Нәтиже
5 20 10 25	45
10 25 5 20	135

**

6-тапсырма. «Қошқар мен теке» ойыны

Қанат пен Самат өртүрлі логикалық ойындарды жақсы көреді. Бұл жолы таңдаған ойындары «Бұқалар мен сиырлар» деп аталса да, өздері жақсы көретін «Қошқар мен теке» мультфильм кейіпкерлерінің атын қойып алған (2-сурет).

Қанат өртүрлі цифрдан тұратын 4 таңбалы сан жасырады. Ал Самат сол санды Қанаттың бағыттап отыруымен табуы керек. Самат берген сандар нұсқасында Қанат қанша «қошқар», қанша «теке» болғанын айтады. «Қошқар» жасырылған санның цифрлары мен Самат ұсынған санның цифрлары бірдей, орындары сәйкес келетін цифрлар. Ал «текенін» цифрлары сәйкес, бірақ тұрған орындары өртүрлі.



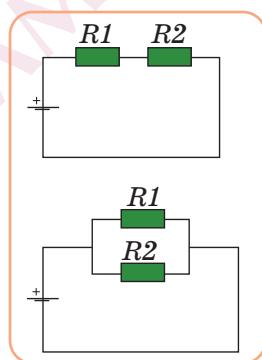
2-сурет. 6-тапсырмада сурет

Мысалы	Нәтиже
5671 7251	1 2
1425 1425	4 0

7-тапсырма. Откізгіштегі ток пен кедергіні анықтау

«Физика» пәнінің «Тұрақты ток» бөлімінде Ом заңын қолдану маңызды. 3-суретте осы тақырыпқа иллюстрация берілген. Тізбек бөлігі үшін Ом заңының формуласы бойынша ток күші, кернеу, кедергіні есептейтін, кедергілерді тізбектей өрі параллель (4-сурет) жалғай алтын, жалпы кедергіні табатын программа дайында. Тапсырманы орындау барысында тізбек бөлігі үшін Ом заңының формуласын пайдаланамыз.

Ом заңының формуласы: $I = \frac{U}{R}$



3-сурет. Тізбектей және параллель жалғау



4-сурет. 7-тапсырмада сурет

Сөздік – словарь – dictionary

қазақша	орысша	ағылшын тілінде
A		
абсолютті сілтеме	абсолютная ссылка	absolute reference
ақиқат	истина	true
ақпарат саны	количество информации	amount of information
ақпаратты өлшеу	измерение информации	measurement of information
ақпараттық қауіпсіздік	информационная безопасность	information security
алгоритмді құру	создание алгоритма	algorithm creation
алгоритмнің трассировкасы	трассировка алгоритма	tracing algorithm
алфавиттік әдіс	алфавитный метод	alphabetic method
алфавиттің құдаттылығы	мощность алфавита	power of the alphabet
аналитикалық	аналитический	analytical
аналық тақшা	материнская плата	motherboard
арифметикалық-логикалық құрылғы	арифметико-логическое устройство	arithmetic and logic unit
арнаның өткізу жылдамдығының қабілеті	пропускная способность передачи канала	capacity band width of channel's transmission
ассиметриялық цифрлық абоненттік желі	ассиметричная цифровая абонентская линия	asymmetric digital subscriber line
Ә		
while (әзірге)	Цикл while (пока)	цикл while while loop
әртүрлі цифрлары бар үштаңбалы сан	трехзначное число с разными цифрами	three – digit number with different digits
Б		
басқару құрылғысы	устройство управления	control device
байт	байт	byte
бейнекарта	видеокарта	graphics card
бит	бит	bit

білімнің белгісіздігі	неопределенность знаний	uncertainty of education knowledge
блок-схемалар	блок-схемы	flowchart
олжам, олжамдау	прогноз, прогнозирование	forecast, prediction
бұлттық есептеулер	облачные вычисления	cloud computing
бұлттық технология	облачная технология	cloud technology

Д

дербес компьютер	персональный компьютер	personal computer
деректерді талдау	анализ данных	data analysis
деректерді тестілеу	тестирование данных	data testing
дәреже	степень	degree
диаграмма	диаграмма	diagram
динамикалық қатар	динамический ряд	dynamic series
диапазон	диапазон	range
дыбыстық тақша	звуковая карта	sound card
деректің ақпараттылығы	информационность сообщения	information message report

Е

егер	если	if
екілік алфавит	двоичный алфавит	binary alphabet
екілік кодтау тілі	двоичный язык кодирования	binary coding language
енгізу-шығару порттары	порты ввода-вывода	input-output ports
ерекше жағдайларда тексеру	проверить в исключительных условиях	check in exceptional conditions
есептеу	вычисление	calculating
есептің шарты	условия задачи	conditions of the problem

Ж

жағымсыз аспект	негативный аспект	negative aspect
жалған	ложь	false
жедел жад	оперативная память	emergency memory
желідегі қауіпсіздік	сетевая безопасность	network security
желілік тақша	сетевая плата	network card

желінің өткізу қабілеті	пропускная способность сети	network's capacity bandwidth
және	и	and
жобаның математикалық моделі	математическая модель проекта	mathematical model of the project
жүйелік шина	системная шина	system bus
жұмыс немесе ақша табуды ұсыну	предложения работы или заработка	work or earnings offer
И		
инженерлік	инженерный	engineering
интернет	интернет	internet
интернетте ақша табу	заработка в интернете	making money online
интерфейс	интерфейс	interface
К		
квадраттың мәні	квадратное значение	square value
кибербуллинг	кибербуллинг	cyberbullying
кірістірілген функциялар	встроенные функции	bult – in functions
кірістірілген цикл	вложенный цикл	embedded loop
кірістіру	вставить	insert
коммутатор бауы	коммутационный шнур	switch cord
компьютерді пайдаланудың аспектілері	асpekты использования компьютера	aspects of using computeruse
компьютердің ішкі құрылымы	внутреннее устройство компьютера	computer structur's inner
компьютерлік желі	компьютерная сеть	computer network
корреляция	корреляция	correlation
көрнекі	визуальный	visual
кэш-жад	кэш-память	cache memory
О		
оптикалық дискжетектер	привод оптических дисков	odd optical disk drive

К		
қаржылық	финансовый	financial
қатыл диск	жесткий диск	hdd – hard disk drive
қоректендіру блогы	блок питания	power supply unit
құпиялышлық	конфиденциальность	confidentiality
қолжетімділік (оңтайшлылық)	доступность	availability
Л		
логарифмдік	логарифмический	logarithmic
логикалық қателер	логические ошибки	logical errors
логикалық	логический	logic
«Лотерея жеңімпазы» немесе «алыс тысқаннан қалған мұра»	«выигрыш в лотерее» или «наследство от дальнего родственника»	«winning the lottery» or «inheritance from a distant relative»
М		
математикалық өрнек	математическое выражение	mathematically expression
мерзім мен уақыт	дата и время	date and time
мәтіндік	текстовой	text
мәтіннің ақпараттық көлемі	объем информации в тексте	the amount of information in the text
П		
палиндром сан	числа палиндром	number palindrome
провайдер	провайдер	provider
программа әзірлеу	разработка программы	program development
программа кодын тексеру	проверка кода программы	checking program code
процессор, микропроцессор	процессор, микропроцессор	processor, microprocessor
процессордың жиілігі	частота процессора	processor frequency
процессордың құрылышы	устройство процессора	processor's device
процессордың разрядтылығы	разрядность процессора	processor's structure

Р		
регистор	регистор	register
С		
CD-DVD диск-жетектері	CD и DVD дисководы	CD and DVD drives
салыстырмалы сілтеме	относительная ссылка	relative reference
сандар тізбегі	последовательность чисел	sequence of numbers
сенімділік	надежность	reliability
символдың ақпараттық салмағы	вес символьной информации	weight of symbol information
синтаксистік қателер	синтаксические ошибки	syntax errors
сілтеме	ссылка	reference
сілтемелер мен жиындар	ссылки и массивы	references and arrays
«Сиқырлы өмірндар»	«волшебные кошельки»	«magic wallets»
слот кеңейтілімі	слот расширения	expansion slot
сопроцессор	сопроцессор	coprocessor
статика	статика	statics
статистикалық	статистический	statistical
сызықтық	линейный	linear
сызықтық функция графигі	график линейной функции	graph linear functions
Т		
тасымалдау уақыты	время передачи	delivery time
түзету	отладка	debugging
тұтастық	целостность	integrity
Ф		
функциялар	функции	functions
Х		
хабар	сообщение	message

Ц

цикл	цикл	cycle
циклдің денесі	тело цикла	loop body
циклдің нұсқаулығы	инструкция цикла	loop instruction
циклдің шарты	условие цикла	loop condition
циклдің басына бару	переход к началу цикла	go to the beginning of the cycle
циклді басқару	управление циклом	cycle control
циклді орындау	выполнение цикла	loop execution
циклдің үзілуі	прерывание цикла	conditional break

Ч

чип	чип	chip
-----	-----	------

III

қосымша құрылғылар	периферийные устройства	peripheras
шектік (шегаралық) шарттарды тексеру	проверить граничные условия	check boundary conditions
шина	шина	bus

Ы

ықтималдық әдісі	вероятностный метод	probabilistic method
------------------	---------------------	----------------------

Э

электромагниттік сәуле	электромагнитное излучение	electromagnetic radiation
электрондық кесте	электронная таблица	spreadsheet

Я

ядролардың саны	число ядер	number of cores
IP-адрес	IP-адрес	internet protocol address

ГЛОССАРИЙ

Абсолютті сілтеме – бастапқы деректер орналасқан, көшіру мен ауыстыру кезінде ұяшық адресінің формуласының өзгермеуі. Абсолютті адresteі көрсету үшін \$ белгісін енгізесің. Абсолютті сілтемелердің толық және *iшинара* деген екі түрі бар.

Абсолютті және салыстырмалы – философиялық категориялар. Абсолютті – шұбәсіз, тәуелсіз, өзгермейтін. Салыстырмалы (релятивті) – басқа құбылыстарға тәуелді, байланысты. Салыстырмалылық (релятивті) құбылысты басқа құбылыстармен қатысы арқылы сипаттайты.

Алфавит (гр. ἄλφαβητος – жазу нұсқасы) – берілген жиында өзгермейтін болып қабылданған, берілген тілде кез келген хабарды құрастыруға болтын символдардың (әріп, цифр және басқа шартты таңбалар) жиынтығы.

Алаяқтық – жымқыру, алдау немесе сенімді теріс пайдалану жолымен бөтен мүлікті иемдену.

Алгоритм – қазіргі математика мен электронды есептеуіш мәшинеде қолданылатын негізгі ұғымдардың бірі. Белгілі бір тендеу түбірінің жуық мәнін кез келген дәлдікпен табу оған арналған алгоритммен есептеледі. Компьютердің кең қолданылуына байланысты, алгоритм жаңа мағынаға ие болды. Берілген есепті шешу барысында орындаушыға біртіндеп қандай әрекеттер жасау керектігін түсінікті әрі дәл көрсететін нұсқау да *алгоритм* деп аталады. Алгоритмді орындаушы – адам, ЭЕМ немесе робот. Әрбір нұсқау – команда. Ал орындаушының жүзеге асыра алатын командалар жиыны *командалар жүйесі* деп аталады.

Аналық тақша (Материнская плата – *motherboard*) – орталық процесор, жедел жад микросхемалары, кеңейту-қуыс орындары мен компьютердің басқа да кейбір жүйелік сынарлары орналастырылған негізгі мөрлі тақта. Кез келген компьютердің негізгі құрамдас бөлігі.

Ақпаратты қорғау – ақпараттық қауіпсіздікті қамтамасыз етуге бағытталған шаралар кешені. Тәжірибе жүзінде ақпаратты қорғау деп деректерді енгізу, сақтау, өндеу мен тасымалдау үшін қолданылатын ақпарат пен қорлардың тұтастығын және құпиялығын қолдауды түсінеді. Сонымен, ақпаратты қорғау – ақпараттық сыртқа кетуінің, оны ұрлаудың, жоғалтудың, рұқсатсыз жоюдың, өзгертудің, маңызына тимей түрлендірудің, рұқсатсыз көшірмесін жасаудың, бұғаттаудың алдын алу үшін жүргізілетін шаралар кешені. Қауіпсіздікті қамтамасыз ету ұйымдастырушылық, программалық, техникалық әдістер мен құралдардан тұрады.

Аспект (лат. *aspectus* – көзқарас). Зерттеліп жатқан түсінікке, затқа немесе құбылысқа деген әрқылы көзқарас.

Болжам (немесе болжау):

1) ғылым методологиясында мәні толық анықталмаған заттар мен құбылыстардың негізінде жатқан ішкі байланыстар мен қатынастар туралы ұсынылатын ғылыми жорамал;

2) құқықтануда логикалық ой қорыту негізінде пайымдау.

Болжам таным әдісі ретінде ежелгі заманда-ақ (мысалы, ежелгі Грекияның философиясы мен математикасында) белгілі болған. Алайда оның ғылыми танымдағы эвристикалық рөлі XVII–XIX ғасырларда эксперименттік математиканың, жаратылыстанудың айрықша жетістіктеріне (Ньютон механикасының дамуы, физика мен химияда атомдық-молекуларлық көзқарастың қалыптасуы, энергияның сақталу және айналу заңының ашылуы т.б.) орай нақтылана бастады.

Буль (логикалық) функциясы (Булева (логическая) функция; *boolean function*) – ауыстырып-қосқыш функция. Оның мүмкін болар қалып күйлерінің саны мен тәуелсіз айнымалы шарының өрқайсысы екіге тең.

Динамикалық қатар – деректердің уақытқа байланысты зерттелуін өрнектейтін функция.

Желінің өткізу қабілеті (деректерді тасымалдау жылдамдығы) уақыт бірлігі ішінде байланыс арнасы арқылы өтетін ақпараттың мөлшерімен анықталады. Өлшем бірліктері: Кбит/сек, Мбит/сек, Гбит/сек.

Жүйелік программа (Системная программа; *system program*) – операциялық жүйенің мүмкіндіктерін арттыратын және ақпараттық жұмыстардың қандай түрін болса да басқаратын программа.

Микропроцессор – жүйелік тақтандың ең маңызды құралы. Ол деректерді тіkelей өнддейді, атап айтқанда, бөлектелген деректермен арифметикалық және логикалық амалдарды орындаиды. Микропроцессор – бір немесе бірнеше үлкен интегралды кестеде орындалған, берілісті өнддейтін программалық құрылғы. Қоліктердің автоматты басқару агрегатында қолданылады.

Жүйелік шина – процессорлар арасында және дербес компьютердің басқа да құрылғыларының арасында ақпарат тасымалдаушы.

Хабардың ақпараттық қөлемі – хабардың ұзындығын, яғни хабарды жазу үшін пайдаланылған символдар саны.

Ықтималдылық теориясы – кездейсоқ бір оқиғаның ықтималдығы бойынша онымен қандай да бір байланыста болатын, басқа бір кездейсоқ оқиғаның ықтималдығын анықтауға мүмкіндік беретін математика білімі. Ықтималдылық теориясында кездейсоқ құбылыстардың заңдылығы зерттеледі. Кездейсоқ құбылыстарға анықталмағандық, курделілік, көп себептілік қасиеттері тән.

Кірістірілген циклдер – бірінің ішіне екіншісін кірістіріп пайдаланған бірнеше қарапайым цикл.

Кибербуллинг (*cyberbullying*) – жасөспірімдер арасында кең тараған виртуалды лаңкестік (террор).

Коммутатор (лат. *commodo* – өзгертемін, ауыстырамын) – электрлік тізбекті жалғастыруға, ажырату мен ауыстырып-қосуға арналған электромеханикалық, электронды немесе электронды-сөүлелік қондырғы.

Компьютерлік желі (ағыл. *computer network*) – хабарламалармен алмасу және желінің ресурстарын бірлесіп пайдалану үшін байланыс

арналары мен коммутация құралдары арқылы бір жүйеге қосылған компьютерлер жиынтығы.

Қолданбалы (арнайы) программалық жасақтама (*application software*) – қолданбалы мәселелерді шешуді жасақтайтын жеке қолданбалы программалар мен қолданбалы дестелерден тұратын программалық жасақтаманың бөлігі.

Қолданбалы программалар интерфейсі (Интерфейс прикладных программ – *application programming interface*):

1) программалаушылар белгілі бір компьютерлік ортада қолданбалы программа жазуға пайдаланатын міндеттер жинағы (интерфейс);

2) операциялық жүйе орындаітын төмен деңгейлі операцияларды жүзеге асыруға қолданбалы программа шақыратын процедуралар жинағы.

Операциялық жүйе (операционная система – *operating system*) – компьютердің басты әрекеттерін (пернелер тақтасын, экранды, дискжеттерді пайдалануды), сондай-ақ операциялық жүйенің басқаруымен іске қосылатын басқа программалардың жұмысын басқаратын, көбінесе тұрақты сақтауыш құрылғыда тұратын, мәшинелік кодта жазылған программа. Алғашқы компьютерлердің операциялық жүйесі болған жоқ, себебі басқару программалары тек компьютердің нақты бір типіне арналып жазылды. Операциялық жүйені жазудың екі түрі бар: тұрақты сақтауыш құрылғыға барлық жүйені жазу және қатқыл дискіден операциялық жүйенің қалған бөлігінің тек жүктеу программаларын ғана жазу. Мекемедегі шағын компьютерлер, әдеттегідей, MS-DOS немесе ең соңғы OS/2 операциялық жүйесін пайдаланады. Ікшам мәшинелерде, дискжеттердің қажеттілігінен аулақ болу үшін, тұрақты сақтауыш құрылғыда жазылған операциялық жүйені пайдаланады.

Параметр (грек. *parametron* – өлшептін, мөлшерләйтін) – мәні кейір жиын элементтерін бір-бірінен ажырату үшін қолданылатын шама.

Программалық жүйелер – программалық өнімдерді дайындау мен енгізу технологиясын қамтамасыз ететін программалар жиынтығы. Программалық жүйелерге трансляторлар, программа дайындаітын орта, анықтамалық программалар кітапханасы, байланыс редакторлары т.б. жатады.

Программалық тестілеу (тестирование – *testing*) – тұтас компьютердің немесе оның жекелеген бөліктері мен программалық жасақтамалары жұмысының дұрыстығын анықтайтын процесс. Альфа-тестілеу (Альфатестирование – *a-testing*) – арнайы жасалған тапсырмалар арқылы дайын программалық өнімді тестілеу. Артығымен тестілеу (избыточное тестирование – *redundancy*) – күрделірек тестіні пайдаланып қайта тестілеу. Бета-тестілеу (бета-тестирование; (*B-testing*) – программалық өнімді келешек өзгертулерге тегін таратып, нақтылы тапсырмалармен тестілеу. Табылған қателер мен ескертулер жайлы программалаушыға хабарлайды. Кездейсоқ тестілеу (стохастическое тестирование – *stochastic testing*) – құрылғыларды, программалар мен жүйелерді, кездейсоқ мәндерді тест

ретінде қолдану арқылы тестілеу. Берілген деректердің барлық мүмкін комбинацияларын шолып шығу қыын болған жағдайларда күрделі жүйелерді тестілеу. Құрылымдық тестілеу (структурное тестирование) – программаның ішкі құрылымын зерттеу арқылы тестілеу. Қызметтік тестілеу (функциональное тестирование) – программаның ішкі құрылымы қарастырылмастан, функционалдық мүмкіндіктерін тестілеу. Өрлемелі тестілеу (восходящее тестирование – *ascending testing*) – программалық өнім төменнен жоғары тексеріліп, жинастырылатын программаны тестілеу өдісі. Мұнда алдымен төменгі модульдер жеке-жеке тексеріліп, олардың интерфейстері одан жоғарырақ деңгейдегі модульдермен бақыланады, сонаң соң осы деңгейдің модульдері тексеріледі.

Салыстырмалы сілтеме – бастапқы деректер орналасқан, көшіру мен ауыстыру кезінде үяшық адресінің формуласының өзгеруі.

Сопроцессор (Coprocessor): 1) негізгі процессордың қызметтік мүмкіндігін толықтыратын микропроцессорлық элемент немесе мамандандырылған арнайы процессор. Олардың ішіндегі кең таралғандарына математикалық есептеулерді жылдам әрі дәл орындаі алатын сопроцессорлар (*math coprocessor*) жатады. Сопроцессорлардың өз программы болмайды, ол негізгі процессор командалары жүйесі мен регистрлерін кеңітіп, сонымен бірігіп жұмыс атқарады. Орталық процессордан айырмашылығы – жүйені басқармайды, тек арифметикалық есептеу мен нөтижелер құрастыруға орталық процессордан команда күтеді.

Соңғы шартты цикл операторы (оператор цикла с постусловием) – 1) *repeat s; s; ... s until B* қаңқасы бойынша орындалатын туынды оператор. Мұндағы *repeat* (қайталу) және *until* (дейін) – қолғабыстық сөздер, ал *s* – кез келген оператор, *B* – логикалық өрнек.

Статистика (лат. *status* – жай-күй) ұғымына төмендегідей сипаттамалар беруге болады.

1. Білім саласы, бұқаралық өлеуметтік-экономикалық құбылыстар мен үдерістердің сандық және сапалық көрсеткіштерін зерделейтін ғылым.

2. Қоғам өмірінің (экономиканың, саясаттың т.б.) сандық заңдылықтарын олардың сапалық мазмұннымен ажырағысыз байланыста сипаттайтын статистика. Ақпаратты жинау, өндеу, талдау мен жариялауды қамтитын практикалық қызмет саласы.

3. Жарияланатын сандық деректердің, бақыланбалы нысанның сан немесе сапа тұлғасындағы күйін тиянақтайтын бақылау нөтижелерінің жиынтығы.

4. Қоғам өмірінің сандық жақтарында болатын құбылыстарды менгеретін, көпжылдық тарихы бар ерекше ғылым немесе білім саласы. Оның шығу төркіні әртүрлі қоғамдық тұтынушыларға, мал санына, жер-су көлеміне, дүние-мұлік және басқа да есебіне байланысты.

5. Әр алуан белгілі бір статистикалық деректер жиынын ретінде де түсінуге болады (өлім статистикасы, торбетке кіру статистикасы т.б.).

Тестілеу – программаның жұмыс қабілетін тексеруге арналған арнайы дайындалған деректерді компьютерге енгізу.

Трассировкалау алгоритмі кездесетін қатемен жұмыс істеу ретін білдіреді. Трассировка – программаны біртіндеп, белгілі бір қадамдар ретімен (step-by-step) орындау процесі.

Функция (лат. *functio* – орындау, атқару, жүзеге асыру) – көп мағыналы сөз.

1. функция – математикадағы негізгі ұғымдардың бірі;

2. қызмет, міндет, жұмыс; қатынастар жүйесіндегі қайсыбір объект қасиеттерінің сыртқы көрінісі;

3. социологияда – белгілі бір әлеуметтік институт атқаратын рөл;

4. лингвистикада – тілдік бірліктің немесе тілдің құрылымы.

Чипсет (*Chipset*) – үқсас міндет атқаратын микросхемалар жиыны. Мысалы, орталық процессордың шалғай құрылғылармен деректер алмастыру жұмысын жасақтайтын (аналық тақшада орналасқан) микросхемалар жиыны. Атқаратын міндеттері бір микросхема ретінде жүзеге асыруға болмайтындықтан бірнеше бөлікке бөлінетін жүйелік тақшалар. Бейнесхемалық контроллерлер сияқты күрделі бөлшектерде кеңінен қолданылады.

Цикл (*cycle, loop*) де көп мағыналы сөз.

1) алгоритмнің немесе программаның бірнеше рет орындалатын бөлігі; ол берілген айнымалы белгілі бір мәнге жеткенше немесе бір көрсетілген шарт орындалып болғанша атқарылатын программадағы бірнеше рет қайталая операторлар жиынынан тұрады;

2) сол бір тізбек бойымен тұрақты түрде қайталап отыратын операциялар жинағы. Операцияны әрбір қайталая кезінде өзгеріс болады.

Циклдік алгоритм – бастапқы қойылған шарт орындалғанша қайталанатын командалар тізбегін көрсететін алгоритмдік құрылым. Циклдер көмегімен программа жазу программа жазу процесін жеңілдетеді.

Электромагниттік толқындар – байланыс тізбегін құрайтын екі сымның арасындағы электрлік және магниттік өрістер бір-бірімен белгілі бір электромагниттік энергия мөлшерінде байланыста болатын толқын.

IP-адрес (ағыл. *Internet Protocol address*) – интернет протокол адресі әр құралға (компьютер, принтер т.б.) жеке-дара бекітілген нөмірлі белгі. Бұл құралдар компьютер желісінде өзара қарым-қатынас жасау үшін интернет протоколын қолданады.

Break (ломать – үзілу) – белгілі бір қойылған шарт орындалуымен ағымдағы цикл жұмысын бірден тоқтатып, программа жұмысын циклден шығарып, әрі қарай жалғастыратын цикл нұсқаулығы.

Continue (продолжить – жалғастыру) – белгілі бір қойылған шарттың көмегімен ағымдағы цикл итерациясын тоқтатып, келесі цикл итерациясына ауыстыратын цикл нұсқаулығы.

Пайдаланылған әдебиеттер

1. Экономикалық ақпараттық жүйелердегі жаңа технологиялар: экономикалық мамандықтар студенттерінің тәжірибелік сабактарына арналған әдістемелік нұсқау / құраст.: Бахралинова Х.Г., Богданова Е.А. – Павлодар: Кереку, 2012. – 55 бет.
2. Каракулов Ю.А., Иванов А.Н. Руководство к решению задач с применением электронных таблиц EXCEL. Учебное пособие. Санкт-Петербург.
3. Карчевский Е.М., Филиппов И.Е., Филиппова И.А.: Excel 2010 в примерах. Учебное пособие – 2012.
4. Левин А. – Самоучитель Левина. Windows XP и Vista (Самоучитель Левина) – 2007.
5. Хакимова Т. «Компьютерлік өндөудің әдістемелері». Оқу құралы. – Алматы – 2007.
6. Гульев И. А. «Компьютерные вирусы, взгляд изнутри».
7. Холмогоров В. «Компьютерная сеть своими руками. Самоучитель». – 2004.
8. Шевелев Г.Е. «Информатика: Лабораторный практикум». – Томск, 2004.
9. Киселев В.Г. «Практическая работа в EXCEL 2007». – Нижний Новгород – 2009.
10. Горбачев А., Котлев Д. «Microsoft Excel. Работайте с электронными таблицами в 10 раз быстрее». – Москва – 2007.
11. Сэнд У., Сэнд К. «Hello World! Занимательное программирование». – СПб.: Питер, 2016. – 400 с.: ил. – (Серия «Вы и ваш ребенок»).
12. Мэтиз Эрик. Изучаем Python. Программирование игр, визуализация данных, веб-приложения. 2-е изд. – Издательский дом «Питер» – 2017.
13. Хахаев И.А. Практикум по алгоритмизации и программированию на Python (2-е изд.)
14. Седжвик, Роберт, Уэйн, Кевин, Дондеро, Роберт. C28 Программирование на языке Python: учебный курс. : Пер. с англ. - СПб. : ООО «Альфа-книга» : 2017. – 736 с. : ил. – Парал. тит. англ.
15. Абрамян М. Э. 1000 задач по программированию.
16. <https://pythonworld.ru/osnovy/cikly-for-i-while-operatory-break-i-continue-volshebnoe-slovo-else.html>.
17. <http://programm.ws/page.php?id=686>
18. <http://www.tvd-home.ru/prog/c6>.
19. <http://www.studfiles.ru/preview/2618857/page:20/>.
20. http://dipls.ucoz.kz/publ/algoritmdeu_zhne_programma-lau/1-1-0-2
21. https://median.kz/kz/blog_companii/press_center/?title = cloud_computing_nemese_blttyi_eseteu_tehnologiyalaryi_degen_tsng_turalyi_ne_blemz.

МАЗМУНЫ

Құрметті оқушы!	3
I бөлім. Компьютер мен желілердің техникалық сипаттамалары	
1.1. Ақпаратты өлшеу. Алфавиттік өдіс.....	4
1.2. Ақпаратты өлшеу. Ікәншілдік өдісі	8
1.3. Компьютердің ішкі құрылымы мен қосымша құрылғылары ...	12
1.4. Процессор мен оның сипаттамалары	19
1.5. Компьютерлік желілер	25
II бөлім. Денсаулық және қауіпсіздік	
2.1. Компьютерді пайдаланудың кері аспектілері.....	30
2.2. Желідегі қауіпсіздік	35
III бөлім. Ақпаратты электронды кестелерде өндіру	
3.1. Статистикалық деректер. Абсолютті және салыстырмалы сілтемелерді қолдану	40
3.2. Абсолютті және салыстырмалы сілтемелерді практикалық түрғыдан қолдану	46
3.3. Кірістірілген функциялар	51
3.4. Кірістірілген функцияларды практикалық түрғыдан қолдану	56
3.5. Қолжетімді ақпараттың негізінде деректерді талдау.....	60
3.6-3.7. Қолданбалы есептерді шешу	66
3.8. Excel программасында жобалық жұмыстар орындау	75

IV бөлім. Python тілінде алгоритмдерді программалау	
4.1. For циклі.....	78
4.2. Кірістірілген циклдерді үйымдастыру.....	83
4.3. While циклі.....	87
4.4. Continue циклін басқару	91
4.5. Break циклін басқару	95
4.6. Else циклін басқару	99
4.7. Алгоритмнің трассировкасы	103
4.8–4.9. Программалауды үйренейік. Практикалық тапсырмалар	107
4.10. Программалау бойынша жоба жұмыстарын орындау	111
V бөлім. Практикалық программалау	
5.1. Мәселе қою (қалыптастыру).....	114
5.2. Алгоритмді өзірлеу	119
5.3–5.4. Алгоритмді программалау. Практикалық жұмыс	123
5.5–5.6. Программаны тестілеу	127
5.7. Олимпиадаға дайындық. Практикалық тапсырмалар	134
Сөздік – словарь – dictionary	137
Глоссарий	146
Пайдаланылған әдебиеттер	149

**Кадиркулов Роман Алауович
Рыскулбекова Асима Даuletбековна
Нурмуханбетова Гулира Кенжебаевна**

ИНФОРМАТИКА ИНФОРМАТИКА

Жалпы білім беретін мектептің
8-сынып оқушыларына арналған оқулық
Учебник для учащихся 8 класса
общеобразовательной школы

Редакторы / Редактор – *Л.С. Ахметова*
Мұқаба / Обложка – *Е. Жұзбаев*
Беттеуші / Верстка – *Г. Тұмарбайқызы*

Басыға 22.05.2023 ж. қол қойылды.
Пішімі 70x100^{1/16}. Есептік баспа табагы 5,73.
Шартты баспа табагы 12,35. Офсеттік басылым.
Әріп түрі «DS SchoolBook». Офсеттік қағаз.
Қосымша таралымы 15 000 дана. Тапсырыс № 3092.

Подписано в печать 22.05.2023 г.
Формат 70x100^{1/16}. Уч.-изд.л. 5,73.
Усл.печ.л. 12,35. Печать офсетная.
Гарнитура «DS SchoolBooks». Бумага офсетная.
Доп. тираж 15 000 экз. Заказ № 3092

Сапасы жөнінде шағымыңыз болса,
мына мекемеге хабарласыңыз:
«АЛМАТЫКИТАП БАСПАСЫ» ЖШС,
050012, Алматы қаласы, Жамбыл көшесі, 111-үй,
т. (727) 250 29 58, факс: (727) 292 81 10.
e-mail: info@almatykitap.kz

С претензиями по качеству обращаться:
Республика Казахстан,
ТОО «АЛМАТЫКИТАП БАСПАСЫ»
050012, г. Алматы, ул. Жамбыла, 111,
тел. (727) 250 29 58; факс: (727) 292 81 10.
e-mail: info@almatykitap.kz

Сапа және қауіпсіздіктің барлық стандартына сай.
Сертификаттау қарастырылмаған.
Сақтау мерзімі шектелмеген.

Соответствует всем стандартам качества и безопасности.
Сертификация не предусмотрена.
Срок годности не ограничен.

Түркіяда басылды / Отпечатано в Турции
Avea Basim Yayin San. ve Tic. Ltd.Sti
Cihangir mah. Guvercin cad. No:3/1. Baha is merkezi A Blok Kat:2 34310. Haramidere - Istanbul

Кітаптарды «АЛМАТЫКИТАП БАСПАСЫ» ЖШС-ның кітап дүкендерінен сатып алуға болады.
Астана қаласы: Иманов көшесі, 10, тел.: (712) 53 70 84, 27 29 54;
Б. Момышұлы даңғылы, 14, тел.: (712) 42 42 32, 57 63 92.
Алматы қаласы: Абай даңғылы, 35/37, тел.: (727) 267 13 95, 267 14 86;
Гоголь көшесі, 108, тел.: (727) 279 29 13, 279 27 86; Қабанбай батyr көшесі, 109, тел.: (727) 267 54 64, 272 05 66;
Жандосов көшесі, 57, тел.: (727) 303 72 33, 374 98 59; Майлина көшесі, 224 «А», тел. (727) 386 15 19;
Төле би көшесі, 40/1, тел.: (727) 273 51 38, 224 39 37.

Интернет-дүкен: www.flip.kz
Сауда белгімі, тел.: (727) 292 92 23, 292 57 20.
e-mail: sale1@almatykitap.kz

Кітаптар мен басылымдар туралы мағлұматтарды
www.almatykitap.kz сайты арқылы білуге болады.



Оқулықты пайдалану туралы деректер. Сведения о пользовании учебником

№	Оқушының аты-жөні Фамилия и имя ученика	Оку жылы Учебный год	Оқулықтың жағдайы Состояние учебника	
			Жылдың басында В начале года	Жылдың аяғында В конце года
1				
2				
3				
4				
5				