

Қазақстан Республикасы Ғылым және жоғары білім министрлігі  
Е.А. Бөкетов атындағы Қарағанды университеті

Құрастырғандар:  
Кельдибекова Алия Болатовна – доцент PhD  
Каменова Шынар Кайкеновна – аға оқытушы

Математика және ақпараттық технологиялар факультеті

Қолданбалы математика және информатика кафедрасы

Оқу бағдарламасы «Алгоритмдеу және программалау тілдері» пәнін оқытуға арналған. Бағдарлама негіздерімен және программалау әдістері мен танысу, қойлған есепке алгоритм және оған Python бағдарламалау тілінде бағдарлама құрып, шыққан нәтижені талдау білу.

#### «АЛГОРИТМДЕУ ЖӘНЕ ПРОГРАММАЛАУ ТІЛДЕРІ» пәні бойынша

6B01505-Информатика  
білім беру бағдарламасы

#### ОҚУ БАҒДАРЛАМАСЫ (SYLLABUS)

Курс: 2

Оқу мерзімі: 4 ж.

Қредит саны: 7

«Алгоритмдеу және программалау тілдері» пәні бойынша  
6B01505-Информатика білім беру бағдарламасының  
студенттеріне арналған оқу жұмыс бағдарламасы  
/Құрастырғандар: А.Б. Кельдибекова, Ш.К.Каменова - Қарағанды:  
ҚарУ баспасы, 2023. – 61б.

Қарағанды 2023

© Қарағанды университеті, 2023

## 1. Күндізгі оқыту нысанының жұмыс оқу жоспары

№	Тақырып атауы	Дәріс	Семинарлық сабақ	СӨЖ	СӨЖ
1	Программалау негіздері және бастамалары. Программалау құралдары.	1	2	1	
2	Python бағдарламалау тіліне кіріспе. IDLE ортасымен танысу. Алфавит. Мәліметтер типі	2	3	1	5
3	Python-да енгізу және шығару. Диалог режимі. Файлдан оқу және файлға жазу	2	3	1	5
4	Тармақтау және таңдау операторлары	2	3	1	5
5	Циклдермен жұмыс	2	3	1	5
6	Тізімдермен жұмыс. Python-дағы операциялардың тізімі	2	3	1	10
7	Процедуралар және функциялар	3	3	1	10
8	Екі өлшемді массивтермен жұмыс	2	3	1	5
9	Объектілі-бағытталған бағдарламалау. Класстар мен объектілер	2	3	1	10
10	Модульдер	3	4	1	10
11	Графикалық интерфейс бар қосымшаларды әзірлеу. Тасбақа көмегімен сурет салу	2	3	2	5
12	GUI қосымшаларын құру. Tkinter модулі	2	4	1	5
13	Tkinter модулімен жұмыс істеу негіздері	3	4	1	5
14	Деректерді визуализациялау. Диаграммалар тұрғызу	2	4	1	5
	Барлығы	<b>30</b>	<b>45</b>	<b>15</b>	<b>85</b>

## 2. Оқытушы туралы мәліметтер:

Кельдибекова Алия Болатовна, Keldibekova\_A\_V@mail.ru  
 Каменова Шынар Кайеновна, Kapenova74@mail.ru  
 Байланыс орны: Математика және ақпараттық технологиялар факультеті, №2 корпус, 321 кабинет, сабақ кестесіне сәйкес.

## 3. Пререквизиттері:

Ақпараттық-коммуникациялық технологиялар, Шетел тілі, Кәсіби бағытталған шетел тілі, Математикалық талдау, Жоғарғы математика.

## 4. Постреквизиттері:

Объектно-ориентированное программирование, Информатика бойынша күрделілігі жоғары есептер, Қолданбалы информатиканың есептері, Современные педагогические технологии в образовании, Инновационные технологии в организации учебного процесса в школе, Информатиканы оқыту әдістемесі

**Білім мен түсінгенін қолдану:** түрлі алгоритмдердің құрылымдық схемаларын жасау, есептің талаптарына байланысты қажетті деректер құрылымдарын ұйымдастыру, программалау тілін таңдау, тілдің құралдарын пайдаланып таңдалған тілде программаларды жақсы стильде жазу, программаларды тексеру және тестілеу, сапалы программалық құжаттама құру.

**Пікір айтуды қалыптастыру:** туындаған есептер мен проблемаларды жүйелі түрде ойлау және дұрыс жіктеу мүмкіндігі; өздігінен оқуға, өзін-өзі дамыту және тәуелсіз зерттеуге ұмтылу.

**Коммуникативтік мүмкіндіктер:** командада жұмыс істей білу, өз көзқарасын дұрыс қорғау, проблемалық жағдайларда жаңа шешімдерді ұсыну, кәсіби қызмет үшін күн сайын жаңа білім алу дағдылары; кәсіби қызметте тиімді пайдалану үшін заманауи ақпараттық және коммуникациялық технологиялармен жұмыс істеу дағдылары.

**Оқу дағдылары:** алгоритмдердің негізгі құрылымдарын құру, жоғары деңгейлі программалау тілдерінде программалар әзірлеу, программалық өнімдерді енгізу және тестілеу. Белгілі бір мәселені шешуге арналған алгоритмдерді таңдауға және құруға құзыретті болу.

### 5. Пәннің қысқаша мазмұны:

Студентті қазіргі дербес компьютермен кәсіби жұмыс жасай алуға үйрету, әрі қарайғы компьютерлік техникамен белсенді жұмыс жасай алудың кейбір ерекшеліктерін меңгеруді үйрету болып табылады.

Алынатын білім деңгейі ақпараттарды әртүрлі деңгейде өңдеудің базалық білімі және қоғамды басқару жүйесінің ерекшелігі болып табылады.

Теориялық білім мазмұнына дербес компьютерде ақпараттарды өңдеудің негізі, бағдарламалаудың негізі, оның ішінде Python тілінде бағдарлама жазудың негізі қарастырылады. Практикалық сабақтарды өту кезеңінде студент құрылғылармен танысады, дербес компьютерді қолдану қағидасымен танысады, ақпаратты енгізу/шығару қағидасымен, оларды қолдану ережелерімен танысады. Практикалық сабақты өту үшін компьютерлік класстар қолданылады.

Сабақты жүргізу барысында нұсқаулар беріліп отыру керек: Ерекше көңіл қауіпсіздік ережесімен студенттің жұмыс жасау орынын анықтауы болып табылады.

Теориялық және практикалық сабақта студенттің материалды қаншалықты қабылдағаны, берілген тапсырманың орындалғаны, қазіргі компьютер жүйесінде тез шығара алуы толығымен қарастырылады.

Жұмыстың мазмұны өзінің кәсіби ісінде компьютерлік техниканы белсенді қолдануы үшін ұсынылады.

### 6. Пән бойынша тапсырмаларды орындау және өткізу кестесі

N	Жұмыс түрі	Тапсырма мазмұны және мақсаты	Пайдалануға ұсынылатын әдебиеттер	Бақылау түрі (рейтингтік шкалаға сәйкес)	Бағалар (рейтингтік шкалаға сәйкес)	Есеп беру түрі	Тапсыру уақыты
1	Семинарлық, СОӨЖ сабақтарына қатысу.	Жоспар бойынша	Пәннің оқу-әдістемелік картасында көрсетілген	АБ 1	50-100	Ауызша	Сабақ кестесіне сай
2	Коллоквиум	СОӨЖ, семинар жоспары бойынша	Пәннің оқу-әдістемелік картасында көрсетілген	АБ 2	50-100	Ауызша	6-7, 10-14 апталарда
3	Реферат	СОӨЖ жоспары бойынша	Пәннің оқу-әдістемелік картасында көрсетілген	АБ 3	50-100	Ауызша	Сабақ кестесіне сай
4	Бақылау	Жоспар бойынша	Пәннің оқу-әдістемелік картасында көрсетілген	АБ 4	50-100	Жазбаша	Сабақ кестесіне сай
5	Белсенділіктің анықтау	Жоспар бойынша	Пәннің оқу-әдістемелік картасында көрсетілген	АБ 5	50-100	Ауызша	Сабақ кестесіне сай

## 7. Пәннің оқу-әдістемелік картасымен қамтамасыз етілуі

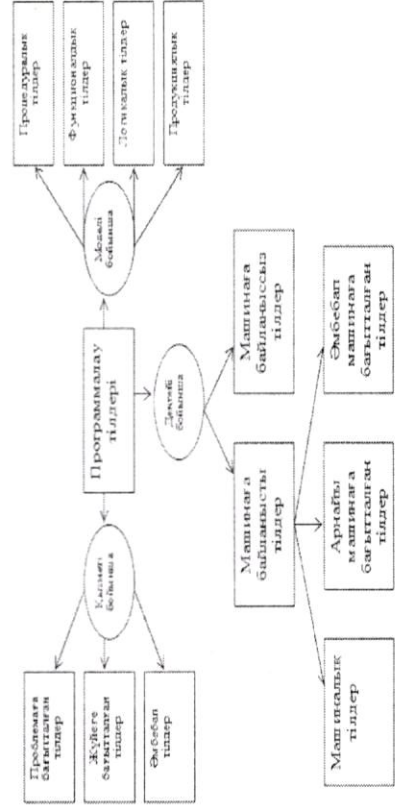
№	Курс бойынша оқулықтар мен оқу құралдарының атауы	КарМУ кітапхана-сындағы оқулықтар мен оқу құралдарының саны
I. Негізгі әдебиеттер		
1	Коксеген, Ә. Е. Алгоритмдеу және программалау тілдері: оқулық / Ә. Е. Коксеген, Ә. О. Сейфуллина. - Алматы : Дәуір, 2023. - 191 б.	9
2	Ғұмаров, Ж. Алгоритмдеу және бағдарламалау негіздері: есептер жинағы: оқу құралы / Ж. Ғұмаров. - Астана : Фолиант, 2023. - 196 б.	30
3	Кучма Н.Г., Введение в программирование: пособие / Н. Г. Кучма ; Карагинский гос. ун-т. - Изд. 2-е, перераб. и доп. - Караганда : Изд-во КарУ, 2023. - 401 с.	50
4	Сузи, Р.А. Язык программирования Python : курс / Р.А. Сузи. - 2-е изд., испр. - Москва : Интернет-Университет Информационных Технологий, 2023. - 327 с.	3
5	Шекербекова, Ш. Т. Программалау: Python тілінде программалау: оқу құралы / Ш. Т. Шекербекова, И. Т. Салғожа ; Абай атынд. Қазақ Ұлттық Педагогикалық ун-ті. - Алматы : Нұр-Принт, 2023. - 199 б.	5
2. Қосымша әдебиеттер		
6	Койбағарова, Т. Python программалау тілі: оқулық / Т. Койбағарова. - Алматы : Эверо, 2023. - 264 б.	45
7	Нұрымбетов, Ә. Ү. Python алгоритмдік тілінде бағдарламалау: оқулық / Ә. Ү. Нұрымбетов. - Алматы : Дәуір, 2023. - 199 б.	8
8	Аубакиров, Г. Д. Языки программирования Python: учеб. пособие для организации техн. и проф. образования / Г. Д. Аубакиров, А. Г. Хмыров. - 2-е изд. - Астана : Фолиант, 2023. - 203 с.	10
9	Смайлова, Ұ. М. Программалау: алгоритм құру технологиялары: оқу құралы / Ұ. М. Смаилова. - 2-ші басылым, толықт. - Алматы : Асыл кітап, 2023. - 168 б.	10
10	Лутц, М. Программирование на Python. Т.2/М.Лутц. - М.:Символ, 2023.-992 с	39
11	Задачи по программированию. /С.А. Абрамов, Г.Г. Гнездилова, Е.Н. Капустина, М.И. Селюн. - М.:Наука, 2023.-224 с.	85
2.1 Мерзімдік әдебиеттер (басылымдар) тізімі		
12	Журнал «В мире науки» 2023	кітапхана 2 қорп.

13	Вестник Карагандинского университета Серия математика 2023	кітапхана 2 қорп.
2.2 Электрондық оқулықтар мен оқу құралдары тізімі		
14	Курс лекций по дисциплине "Программалау тілдері" [Электронный ресурс]	Электронды кітапхана
15	Электрондық дәріс «Программалау тілдері»	Электонды кітапхана
2.3 Интернет көздері		
16	<a href="http://www.prgramming.ru">http://www.prgramming.ru</a>	
17	<a href="http://python.cs.msu.su/library/">http://python.cs.msu.su/library/</a>	
18	<a href="http://www.programming-free.python.ru">http://www.programming-free.python.ru</a>	
19	<a href="https://python-scripts.com/pdf-splitter-appjar">https://python-scripts.com/pdf-splitter-appjar</a>	
20	<a href="https://younglinux.info/oopython/module.php">https://younglinux.info/oopython/module.php</a>	

## 8. Лекциялық кешен (лекция тезісі)

### Тақырып. Программалау негіздері және бастамалары. Программалау құралдары.

Машиналық тілдердегі барлық бұйрықтар мен нұсқаулар компьютердің құрамына кіретін функционалдық бөліктер арқылы орындалады. Машиналық тілдер компьютерлердің алғашқы интеллектуалдық (зерделік) деңгейін анықтайды. Кейін неше түрлі программаларды компьютердің жадына сақтай отырып, оның интеллектуалдық деңгейін көтеруге болады.



1-сурет Программалау тілдерінің кестесі

Жасанды тілді компьютер өзінің машиналық тіліне адамның қатысуынсыз (автоматты түрде) аударатындай етіп жасайды. Аударуды программа орындайды (Сурет 1). Ондай программаны ағылшын тілінде «транслятор» (аудармашы) деп атайды, ал аудару процесін «трансляция» дейді. Сондықтан программаны жасанды тілді пайдаланып жазған кезде компьютердің жұмысы екі кезеңнен тұрады:

1. Жасанды тілден машиналық тілге трансляциялау;
  2. Машиналық тілдегі программаны орындау.
- Қазіргі кезде компьютердің көмегімен әр түрлі есетерді шығаруға мүмкіндік беретін сан алуан программалау тілдері бар. Әр тілдің кемшілігі де, жетістігі де бар. Мысалы, кейбір тілде программаны жазу оңай болғанымен оны орындау өте көп уақытты талап етеді немесе жадының көп орнын алады.

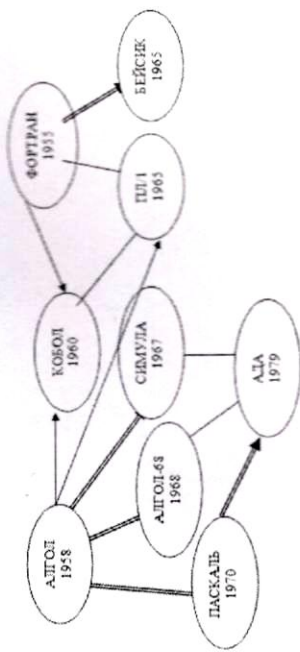
Жалпы, программалау тілдерін мынадай белгілері, қасиеттері бойынша жіктеуге болады:

- 1) Тілдің деңгейіне байланысты, яғни машиналық тілге жақындығына байланысты;
- 2) Тілдің қызметіне байланысты, яғни қандай есетерді шешуге мүмкіндік беретініне байланысты;
- 3) Тілдің моделіне байланысты, яғни құрылу принципіне байланысты.

Программалау тілдері, табиғи тілдері сияқты, өздерінің моделі (құрылу негізі) бойынша бөлінеді. Мысалы, табиғи тілдерді біз түркі тілдері, славян тілдері, роман тілдері, араб тілдері, парсы тілдері және т.с.с. деп топқа бөлеміз. Ал программалау тілдерін процедуралық тілдер, функционалды тілдер, логикалық тілдер және продукциялық (алмастырымдық) тілдер деп бөлуге болады.

Программалық тілдердің ішінде ең алғаш пайда болғандары және ең көп тарағандары процедуралық тілдер, ал процедуралық тілдердің ішіндегі ең бірінші пайда болғаны ФОРТРАН (1955ж.).

Төменгі 2-суретте қазіргі кезде кеңінен тараған, былайша айтқанда, «классикалық» процедуралық тілдердің даму тарихы көрсетілген.



2-сурет Процедуралық программалық тілдердің даму тарихы

### Негізгі әдебиеттер: [1-5] Қосымша әдебиеттер: [6-11]

**Тақырып Python бағдарламалау тіліне кіріспе. IDLE ортасымен танысу. Алфавит. Мәліметтер типі**

Python - жоғары дәрежелі кодтың оқылуын және әзірлеушінің өнімділігін арттыруға мақсатталған жалпы мақсаттағы бағдарламалау тілі. Python тілі аз синтаксисті талап етеді. Бірақ сол уақытта стандартты кітапханасы үлкен көлемді пайдалы функцияларды қамтиды.

Python тілі қазіргі кезде ең көп таралған бағдарламалау тілі болып саналады. Бұл тілде жасалған ең атакты бағдарлама ол бәріміз білетін Instagram әлеуметтік желісі. Бұдан басқа BitTorrent, Ubuntu, Blender, GIMP, Civilization IV секілді бағдарламаларды кездестіре аламыз.

Python интерпретаторымен бірге жиынтықта IDLE (интеграцияланған даму ортасы) бар. Өзінің мәні бойынша ол кеңейтілген мүмкіндіктер жиынтығымен интерактивті режимде іске қосылған интерпретаторға ұқсас.

Бағдарлама комментарий; командалар; пунктуация белгілері; идентификаторлар; кілт сөздерден тұрады.

Python-дағы пікірлер олардың алдын ала # символымен белгіленеді және жолдың соңына дейін жалғасады (яғни Python-да барлық пікірлер бір жолдық болып табылады), бұл ретте # тырнақшаның алдында қолдануға жол берілмейді:

>>> num = num \* 3 # сокращенно: num \*= 3

Python алфавитінде әр түрлі мақсатта қолданылатын тыныс белгілерінің жеткілікті саны бар. Мысалы, «+» немесе «\*» белгілерін қосу және көбейту үшін, ал үтір «,» функциялардың параметрлерін бөлу үшін қолданылады.

Pythonда идентификаторлар айнымалыны, функцияны, классты, модульді немесе басқа объектіні білдіретін атаулар.

Python тілінде жиі қолданылатын кілт сөздер: 'False', 'None', 'True', 'and', 'as', 'return', 'try', 'while', 'with' және т.б. Кзлттзк с-здер IDLE қызғылт сары түспен боялады.

Мәліметтер типтері:

- None (айнымалының белгісіз мәні);
- Логикалық айнамалылар: True (ақиқат), False (жалған).

Логикалық мәндер логикалық операциялар және логикалық өрнектерді есептеу нәтижесінде алынады.

#### Кесте Негізгі логикалық операциялар мен өрнектер

Операция немесе өрнек	Мағынасы
>	«Көбірек» шарты (мысалы, a > b екенін тексереміз)
<	«Аз шарты» (мысалы, a < b екенін тексереміз)
==	Теңдік шарты (a-ның b-ға тең екенін тексерелі)
!=	Тенсіздік шарты (a-ның b-ға тең емесігін тексереміз)
not x	Бас тарту (x шарты қанағаттандырылмайды)
x and y	Логикалық "және" (көбейту). X және y шартын орындау үшін X және y шарттарын бір мезгілде орындау қажет
x or y	Логикалық "немесе" (қосу). X or y шартын орындау үшін шарттардың бірін орындау қажет.
x in A	X элементінің A жиынына (құрылымына) жататынын тексеру
a < x < b	X > a) және (x < b) -ге тең

- Сандар:

1. int-бүтін сан
2. float-өзгермелі нүктесі бар сан
3. complex-комплекс сандар. Python сандарына математикалық операторлар қолданылады.

#### Кесте Математикалық операторлар

Оператор	Мағынасы
+	Қосу
-	Азайту
/	Бөлу (санның бүтін бөлігі)
*	Көбейту
//	Төмен дөңгелектеу арқылы бөлу
**	Дәрежеге шығару
%	Бөлуден қалған қалдық

- Тізімдер:

1. list – тізім
2. tuple – кортеж
3. range – аралық

Python-дағы кортеждер - бұл бір уақытта әртүрлі типтегі объектілерді (сандар, жолдар және басқа құрылымдар, соның ішінде кортеждер) қамтитын реттелген нысандар жиынтығы. Кортеж үтір арқылы дөңгелек жақшада оның элементтерін тізімдеу арқылы жазылады: T=(12, 'B', 3 4 . 6, 'degree' ).

- Жолдар: str.

Сонымен қатар, Python-да сандарға арналған abs (абсолюттік мәнді есептеу — модуль, abs (-3)→ 3), pow(дәрежеге тұрғызу, pow(2,3) → 8), div mod (бүтін бөлу және қалдық нәтижесін есептеу, divmod (17,5)→ (3,2)) және round (дөңгелектеу, round (100.0/6) → 17.0) функциялары қолданылады.

Квадрат түбірі, синус және т.б. есептеу сияқты сандарға (математикалық) жұмыс істеу үшін барлық басқа функциялар math модулін қосуды талап етеді. Модульдер бағдарламаның басында *import модуль атауы* командасы арқылы қосылады.

Негізгі әдебиеттер: [1-5]

Қосымша әдебиеттер: [6-11]

Тақырып Python-да енгізу және шығару. Диалог режимі. Файлдан оқу және файлға жазу

Жұмыс кезінде интерактивті қабықшамен жұмыс істегенде немесе бағдарламаның қолданушымен өзара әрекетін ұйымдастыру барысында («диалог») input () және raw\_input () функцияларын пайдаланып, сандарды енгіземіз және сәйкесінше

айнымалылардың мәндерін анықтаймыз.

Input () функциясын пайдаланғанда сандық мәндер едетте жазылады, ал жолдар тырнақшаларда жазылуы керек (қос немесе жалғыз). Мысалдар:

```
>>> a=input(' # карапайым нұсқа  
12 <ENTER>  
>>> a <ENTER> # а айнымалы мәнін тексереміз  
12  
>>>  
>>> a,b,c=input('Үшбұрыш _ кабырғаларының _ ұзындықтарын _ үтір _ а  
рқылы _ жазыңыз: _ ') <ENTER> # бірнеше сандарды қатар қолдану  
Үшбұрыш кабырғаларының ұзындықтарын үтірарқылы жазыңыз: 3, 4, 5  
>>> b # b айнымалысының мәнін тексереміз  
4  
>>>
```

Python-да жол мәндерін енгізу үшін raw\_input () функциясы қолданылады. Оның ерекшеліктері көбінесе input() функциясымен сәйкес келеді. Мысалы:

```
>>> name=raw_input ('Сенің_ атың_ кім? _ ') <ENTER>
```

```
Сенің атың кім? Вася <ENTER>
```

```
>>> name <ENTER>
```

```
'Вася'
```

```
>>> age=raw_input ('Сенің_ жасың_ нешеде? _ ') <ENTER>
```

```
Сенің жасың нешеде? 12 <ENTER>
```

```
>>> age <ENTER>
```

```
'12'
```

```
>>>
```

Жұмыс нәтижелерін шығару үшін функция болып табылмайтын print нұсқауы қолданылады. Нұсқауларды (командаларды) басып шығару есептеулерді орындауға және бір уақытта (бір оператормен) жолдар мен сандарды көрсетуге мүмкіндік береді.

```
>>> t=(a,b,c,d)=input ('Элементтерді _ енгізіңіз : _ ') <ENTER>
```

```
5, 'bob', -3, 'dno' <ENTER>
```

```
>>> t1=(1,2,3) <ENTER>
```

```
>>> print 'Қортынды _ кортеж', t+t1 <ENTER> #Түсініктеме және есептеу
```

```
Қортынды кортеж (5, 'bob', -3, 'dno', 1, 2, 3)
```

```
>>>
```

### Файлдан оқу және файлға жазу

Файлмен жұмыс істеу үшін алдымен арнайы объект - «файл дескрипторын» жасау керек, содан кейін мәліметтерді оқу және жазу үшін осы дескриптордың әдістерін қолдану керек.

Дескрипторды құру кезінде файл атауы жазылған жолды және қатынас опциясы сипатталған жолды көрсету керек. Үш қол жетімді опциялар бар - 'r' (тек оқуға), 'w' (жазу) және 'a' (қосымша).

Оқу тек қолда бар файлдан мүмкін. Егер жазбаны немесе қосымшаны ашу кезінде жоқ файлдың атауы көрсетілсе, ол файл жаңадан құрылады.

Бірнеше мысалдарды қарастырып өтейік.

1.txt деген деректер файлы болсын. Осы файлдан деректерді оқу үшін дескриптор құрамыз: >>> f=open ('1.txt', 'r')

Файлдағы жолдарды оқимыз:

```
>>> s=f.read ()
```

```
>>> s
```

```
'1 2 3\na b c'
```

```
>>>
```

Файлмен жұмысты тоқтату үшін ("дескрипторды босату") close() әдісі қолданылады: >>> f.close().

Файлға жаңа мәліметті жазуды қарастырайық. Жазу үшін дескриптор жасау керек, жолды қалыптастыру және төменде көрсетілгендей write () әдісін пайдалану керек.

```
>>> f=open ('1.txt', 'w')
```

```
>>> s2='c_d_e'
```

```
>>> f.write (s2)
```

```
>>> f.close ()
```

Нәтижесінде, файлдың мазмұны 'c\_d\_e' жолымен ауыстырылады және файлдағы бұрынғы жазба жойылады. Close () әдісі файлға өзгертулер енгізіліп, дискіге жазылуын қамтамасыз етеді.

Енді жаңа файлды құру және оған деректерді қосу жағдайын қарастырайық.

```
>>> f=open ('2.txt', 'a')
```

```
>>> s1='c_d_e\n'
```

```
>>> f.write (s1)
```

```
>>> s2='3_4_5\n'
```

```
>>> fd.write (s2)
>>> fd.close ( )
```

Нәтижесінде біз 2.txt файлын аламыз.

### Негізгі әдебиеттер: [1-5]

Қосымша әдебиеттер: [6-11]

### Тақырып Тармақталу және таңдау операторлары

Алгоритмдік құрылымның негізгі бір бөлігі болып альтернатива табылады.

If тармақталу операторы кейбір жағдайларға байланысты белгілі бір нұсқаулық жиынтығын орындауға мүмкіндік береді.

Келесі пайдалану опциялары мүмкін:

#### 1. if конструкциясы

if операторының синтаксисі төмендегідей жазылады:

#### if логикалық өрнек:

```
команда_1
команда_2
...
команда n
```

If операторынан кейін логикалық өрнек жазылады. Логикалық өрнек-есептеу нәтижесі "ақиқат" немесе "жалған" болып табылатын бағдарламалау тілінің құрылымы.

Егер бұл өрнек ақиқат болса, онда осы оператор анықтайтын нұсқаулықтар орындалады.

**Ескерту:** өрнек ақиқат болған жағдайда, орындау қажет код блогы сол жақта төрт бос орынмен бөлінеді!

#### 2. if – else конструкциясы

Бағдарламаны орындаудың баламалы нұсқасын қарастыру қажет болған жағдайлар да болады. Яғни, ақиқат жағдайда нұсқаулардың бір жиынтығын, жалған болса – екіншісін орындау керек. Ол үшін if-else құрылымы қолданылады.

If-else операторының синтаксисі төмендегідей жазылады:

#### if логикалық өрнек:

```
команда_1
команда_2
...
```

```
команда_n
else:
команда_1
команда_2
...
```

```
команда_n
```

### 3.if – elif – else конструкциясы

Бірнеше баламаларды таңдау үшін if-elif-else құрылымын пайдалануға болады.

if-elif-else операторының синтаксисі төмендегідей жазылады:

#### if логикалық өрнек\_1:

```
команда_1
команда_2
...
```

```
команда_n
```

#### elif логикалық өрнек\_2:

```
команда_1
команда_2
...
```

```
команда_n
```

#### elif логикалық өрнек\_3:

```
команда_1
команда_2
...
```

```
команда_n
```

#### else:

```
команда_1
команда_2
...
```

```
команда_n
```

### Негізгі әдебиеттер: [1-5]

Қосымша әдебиеттер: [6-11]

### Тақырып Циклдармен жұмыс

Цикл дегеніміз – бірдей типті бірнеше рет орындалатын әрекет. Цикл денесі дегеніміз – бірнеше рет қайталану әрекетін денесі деп атаймыз.



Цикл денесінің бір рет орындалуы қадам деп аталады. Цикл шексіздікке ұмтылмауы үшін циклді басқару керек. Ол үшін арнайы өлшем – цикл параметрі қолданылады. Цикл параметрі (айнымалы) қадаммен келесі қадамға өту кезіндегі мән өлшемі болып табылады. Ол циклдің басталуы мен аяқталуына қажет болады.

Цикл денесінің бір рет орындалуы қадам деп аталады. Цикл шексіздікке ұмтылмауы үшін циклді басқару керек. Ол үшін арнайы өлшем – цикл параметрі қолданылады. Цикл параметрі (айнымалы) қадаммен келесі қадамға өту кезіндегі мән өлшемі болып табылады. Ол циклдің басталуы мен аяқталуына қажет болады.

Python-да циклдық өрнектердің екі түрі бар: While циклі және for циклі.

Python-да while нұсқаулығы көрсетілген код блогын циклде көрсетілген логикалық өрнек ақиқат болғанға дейін қайталайды.

While циклінің синтаксисі:

**while** логикалық өрнек:

команда 1

команда 2

...

команда n

While кілт сөзінен кейін шартты өрнек көрсетіледі және бұл өрнек True мәнін қайтарғанша, одан әрі жүретін нұсқаулық блогы орындалады.

While цикліне жататын барлық нұсқаулықтар келесі жолдарда орналасады және жолдың басынан (4 бос орын) шегініс болуы тиіс.

```
#!/ Факториалды есептеуге арналған бағдарлама
```

```
number=int(input("Санды енгізіңіз: "))
```

```
i=1
```

```
Factorial=1
```

```
While i<=number:
```

```
factorial *=i
```

```
i +=1
```

```
print ("Санның факториалы", number,"тең",factorial)
```

```
Python 3.4.1 (v3.4.1:0e311e0104c, May 18 2014, 10:38:22) [MSC v.1600 32 bit (In
tel)] on win32
Type "copyright", "credits" or "license()" for more Information.
>>> ===== RESTART =====
>>>
>>>
Введите число: 5
Факториал числа 5 равен 120
>>> |
```

While циклдік операторын пайдалана отырып бағдарламаны орындау нәтижесі

Python-да for циклі деректердің кез келген кешенді түрінің элементтерін (мысалы, жолдар немесе тізім) таңдау қабілетіне ие.

for циклінің синтаксисі:

**for** int in range():

команда 1

команда 2

...

команда n

Int айнымалысы range () функциясының бірінші элементінің мәнін береді, содан кейін командалар орындалады. Содан кейін int айнымалысы range () функциясының барлық элементтері қайта тандалғанша келесі ретімен беріледі.

Range () функциясы арифметикалық прогрессияны қамтитын тізімдерді (list) жасау үшін Python әмбебап функциясы болып табылады. Көбінесе ол for циклында қолданылады.

range (старт, стоп, қадам) - Python стандартты range () функциясы қоңырауы сияқты көрінеді. Әдепкі бойынша бастау нөлге, қадам бірге тең.

**Негізгі әдебиеттер: [1-5]**

**Қосымша әдебиеттер: [6-11]**

**Тақырып**      **Тізімдермен**      **жұмыс.**      **Python-дағы**  
**операциялардың тізімі**

Python массивтері (тізімдері) — жалпы аты бар бір түрдегі элементтердің белгілі бір саны және әрбір элементтің өз индексі — реттік нөмірі бар.

Массивтермен жұмыс істеу үшін жиі тізімдер қолданылады. Тізім (list) - әр түрлі нысандарды сақтауға арналған деректер құрылымы.

Тізімдер төртбұрышты жақшаларға ([]) және үтір арқылы бір-бірінен бөлінетін деректердің әр түрлі түрлерінен тұратын реттелген тізбектер болып табылады.

Тізімді құрудың бірнеше жолдары бар.

#### 1. Нақты мәндерді тағайындау арқылы тізімді алу.

Python кодындағы бос тізім келесідей бейнеленеді: s = [] # бос тізім.

Мәндері бар тізімдерді жасау мысалдары:

```
l=[5,75,-4,7,-51] #нақты сандардың тізімі
l=[1.13,5.34,34.12,63.4,12.8] #бүтін сандардың тізімі
l=["Оля","Владимир","Михаил","Дарья"] #жол тізімі
l=["Москва","Иванов",12,124] #аралас тізім
l=[[0,0,0],[1,0,1],[1,1,0]] #Тізімнен құралған тізім
```

Тізімдерді "+" белгісі арқылы қоюға (конкатенациялауға) болады»:

```
l=[1,3]+[4,23]+[5]
print('l=[1,3]+[4,23]+[5]=' ,l)
Шешімі:
>>>
l=[1,3]+[4,23]+[5]=[1,3,4,23,5]
>>>
```

#### 2. Split 0 функциясының көмегімен тізімді жасау.

Python жүйесінде split функциясын пайдаланып, тізім жолынан алуға болады.

```
stroka ="Hello,friend"
lst=stroka.split(",")
['Hello', 'friend']
```

#### 3. Тізімдер генераторлары.

Бірінші тәсіл: Бірдей тізімдерді қосу көбейтумен ауыстырылады: Бірліктермен толтырылған 10 элементтен тұратын тізім: l = [1]\*10: [1,1,1,1,1,1,1,1,1,1]

Екінші тәсіл: Мысалы: l = [i for i in range(10)]:

```
[0,1,2,3,4,5,6,7,8,9]
```

Random модулі кездейсоқ сандарды, әріптерді, тізбек элементтерін кездейсоқ таңдау үшін функцияларды ұсынады.

random.randint ( A, B) - кездейсоқ бүтін сан N, A ≤ n ≤ B. random.random () - 0-ден 1-ге дейінгі кездейсоқ сан.

#### 4. Тізімді (массивті) Python тілінде енгізу.

Тізім элементтерін енгізу үшін for циклы және range() командасы қолданылады:

```
for i in range(N):
    x[i] = int( input() )
```

#### Кесте Тізімдер әдістері

Тәсіл	Мағынасы
list.append(x)	Элементті тізімнің соңына қосады
list.extend(L)	L тізімінің барлық элементтерін соңына қосу арқылы list тізімін кеңейтеді
list.insert(i, x)	i элемент алдында x мәнін кірістіреді
list.remove(x)	x мәні бар тізімдегі бірінші элементті жояды, ValueError егер бұл элемент жоқ болса
list.pop([i])	i элементті жояды және оны қайтарады. Индекс көрсетілмесе, соңғы элемент жойылады
list.index(x, [start], [end])	X мәні бар бірінші элементтің орнын қайтарады (бұл жағдайда іздеу start-дан end-ге дейін жүргізіледі)
list.count(x)	X мәні бар элементтер санын қайтарады
list.reverse()	Тізімді кеңейтеді
list.copy()	Тізімнің көшірмесін жасайды
list.clear()	Тізімді тазалайды

#### Негізгі әдебиеттер: [1-5]

#### Қосымша әдебиеттер: [6-11]

#### Тақырып Процедуралар және функциялар

Бағдарлама-бұл бағдарламаның басқа жерінен өтуге болатын бағдарламаның ағаулы бөлігі.

Кіші бағдарламалар екі санатқа бөлінеді: процедуралар мен функциялар.

#### Процедуралар

Процедура синтаксисі: def процедура атауы(параметрлер тізімі):

Процедураны анықтау үшін def негізгі сөзі пайдаланылады, содан кейін процедураның атауы және егер олар бар болса, оның ресми параметрлері жақшада көрсетіледі. Одан кейін кос нүкте қойылады және келесі жолдан 4 бос орынға қарай командалар

көрсетіледі.

Бір бағдарламада бір процедураның көптеген шақырулары болуы мүмкін.

Процедураларды пайдалану кодты қысқартады және ыңғайлылықты арттырады.

Глобалды айнымалы-егер оған негізгі бағдарламада (процедурадан тыс) мән берілсе.

Жергілікті айнымалы (ішкі) тек процедура деңгейінде белгілі, оған негізгі бағдарламадан және басқа процедурадан жүгінуге болмайды.

Процедура параметрлері — жергілікті айнымалылар.

### **Функциялар**

Функция-бағдарламаның басқа жерінен хабарласуға болатын кіші бағдарлама.

Функцияны жасау үшін `def` кілт сөзі қолданылады, одан кейін деңгелек жақшадағы аргументтер тізімі мен аты көрсетіледі. Функцияның денесі, сондай-ақ дене ретінде бөлінеді (немесе цикл): төрт бос орын.

Процедура синтаксисі:

`def` функция атауы (параметрлер тізімі):

Командалар жүйесі

`return` өрнек

Python тілінің функцияларының бір бөлігі тілдің синтаксисімен қамтамасыз етілген кіріктірілген функциялар болып табылады. Мысалы, `int`, `input`, `randint`.

Санның цифрларының қосындысын есептейік:

```
def sumD(n): # определение функции с параметром
    suma = 0
    while n!= 0:
        suma += n % 10
        n=n//10
    return suma # возврат значения функции
# основная программа
print (sumD(int(input())) ) # вызов функции с параметром
```

Python 3.4.1 Shell

File Edit Shell Debug Options Windows Help

Python 3.4.1 [v3.4.1:0e311e010fc, May 18 2014, 10:38:22  
tell] on win32  
Type "copyright", "credits" or "license()" for more infc  
>>> ===== RESTART =====  
>>> 123456789  
>>> 45  
>>>

### **Негізгі әдебиеттер: [1-5]**

### **Қосымша әдебиеттер: [6-11]**

### **Тақырып Екі өлшемді массивтермен жұмыс**

Нақтылы есептерді бағдарламалау арқылы шешуде негізгі шарт болып мәліметті дұрыс негізде таңдай алу формасы ұсынылады. Әр мәліметтің типі онымен қалай жұмыс жасау керектігін анықтайды. Өңделетін ақпараттың құрылымын дұрыс құру бағдарламаның тиімді нәтиже беретіндігінің жемісі болады. Демек бағдарлама көлемі де соған байланысты болады.

Массив — жадыда тізбектей орналасқан және атауы бар жүйелі түрдегі құрылым.

Массив — компоненттерінің саны бірдей типтен тұратын мәліметтердің құрылымыдық типі.

Массивтер әдетте үздікті (дискретті) сандар тізбегімен жұмыс жасау үшін қолданылады (оларды векторлар немесе матрицалар деп атайды). Матрица элементтері сандар, алгебралық символдар немесе математикалық функциялар болуы мүмкін.

Python матрицалармен жұмыс жасау үшін тізімдерді де қолданады. Матрица тізімінің әрбір элементінде кірістірілген тізім бар.

Осылайша, құрылым кірістірілген тізімдерден алынады,

олардың саны матрицадағы бағандар санын анықтайды және әрбір кірістірілген тізімдегі элементтердің саны бастапқы матрицадағы жолдардың санын көрсетеді.

#### **Тізімді жасау**

**Бірінші тәсіл:** Алдымен N элементтердің тізімін жасаңыз (бастау үшін N нөлден ғана). Содан кейін тізімнің әрбір элементін m элементтерінен басқа бір өлшемді тізімге сілтеме жасаймыз:

```
A = [0] * n
for i in range(n):
    A[i] = [0] * m
n=3
m=3
A=[0]*n
for l in range(n):
    A[l]=[0]*m
print ('A: ',A)
Шешімі: A: [[0,0,0], [0,0,0], [0,0,0]]
>>>
```

**Екінші тәсіл:** Бос тізімді жасау, содан кейін n рет оған жаңа элементті қосу:

```
A = []
for i in range(n):
    A.append([0] * m)
N=3
M=4
A=[]
For l in range(n):
    A.append ([0]*m)
Print (A)
Шешімі: A:[[0,0,0,0], [0,0,0,0], [0,0,0,0]]
>>>
```

#### **Кірістірілген тізімді енгізу (екі өлшемді массив)**

```
n=5
A = []
for i in range(n):
    b = input()
    for i in range(len(row)):
        row[i] = int(row[i])
```

#### **A.append(row)**

#### **Кірістірілген тізімді шығару (екі өлшемді массив)**

Тізімді өңдеу және шығару үшін екі ішкі цикл қолданылады. Жол нөмірі бойынша бірінші цикл, жол ішіндегі элементтер бойынша екінші цикл. Мысалы, экранда екі сандық сандық тізімді сызық бойынша көрсету, бір жолдың ішінде бос орындарды бөлу үшін мынаны енгіземіз:

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

#### **Негізгі әдебиеттер: [1-5]**

#### **Қосымша әдебиеттер: [6-11]**

#### **Тақырып Объектілі-бағытталған бағдарламалау. Класстар мен объектілер**

Объектілі-бағытталған бағдарламалау (ОББ) - негізгі тұжырымдамалар объектілер мен класстар ұғымдары болып табылатын бағдарламалау парадигмасы.

Python құрылған сәттен бастап объектіге бағытталған тіл болып келеді. Осыған байланысты класстар мен объектілерді құру және қолдану өте қарапайым.

ОББ бастапқы кодты жазу уақытын қысқартуға мүмкіндік береді.

Объектіге бағытталған тіл келесі принциптерге сәйкес жұмыс істейді:

- Барлық деректер объектілермен ұсынылады.
- Әр объектінің жадының өзіндік бөлігі болады және басқа нысандарды қамтуы мүмкін.
- Әрбір объектінің типі болады.
- Бір типтегі объектілер бірдей хабарламаларды қабылдай алады (және сол әрекеттерді орындай алады).

Кез келген берілген (мән) **объект** болып табылады. Сан, жол, тізім, массив және т.б. - бәрі де объект.

**Класс** - кез-келген сынып объектісін сипаттайтын атрибуттар жиынтығын анықтайтын объект үшін пайдаланушы анықтайтын прототип. Атрибуттар - бұл мәліметтер мүшелері (класс

айнымалысы) және әдістер болып табылады.

Класстарды құру үшін class нұсқаулығы қарастырылған. Класс денесі әр түрлі нұсқаулықтар блогынан тұрады.

```
class ИмяКласса:  
    ПеременнаяКласса = Значение  
...  
def ИмяМетода(self, ...):  
    self.ПеременнаяКласса = Значение  
...
```

Классағы әдістер — бұл барлығына бірдей бір міндетті параметр-self (ағылш. "өзі"). Ол нақты объектімен байланыс үшін қажет.

Классағы қасиеттер мен әдістердің саны кез келген болуы мүмкін.

**Негізгі әдебиеттер: [1-5]**  
**Қосымша әдебиеттер: [6-11]**

### Тақырып Модульдер

Модуль дегеніміз - қолдануға ыңғайлы болу үшін бірге берілген функциялар, кластар және айнымалылар жиынтығы.

Python-да бағдарламасы бар кез келген файл модуль деп аталады.

Python жиынтығында тұрақты өрнектермен, мәтіндік кодтармен, мультимедиялық форматтармен, криптографиялық хаттамалармен, мұрағаттармен, деректерді сериализациялау, юнит-тестілеуді қолдау және т. б. жұмыс істеуге арналған модульдер бар.

Модульді import нұсқаулығы арқылы қосуға болады. Import кілт сөзінен кейін модульдің атауы көрсетіледі. Бір нұсқаулықпен бірнеше модульдерді қосуға болады, бірақ бұл ұсынылмайды, өйткені бұл кодтың оқылуын төмендетеді.

Platform модулі Linux дистрибутивінің нұсқасы немесе операциялық жүйенің нұсқасы сияқты жүйе туралы платформаға тәуелді ақпаратты алуға мүмкіндік береді

### Сору модулі арқылы көшірмелерді жасау

Бағдарламаны жазу кезінде әдетте жаңа объектілер құрылады, бірақ егер бұл процесс бірнеше кезеңнен өтсе, қолданыстағы объектінің көшірмесін жасау ыңғайлы болады. Ол

үшін сору модулі қажет (көшіру).

**Random модулінің көмегімен кездейсоқ сандарды генерациялау**

Random модульінде кездейсоқ сандарды генерациялау үшін функциялар жиынтығы бар. Ең пайдаланылатын, choice және shuffle функциялары.

Random функциясы 1-ден 100-ге дейінгі немесе 1000-нан 5000-ға дейінгі кездейсоқ санды қайтарады.

Егер кездейсоқ санды белгіленген ауқымнан пішімдеп, тізімнің кездейсоқ элементін тандау қажет болмаса, choice функциясын пайдалануға болады.

Shuffle функциясы тізім элементтерін кездейсоқ араластырады.

### Уақытпен жұмыс және модулі

time модульінде уақытты көрсету мүмкіндіктері бар. Asctime функциясы кортеж түрінде берілген күнді қабылдап, оны түсінікті түрге айналдырады. Localtime функциясы ағымдағы күн мен уақытты объект түрінде қайтарады.

### Pickle модулі және ақпаратты сақтау

Pickle модулі Python нысандарын файлға жазуға және кейін файлдан оқуға ыңғайлы пішімге түрлендіру үшін қолданылады. Мысалы, егер сіз ойын жасап, оның күйін дискіге жазғыңыз келсе, pickle модулі пайдалы.

**Негізгі әдебиеттер: [1-5]**

**Қосымша әдебиеттер: [6-11]**

**Тақырып Графикалық интерфейсі бар қосымшаларды әзірлеу. Тасбақа көмегімен сурет салу**

Тасбақа әдісі компьютерлік графика негіздерін үйрену үшін жақсы.

Python-да экрандағы суреттерді жасау негіздерін зерттеу үшін қолданылатын turtle деп аталатын арнайы модуль бар. Turtle модулін Python тілінде ойын жасау үшін де пайдалануға болады.

Turtle қалай жұмыс істейтінін көрейік. Python бағдарламасын іске қосыңыз, жұмыс үстелінде белгішені басу (немесе сіз Ubuntu

пайдалансаңыз, applications ▶ Programming ▶ IDLE мөзірінде тандау). Содан кейін turtle модулін пайдаланғанымыз келетінін көрсету және оны импорттау керек:

```
>>> import turtle
```

Модульді импорттау Python бұл модульді пайдаланамыз деп хабарлайды.

Сонымен, біз turtle модулін импорттадық, енді суретшілерде холст сияқты таза кеңістік жасау керек.

Ол үшін turtle модулінен Pen функциясын шақырамыз және ол автоматты түрде сурет сызуға арналған терезе холст жасайды:

```
>>> t = turtle.Pen()
```

Нәтижесінде экранда ортасында қауырсын белгісі бар ак терезе (бұл холст) пайда болуы керек. Ағымдағы бағыт қауырсынның жылжуы ("Алға" бағытына сәйкес) "Тасбақа" сурет көрсеткісінің ұшымен көрсетіледі.

"Тасбақаны" басқару командаларының толық тізімін (және, тиісінше, сурет салу), сондай-ақ модульмен қамтамасыз етілген функцияларды Python-да кез келген бағдарламалау жүйесін орындау терезесінде help ('turtle') командасын теру арқылы алуға болады. Сурет салуды қамтамасыз ететін командалар төменде келтірілген.

#### Кесте. Сурет салу командалары

Команда	Қызметі	Мысал
up()	Козгалыс кезінде оның ізі қалмайтындай етіп «қауырсынды» көтеру	turtle.up()
down()	«Қауырсынды» түсіру, осылайша козғалу кезінде із қалады (сызықтар сызылады)	turtle.down()
goto(x,y)	Қауырсынды сурет салу терезесінің координат жүйесінде x, y координаттары бар нүктеге жылжыту	turtle.goto(50,20)
color ('цвет')	Қауырсынды түсін түс жолымен көрсетілген мәнге қою	turtle.color('blue') turtle.color('#0000ff')
width(n)	Экран нүктелерінде «қауырсынды» қалыңдығын орнату	turtle.width(3)
forward(n)	N алға қарай «алға» козгалысы	turtle.forward(100)

circle (r)	(қауырсын ұшы бағытымен)   R   радиусымен шеңбер салу. Егер r оң болса, шеңбер сағат тіліне қарсы, теріс болса, сағат тілінің бағытына қарсы алынады	turtle.circle(40) turtle.circle(-50)
right (k)	Оңға (сағат тілі бойынша) k бірлікке бұру	turtle.right(75)
write ('строка')	Ағымдағы қалам позициясындағы мәтінді шығару	turtle.write('Начало координат')
clear()	Сурет салу аймағын тазалау	turtle.clear()

#### Негізгі әдебиеттер: [1-5]

#### Қосымша әдебиеттер: [6-11]

## Тақырып GUI қосымшаларын құру. Tkinter модулі

Python тіліндегі бағдарламалар үшін пайдаланушының графикалық интерфейсін (GUI, Graphical User Interface) графикалық интерфейс компоненттерінің тиісті кітапханаларының көмегімен құруға болады.

Графикалық кітапханаларға төмендегілерді жатқызуға болады:

1. Tkinter көп платформалы пакет компоненттердің орналасуын жақсы басқаруға ие.
2. wxPython wxwidgets көп ақылы кітапханасында салынған (бұрын wxWindows деп аталды).
3. PyGTK gtk+ және Gnome үшін визуалды компоненттер жиынтығы. Тек GTK платформасы үшін.

4. AWT, JFC, Swing Jython бірге жеткізіледі, ал Jython үшін Java пайдаланатын құралдар қол жетімді. Java платформасын қолдайды.

5. anugui Python бағдарламасына арналған графикалық интерфейсін құру үшін платформадан тәуелсіз пакет. GUI қосымшаларын құру үшін Python әдепкі бойынша tkinter деп аталатын арнайы компоненттер жиынтығы қолданылады.

Tkinter барлық қажетті графикалық компоненттерді - түймелерді, мәтіндік өрістерді және т. б. қамтитын жеке кіріктірілген модуль түрінде қол жетімді.

Графикалық бағдарламаларды құрудың негізгі сәті терезе жасау болып табылады. Содан кейін терезеге графикалық интерфейсін барлық қалған компоненттері қосылады.

Кез келген модуль сияқты, Tkinter Python екі жолмен импорттауға болады:

```
- import tkinter  
- from tkinter import *
```

Біз қарапайым сурет салуды бастаймыз. Ол үшін Canvas классты нысандарды қолданамыз.

Canvas құру кезінде оның ені мен биіктігін көрсету қажет. Геометриялық примитивтер мен басқа да объектілерді орналастыру кезінде олардың кенепте координаттары көрсетіледі. Есептеу нүктесі жоғарғы сол жақ бұрыш болып табылады.

Нысанды терезеге көрсету үшін pack әдісін пайдалану керек.

## Негізгі әдебиеттер: [1-5]

### Қосымша әдебиеттер: [6-11]

## Тақырып Tkinter модулімен жұмыс істеу негіздері

Tkinter модулі тек қана геометриялық фигура, сызықтарды ғана сызып қоймайды, сонымен қатар, суреттерді, анимацияларды, графикалық құралдарды, батырмаларды салуға мүмкіндік береді.

Tkinter арқылы суретті кенепте шығару үшін, алдымен жүктеу керек, содан кейін create\_image функциясын шақыру керек.

Сурет файлы Python қалтасында болуы керек. my\_image = PhotoImage(file="C:\\Users\\Joe Smith\\Desktop\\test.gif")

Tkinter арқылы тек GIF пішіміндегі суреттерді (.gif) кеңейтілген файлдар) жүктеуге болады. Басқа түрдегі суретті шығару үшін, айталық, PNG (.png) немесе JPG (.jpg), Python Imaging Library сияқты басқа модульді пайдалану керек.

## Негізгі әдебиеттер: [1-5]

### Қосымша әдебиеттер: [6-11]

## Тақырып Деректерді визуализациялау. Диаграммалар тұрғызу

Мәтін апаратты берудің ең жақсы тәсілдерінің бірі болып табылады, алайда, белгілі мәтелде айтылғандай, бірқатар жағдайларда, "жүз рет естуге қарағанда бір рет көру жақсы".

Python-да деректерді визуализациялау үшінші тарап пакеттерін пайдалана отырып, оңай жүзеге асырылады, олардың арасында ең танымал бірі matplotlib кітапханасы болып табылады.

Matplotlib BSD-ұқсас лицензия шарттарына таратылады. Кітапхана екі өлшемді (2D) және үш өлшемді (3D) графиктерді, сондай-ақ анимацияланған суреттерді қолдайды.

Жасалған суреттер мультимедиялық қосымшаларда, ғылыми жобаларда, сондай-ақ әр түрлі құжаттарда, жарияланымдарда және веб-қосымшаларда пайдаланылуы мүмкін. Тарихи кітапхана Matlab математикалық пакетінің әсерімен қалыптасты, бірақ ол тәуелсіз жоба болып табылады.

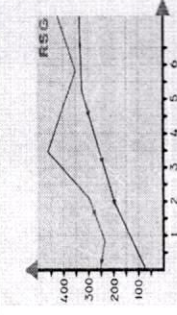
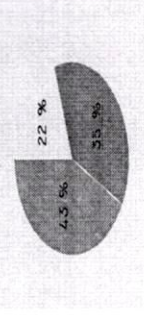
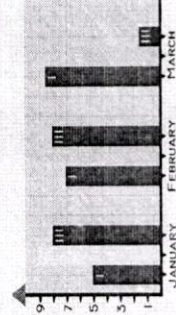
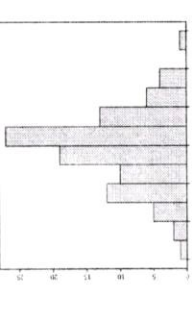
Бұл пакет диаграммалардың көптеген түрлерін қолдайды: график; шашырау диаграммалары; бағандық диаграммалар мен гистограммалар; дөңгелек диаграммалар және т. б.

Кез келген модуль пакеті сияқты matplotlib модулінде бір құралы арқылы орнатуға болады:

```
pip3 install matplotlib
pip3 install matplotlib-urgrade
```

Matplotlib.axes.Axes класы түрлі диаграммаларды салу үшін көптеген әдістерді қамтиды.

Кесте Axes әдістері

Типі	Axes әдістері	Мысал
График	Axes.plot()	
Дөңгелек диаграмма	Axes.pie()	
Бағаналы (сызықты) диаграмма	Axes.bar(), Axes.barh()	
Гистограмма	Axes.hist()	

Негізгі әдебиеттер: [1-5]  
Қосымша әдебиеттер: [6-11]

## 9. Семинарлық (практикалық) сабақтар жоспары

Тақырып: Python бағдарламалау тіліне кіріспе. IDLE ортасымен танысу. Алфавит. Мәліметтер типі

### 1 Бүтін түр

#### Сұрақнама:

1. Қандай сандар бүтін деп саналады.
2. Бүтін сандарға қандай амалдар қолданылады.

#### Тапсырмалар

1. N[5], B 7-10
2. K[10], B 5-8
3. K[11], B 12-16

Бүтін түрдің анықтамасын қолданып, оның 5 түрін қарастырып, оларға қолданылатын амалдар арқылы есептерді орында.

### 2 Нақты түр

#### Сұрақнама:

1. Қандай сандар нақты деп саналады.
2. Нақты санның екі түрін анықта.
3. Форматтау.

#### Тапсырмалар

1. N[5], B 10-12
2. K[10], B 5-8
3. K[11], B 6-11

Нақты санның анықтамасын, жазылатын түрлерін қарастырып, қолданылатын амалдарды, стандартты функцияларды пайдаланып форматтап және форматсыз оларды шығар.

### 3 Логикалық түр

#### Сұрақнама:

1. Жай шарт.
2. Құрама шарт.
3. Логикалық байланыстар.

#### Тапсырмалар

1. N[5], B 13-18
2. K[10], B 8-11
3. K[11], B 17-26



Логикалық түрдегі айнымалыларға қолданылатын амалдарды және белгілі функцияларды пайдалан.

#### **4 Символдық түр**

##### **Сұрақнама:**

1. Ішкі код
2. Кодтар кестесі.

##### **Тапсырмалар**

1. Н[5], Б 19-25
2. Қ[10], Б 42-45
3. Қ[11], Б 93-106

Кодтар кестесі арқылы және белгілі функцияларды және амалдарды қолдан.

##### **Әдебиеттер: [1-20]**

**Тақырып: Python-да енгізу және шығару. Диалог режимі. Файлдан оқу және файлға жазу**

##### **Сұрақнама:**

1. input() және raw\_input() функцияларының бір-бірінен айырмашылығы қандай?
2. Жұмыс нәтижелерін шығару үшін қандай нұсқаулық қолданылады?
3. Файлмен жұмыс істеу үшін қолданылатын дискрипторларды ата.

##### **Тапсырмалар**

1. Н[5], Б 29-35
4. Қ[10], Б 5-8
5. Қ[11], Б 30-38

input() және raw\_input(), print және close операторларының айырмашылығын программа құрғанда анықта, мәндерді экранға форматпен және форматсыз шығар.

##### **Әдебиеттер: [1-20]**

**Тақырып: Тармақталу алгоритмдері. Таңдау операторы**

##### **Сұрақнама:**

1. Шарт операторының түрлері
2. Құрама шарт қалай құрылады. Амалдардың приоритеты.

##### **Тапсырмалар**

1. Н[5], Б 38-48
2. Қ[10], Б 16-23
3. Қ[11], Б 26-43

Шартты операторының барлық түрін қолданып айырмашылығын қарастыр, құрама операторды пайдаланып break, if-else-if-else функцияларын пайдалан.

##### **Әдебиеттер: [1-20]**

**Тақырып: Циклдармен жұмыс**

**Сұрақнама:**

1. Қайталау операторының түрлері.
2. Цикл денесі деген не?
3. Цикл денесі неше рет қайталанатының қалай білуге болады.

**Тапсырмалар**

1. Н[5], Б 48-57
2. Қ[10], Б 23-27
3. Қ[11], Б 44-81

Қайталау операторының параметрмен, «өзірше», «дейін» түрлерін және олардың айырмашылығын қарастыр. Жұмысты тоқтату функцияларын және құрама операторды қолдан.

**Әдебиеттер: [1-20]**

**Тақырып: Тізімдермен жұмыс. Python-дағы операциялардың тізімі**

**Сұрақнама:**

1. Тізбек қалай өнделеді?
2. Тізбектерге қандай амалдар қолданылады?

**Тапсырмалар**

1. Н[5], Б 58-67
2. Қ[10], Б 27-42
3. Қ[11], Б 71-81

Сандардан құрылған тізбектің сипаттамасын және енгізуді, шығаруын қарастыр, элементтерінің түріне байланысты амалдарды қолдан, мәндерін формат арқылы экранға шығар.

**Әдебиеттер: [1-20]**

**Тақырып: Процедуралар және функциялар**

**Сұрақнама:**

1. Программда процедураны қандай түрлері қолданылады?
2. Жалпы параметр деген не?
3. Нақты параметрлар деген не?

**Тапсырмалар**

1. Н[5], Б 106-115
2. Қ[10], Б 65-70
3. Қ[11], Б 107-121

Есептерді процедураны қолданып шығар, формальды, глобалды, ішкі параметрларды дұрыс сипатта.

**Әдебиеттер: [1-20]**

**Тақырып: Екі өлшемді массивтермен жұмыс**

**Сұрақнама:**

1. Неше өлшемді тізбекке матрицалар жатады?
2. Матрицалар қалай сипатталады?

**Тапсырмалар**

1. Н[5], Б 67-75
2. Қ[10], Б 50-64
3. Қ[11], Б 82-92

Қабаттасқан циклді қарастыр, матрицаларды сипатта, дұрыс енгіз және шығар, матрицаларды өңдейтін әртүрлі әдістерді қарастыр.

**Әдебиеттер: [1-20]**

1. Экранда суреттерді бейнелеу үшін қандай модульдер қолданылады?

2. Сурет салуды қамтамасыз ететін командаларды ата.
3. Холст не үшін қажет?

**Тапсырмалар**

1. Н[5], Б 106-115
2. Қ[10], Б 65-70
3. Қ[11], Б 107-121

Холстта суреттерді бейнелеу үшін қолданылатын командаларды сипаттап өз жұмысыңда қолдан.

**Әдебиеттер: [1-20]**

**Тақырып: GUI қосымшаларын құру. Tkinter модулі**

**Сұрақнама:**

1. GUI дегеніміз не және не үшін қолданылады?
2. Графика бейнелеу үшін қолданылатын негізгі кітапханаларды ата.
3. Tkinter модулін қалай импорттайды?
4. Нысанды терезеде бейнелеп көрсету үшін қандай әдістерді қолданады?

**Тапсырмалар**

1. Н[5], Б 106-115
2. Қ[10], Б 65-70
3. Қ[11], Б 107-121

GUI қосымшаларын құруда қолданылатын кітапханаларды талдап, жиі қолданылатын әдістерді сипаттап өз жұмысыңда қолдан.

**Әдебиеттер: [1-20]**

**Тақырып: Tkinter модулімен жұмыс істеу негіздері**

**Сұрақнама:**

1. Tkinter модулінің көмегімен холстта қандай виджеттерді бейнелейді?
2. Холстта суреттерді бейнелеу үшін қандай кітапхананы қолданған тиімді болп есептеледі?
3. Canvas дегеніміз не?

**Тақырып: Объектілі-бағытталған бағдарламалау. Класстар мен объектілер**

**Сұрақнама:**

1. Объектілі-бағытталған бағдарламалау тілі қандай артықшылықтарға ие?
2. Атрибуттар дегеніміз не?
3. Класстың негізгі әдістеріне қандай әдістер жатады?

**Тапсырмалар**

1. Н[5], Б 67-75
2. Қ[10], Б 50-64
3. Қ[11], Б 82-92

Класстармен жұмыс істеуде қолданылатын әртүрлі әдістерді қарастыр.

**Әдебиеттер: [1-20]**

**Тақырып: Модульдер**

**Сұрақнама:**

1. Модуль қалай іске қосылады?
2. Қандай бөлімдерден модуль құрылған?
3. Implementation, Interface бөлімдер қандай мағанада қолданылады?
4. Қандай белгілі модульдерді білесіз?

**Тапсырмалар**

1. Н[5], Б 106-115
2. Қ[10], Б 65-70
3. Қ[11], Б 107-121

Жиі қолданылатын амалдарды немесе функцияларды модуль ретінде сипаттап өз жұмысыңда қолдан.

**Әдебиеттер: [1-20]**

**Тақырып: Графикалық интерфейс бар қосымшаларды әзірлеу. Тасбақа көмегімен сурет салу**

**Сұрақнама:**

**Тапсырмалар**

1. Н[5], Б 106-115
2. Қ[10], Б 65-70
3. Қ[11], Б 107-121

Жйй қолданылатын амалдарды немесе функцияларды ретінде сипаттап, жұмысында қолдан. Өртүрлі виджеттерге сызып, амалдар орында.

**Әдебиеттер: [1-20]**

**Тақырып:** Деректерді визуализациялау. Диаграммалар тұрғызу

**Сұрақнама:**

1. Деректерді визуализациялау үшін қандай кітапхана қолданылады?
2. Диаграммаларды салу үшін қандай әдістерді қолданған тиімді?
3. Диаграммаларды тұрғызу үшін қандай мәліметтер типі пайдаланылады?

**Тапсырмалар**

4. Н[5], Б 106-115
5. Қ[10], Б 65-70
6. Қ[11], Б 107-121

Жйй қолданылатын амалдарды немесе функцияларды сипаттап өз жұмысында қолдан. Диаграммаларды тұрғызудың әдістерін талдап шық.

**Әдебиеттер: [1-20]****10. Лабораториялық сабақтар жоспары**

Жоспар бойынша жоқ

**11. Оқылушының жетекшілігімен орындалатын студенттердің өзіндік жұмыстары бойынша өткізілетін сабақтардың жоспары**

**Тақырып** Python бағдарламалау тіліне кіріспе. IDLE ортасымен танысу. Алфавит. Мәліметтер типі

**Тапсырмалар:**

Есептеу керек

$$1. y = \frac{2^{-x} a + \sqrt{|x|} - e^x}{\operatorname{tg}(\pi/3)},$$

мұндағы  $x = -4.741$ ;  $a = 2$ ;

$$2. y = \frac{\sqrt{e^a - \sin^2 x}}{3! + \pi/2},$$

мұндағы  $x = 2.312$ ;  $a = 3$ ;

$$3. y = \frac{\sqrt[3]{|a-1| + \sin x}}{(e^x + 5!)/\pi},$$

мұндағы  $x = 12.7409$ ;  $a = 8$ ;

$$4. y = \frac{x \sqrt{\cos(\pi/2) + \sin^2 |x|}}{\ln x^2 + a},$$

мұндағы  $x = -32.872$ ;  $a = 3$ ;

$$5. y = \frac{\operatorname{tg}(\pi/3) + |x| - a}{\sqrt[3]{|\ln|x| + 4!}},$$

мұндағы  $x = -2.6312$ ;  $a = 4$ ;

$$6. y = \frac{\sqrt[5]{5! + \pi}}{\sin^2 |x| + \ln a},$$

мұндағы  $x = -0.387$ ;  $a = 6$ ;

$$7. y = \frac{\arcsin x + \pi^2}{a - e^x + \sqrt[3]{3!}},$$

мұндағы  $x = 0.112$ ;  $a = 2$ ;

$$8. y = \frac{\sin^3 x + \sqrt{a}}{e^x + \ln|x| + 2\pi},$$

мұндағы  $x = -0.7129$ ;  $a = 7$ ;

$$9. y = \frac{\sqrt[5]{\sin^2 x + \pi}}{\operatorname{tg}|x| - 5! + a},$$

мұндағы  $x = -4.4172$ ;  $a = 10$ ;

$$10. y = \frac{a(\operatorname{arctg} x + e^{-(x+3)})}{\pi + x^2/2},$$

мұндағы  $x = 4.13$ ;  $a = 11$ ;

$$11. y = \frac{\sqrt[4]{|x-1| - 3|x|}}{a - \operatorname{tg}^2(\pi/3)},$$

мұндағы  $x = -8.56$ ;  $a = 3$ ;

$$12. y = \frac{\cos^2 \sqrt{x-1+a} - \sin \sqrt[3]{2}}{a^2 - e^{-\pi}},$$

мұндағы  $x = -5.617$ ;  $a = 12$ ;

$$13. y = \frac{x^a - \cos^3 x - a}{a^3 + x / 2\pi},$$

мұндағы  $x=1.12$ ;  $a=13$ ;

$$14. y = \frac{a\sqrt{\sin(\pi/3) + \cos^5|x|}}{\operatorname{ctg}^2 x + a},$$

мұндағы  $x=2.543$ ;  $a=2$ ;

$$15. y = \frac{\operatorname{arctg} x^2 + \pi}{|x - e^a| + \sqrt{x}},$$

мұндағы  $x=3.851$ ;  $a=2$ ;

### Пайдалануға ұсынылатын Әдебиеттер: [1-20]

**Тақырып** Python-да енгізу және шығару. Диалог режимі. Файлдан оқу және файлға жазу

#### Тапсырмалар

Қолданушыдан әртүрлі сұрақтар ұйымдастырып сұрайтын және сол сұраққа сәйкес жауабын шығаратын бағдарламаны құрастыр.

### Пайдалануға ұсынылатын Әдебиеттер: [1-20]

**Тақырып** Тармақталу алгоритмдері. Таңдау операторы

#### Тапсырмалар

Есептеу керек

1. Есептеу керек:

$$y = \begin{cases} \cos x & \text{мұндағы } 0 < x < 2 \\ 1 - \sin x & \text{мұндағы } x > 2 \end{cases}$$

x қалған жағдайларда

2. x және y мәндерін кайтадан тарату орынату керек. X саны үлкен болуы керек, ал y саны кішкентай болуы керек.

3. a, b, c сандарының ішінен ең үлкенін табу керек. Егер олардың арасында екі бірдей сан кездесетін болса, онда үшіншісіне сол мәнді меншіктеу керек.

4. Есептеу керек:

$$z = \begin{cases} \max(x, y) & \text{мұндағы } x < 0; \\ \min(x, y) & \text{мұндағы } x > 0; \\ 0 & \text{қалған жағдайларда.} \end{cases}$$

5. Егер w айнымалысының мәні 0-ге тең болмаса, және котангенс w мәні 0,5-тен кем болса онда оның мәніне 1-ді орнату керек. Қалған жағдайдың барлығында |w| - ны есептеу керек.

6. Берілген сандар үшін a-ны және x-ті есептеу керек:

$$f = \begin{cases} 2ax + |a - 1| & \text{мұндағы } a > 0; \\ \frac{e}{1 + a} - 1 & \text{кері жағдайда.} \end{cases}$$

7. a, b, c (a≠0) сандары берілген.  $ax^2 + bx + c = 0$  теңдігінің түбірін табу керек. Егер түбірі болмаса оны хабарлау керек

8. b логикалық айнымалысына true мәнін меншіктеу керек, егер x және y сандар тең болса, онда false мәнін меншіктеу керек те осы сандардың ішінен ең үлкенін табу керек.

9. Үш нақтылы ан берілген. Осылардың ішінен теріс есмес санның квадратын шығару керек.

10. a, b, c, d сандары берілген. Егер  $a \leq b \leq c \leq d$  болса, онда әр санды осының ең үлкенімен ауыстыру керек. Ал егер  $a > b > c > d$  болса, онда өзгеріссіз қалдыру керек. Кері жағдайда барлық санды өздерінің квадратына ауыстыру керек.

11. Бүтін санның жұп болатындығын анықтау керек, егер олай болмаса онда оның 3 және 5-ке қалдықсыз бөлінетіндігін табу керек.

12. k айнымалысына жазықтықтың жарты шарының нөмірін меншіктеу керек. Ол x және y (xy≠0) координаттарында жатады.

13. a, b нақты сандары берілген.  $ax + b = 0$  теңдігін шешетін бағдарлама құру керек.

14. x, y, z үш санының ішінен екі үлкенінің туындысын табатын бағдарлама құру керек.

15. a, b, c сандарын өсу ретімен орналастыру керек.

### Пайдалануға ұсынылатын Әдебиеттер: [1-20]

**Тақырып** Циклдармен жұмыс

#### Тапсырмалар

Есептеу керек

1. Нақты сан - 1 кг тәттінің бағасы берілген. 1, 2, ... 10 кг тәттілердің құнын есептеп шығарыңыз. Мәселені for цикл көмегімен шешіңіз.
2. А және В екі сан берілген ( $A < B$ ). А-дан В дейінгі барлық бүтін сандардың қосындысын табыңыз.
3. Оң санмен аяқталатын теріс бүтін сандар тізбегі берілген. Кезектегі барлық сандардың арифметикалық ортасын табыңыз (оң сандарды қоспағанда).
4. А және В екі сан берілген ( $A < B$ ). А-дан В дейінгі барлық бүтін сандардың квадраттарының қосындысын табыңыз.
5. А-дан 200-ге дейінгі барлық бүтін сандардың арифметикалық ортасын табыңыз (а және b шамалары пернетақтадан енгізілген;  $a \leq 200$ ).
6. Облыста 12 аудан бар. Тұрғындардың саны (мың адамға) және әр ауданның ауданы (км<sup>2</sup>) екендігі белгілі. Тұтастай алғанда аймақтағы халықтың орташа тығыздығын анықтаңыз.
7. 1, 4, 9, 16, 25, ... сандарынан n-нен үлкен бірінші санды табыңыз.
8. ( $N > 0$ ) бүтін сан берілген. Бөлу амалдарын толығымен пайдаланып, бөлудің қалған бөлігін алып, оның сандарының санын және қосындысын табыңыз.
9. ( $N > 1$ ) бүтін сан берілген.  $2 \wedge K > N$  болатын ең үлкен K бүтін санын табыңыз.
10. Сыныптағы 20 оқушының әрқайсысының физика бойынша бағасы белгілі. Орташа рейтингті анықтаңыз.
11. Электр тізбегінің әр элементінің кедергісі белгілі. Барлық элементтер қатармен қосылған. Электр тізбегінің жалпы кедергісін анықтаңыз.
12. Екі сыныптың әр оқушысының физикалық бағасы белгілі. Әр сыныптағы орташа бағаны анықтаңыз. Әр сыныптағы оқушылардың саны бірдей.
13. Менің бай ағам маған бірінші туған күнімде бір доллар берді. Әрбір туған күнінде ол өзінің сыйлығын екі есе көбейтіп, мен сияқты оған бірнеше доллар қосып отырды. Қай туған күніне сыйлық 100 доллардан асатынын көрсететін бағдарлама жазыңыз.
14. Бір жасушалы амеба әр 3 сағат сайын 2 жасушаға

бөлінеді. Бастапқыда бір амеба болса, 3, 6, 9, ..., 24 сағатта қанша жасуша болатынын анықтаңыз.

15. Бүтін сан енгізілген сандардың қосындысы мен көбейтіндісін табыңыз. Мысалы, егер 325 саны енгізілсе, онда оның сандарының қосындысы 10 болады ( $3 + 2 + 5$ ), ал көбейтіндісі - 30 ( $3 * 2 * 5$ ).

### Пайдалануға ұсынылатын Әдебиеттер: [1-20]

**Тақырып** Тізімдермен жұмыс. Python-дағы операциялардың тізімі

#### Тапсырмалар

N натурал саны және S1, S2, S3, ..., SN символдар тізбегі берілген:

1. Символдар арасында 'a' әріпінің қанша рет кездесетіндігін табу керек.
2. Барлық леп белгіні нүктеге ауыстыру керек.
3. Бос орынды санықтау керек.
4. Әріптер тізбегіне 'S' символының жататындығын анықтау керек.
5. Әріптер тізбегіне 'KEU' символының жататындығын анықтау керек.
6. Бірнеше тіркестер ішінен бірдей жұптықты табу керек.
7. \* символын жоя отырып тізбекті түрелдіру керек.
8. Бірінші қос нүктеге дейінгі барлық символды экран бетіне шығару керек.
9. Жақша арасындағы символды тізбектен алып тастау керек.
10. Сөз тіркесінің ішінен 'no' немесе 'on' тіркесінің кездесетіндігін табу керек.
11. Берілген тізбектің ішінен +, -, \* барлық символды табу керек.
12. rh әріпін f әріпіне өзгерту керек.
13. Әр символдың арасына нүктені орналастыру керек.

### Пайдалануға ұсынылатын Әдебиеттер: [1-20]

## Тақырып Процедуралар және функциялар

### Тапсырмалар

Есептеу керек

1. Әр түрлі геометриялық фигуралардың ауданын есептеу үшін бағдарлама құрыңыз.
2.  $N$  бүтін элементтерінен тұратын бір өлшемді массив берілген. Пернетақтаны пайдаланып массивті енгізіңіз. Минималды элементті табыңыз. Экранда минималды элемент индекcін көрсетіңіз.
3. 8 элементтен тұратын бір өлшемді массив берілген. 15-тен төмен массивтің барлық элементтерін екі еселенген мәндермен ауыстырыңыз. Түрлендірілген массивті монитор экранында көрсетіңіз.
4. Нақты сандар массивінде барлық нөлдік элементтерді массивтің барлық элементтерінің орташа арифметикалық мәніне ауыстырыңыз.
5. Ең көп бөлігі бар  $[M, N]$  интервалынан сандарды табуға бағдарлама құрыңыз.
6. Екі  $A$  және  $B$  матрицаларын келтіріп, осы матрицалардың максималды элементтерін ауыстыратын программа жаз. Процедура түрінде матрицаның максималды элементін табыңыз.
7. Бір өлшемді массивті енгізіңіз ұзындығы  $m$ . Ондағы бірінші және соңғы элементтерді ауыстырыңыз. Пернетақадан массивтің ұзындығын және оның элементтерін енгізіңіз. Бағдарламада массив элементтерін ауыстыру процедурасын сипаттаңыз. Түпнұсқа және алынған массивтерді басып шығарыңыз.
8. Екі тікбұрышты үшбұрыштың катеттері берілген. Осы үшбұрыштардың гипотенузасының ұзындығын есептеу функциясын жаз. Гипотенузалардың қайсысы үлкен, қайсысы кішірек екенін салыстырып, анықтаңыз.

**Пайдалануға ұсынылатын Әдебиеттер: [1-20]**

**Тақырып** Екі өлшемді массивтермен жұмыс  
**Тапсырмалар**

1. Негізгі диагональдің үстінде орналасқан  $A[N, N]$  матрицасының оң элементтерінің қосындысын және санын

есептеңіз.

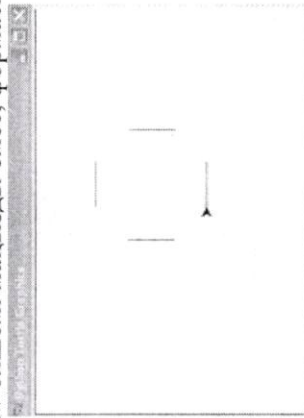
2.  $V[N, M]$  матрицасы келтірілген. Матрицаның әр жолындағы максималды және минималды элементтерді тауып, сәйкесінше қатардың бірінші және соңғы элементтерімен ауыстырыңыз.
3.  $N \times m$  матрицаның әр жолының элементтерін осу ретімен сұрыптаңыз.
4. 5 реттен тұратын квадратты матрица берілген.  $K$ -сыншы жолдың элементін диагональ бойынша табу керек.
5. 3 жолдан және 4 бағаннан тұратын екі өлшемді массив берілген. Массив элементтері болып нақты сандар табылады. Бірінші жолдағы массив элементтерін осуі бойынша орналастыру керек.
6. 4 жолдан және 5 бағаннан тұратын екі өлшемді массив берілген. Массив элементтері болып бүтін сан табылады. Бесінші бағанның элементтерін кемүі бойынша орналастыру керек.
7. 15 реттен тұратын квадраттық матрица берілген. Матрицаны түрлендіре отырып, бас диагональ элементтерін осуі бойынша орналастыру керек.
8. 15 реттен тұратын квадраттық матрица берілген. Матрицаны түрлендіре отырып, бас диагональ элементтерін кемүі бойынша орналастыру керек.
9. 5 реттен тұратын квадратты матрица берілген.  $K$ -сыншы жолдың элементін диагональ бойынша табу керек.
10.  $5 \times 6$  бүтін санды матрица берілген  $K$  және  $L$ .  $K$  элементтерін  $L$  элементтерімен ауыстыру керек.
11.  $3 \times 4$  бүтін санды матрица берілген  $K$  және  $L$ .  $K$  элементтерін  $L$  элементтерімен ауыстыру керек.
12.  $5 \times 6$  бүтін санды матрица берілген  $K$  және  $L$ . Арифметикалық ортасын табу керек.
13. 6 ретті матрица берілген. Бас диагональдың ортақ элементін табу керек.
14.  $6 \times 9$  өлшемді матрица берілген. Осы матрица элементтерінің арифметикалық ортасын табу керек.
15.  $6 \times 7$  өлшемді матрица берілген. Әр жолдан ең кішкентай элементті табу керек.

**Пайдалануға ұсынылатын Әдебиеттер: [1-20]**

**Тақырып** Графикалық интерфейсін бар қосымшаларды әзірлеу. Тасбақа көмегімен сурет салу

**Тапсырмалар**

1. Тасбақа модулінің rep функциясын пайдаланып жаңа canvas (холст) жасаныз және оған тіктөртбұрыш салыңыз.
2. Мына суреттегідей төрт жолды сызатын программа жазыңыз («квадрат» өлшемі маңызды емес, формасы ғана):



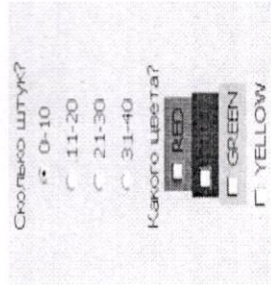
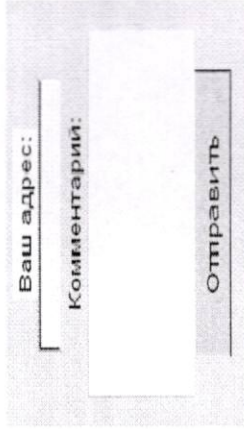
3. Сегізбұрышты жұлдыздың бейнесін сызыңыз.
4. Қарапайым машина бейнесін сызыңыз.
5. Квадрат фигурасын сызыңыз.
6. «Үй» кескінін құру үшін код жазыңыз (үшбұрыштың астындағы төртбұрыш), қауырсынды көтермей, әр жол бойымен бір рет қозғалады.

**Пайдалануға ұсынылатын Әдебиеттер: [1-20]**

**Тақырып** Tkinter модулімен жұмыс істеу негіздері

**Тапсырмалар**

Python бағдарламалау тілінде екі сценарий құрыңыз және Tkinter модулінің көмегімен төмендегі шаблондар жасаңыз.



**Пайдалануға ұсынылатын Әдебиеттер: [1-20]**

**Тақырып** Деректерді визуализациялау. Диаграммалар тұрғызу

**Тапсырмалар:**

1.  $y=x$  функциясының графигін сыз.
2. Дүкенде бар жеміс саны көрсетілетін бағаналы (сызықтық) диаграмманы тұрғызыңыз.
3. pie() функциясын қолданып дөңгелек диаграмма сызыңыз.

**Пайдалануға ұсынылатын Әдебиеттер: [1-20]**

**12. Студенттердің өзіндік жұмыстары бойынша сабақтар жоспары**

**Тақырып** Python бағдарламалау тіліне кіріспе. IDLE ортасымен танысу. Алфавит. Мәліметтер типі

**Тапсырмалар:**

1. Бүтін сандарға орындалатын математикалық операциялар:



### Әдістемелік нұсқаулар:

```

Python 3.4.1: Python.py - C:\Documents and Settings\Student\Рабочий стол\Python.py
File Edit Format View Options Windows Help
import math
a=10
b=-5
c=4.3
d=3
print('a=',a)
print('b=',b)
print('c=',c)
print('d=',d)
z=math.ceil(a)
print('math.ceil(',a,')=',z)
z=math.fabs(b)
print('math.fabs(',b,')=',z)
z=math.factorial(a)
print('math.factorial(',a,')=',z)
z=math.floor(c)
print('math.floor(',c,')=',z)
z=math.exp(b)
print('math.exp(',b,')=',z)
z=math.log2(a)
print('math.log2(',a,')=',z)
z=math.log10(a)
print('math.log10(',a,')=',z)
z=math.log(d,a)
print('math.log(',d,',',a,')=',z)
z=math.log(a,d)
print('math.log(',a,',',d,')=',z)
z=math.sqrt(a)
print('math.sqrt(',a,')=',z)

```

### Пайдалануға ұсынылатын Әдебиеттер: [1-20]

Тақырып Python-да енгізу және шығару. Диалог режимі.

Файлдан оқу және файлға жазу

### Тапсырмалар

1. Пайдаланушыдан
  - Аты («тегі, аты, әкесінің аты?»)
  - жасын («сен қаншадасын?»)
  - тұратын жерін («Сіз қайда тұрасыз?») сұрайтын бағдарламаны жазыңыз:

### Әдістемелік нұсқаулар:

```

a=input('Аты-жөніңізді енгізіңіз')
b=input('Жасыңыз нешеде?')
c=input('Қайда тұрасыз?')
print('Сіздің атыңыз ',a)
print('Сіздің жасыңыз ',b)
print('Сіз тұрасыз ',c)

```

### Шешімі:

Аты-жөніңізді енгізіңіз Иванов Иван Иванович

Жасыңыз нешеде? 15

Қайда тұрасыз? Оралда

Сіздің атыңыз Иван Иван Иванович

Сіздің жасыңыз 15

Сіз тұрасыз Оралда

### Пайдалануға ұсынылатын Әдебиеттер: [1-20]

Тақырып Тармақталу алгоритмдері. Таңдау операторы

### Тапсырма

3 сан берілген. Осылардың ішінен минималды санды тауып, экранға шығару.

### Әдістемелік нұсқаулар:

```

#нахождение минимального из 3-х чисел
a=input('Введите целое число \n')
b=input('Введите целое число \n')
c=input('Введите целое число \n')
if a<b:
    if a<c:
        y=a
    else:
        y=c
else:
    if b<c:
        y=b
    else:
        y=c
print('Минимальное:', y)

```

The flowchart starts with an oval labeled 'Басы' (Start) leading to a parallelogram labeled 'abc' (input). A diamond labeled 'a<b' branches into two paths: 'ЖОК' (Yes) leading to a parallelogram 'min=a', and 'ИЯ' (No) leading to another diamond 'a<c'. From 'a<c', 'ЖОК' leads to 'min=c' and 'ИЯ' leads to 'min=a'. From the first 'a<b' diamond, 'ИЯ' leads to a second diamond 'b<c'. From 'b<c', 'ЖОК' leads to 'min=c' and 'ИЯ' leads to 'min=b'. Both 'min=c' and 'min=b' paths merge into a parallelogram 'min'. Finally, a parallelogram 'min' leads to an oval labeled 'Соңы' (End).

Шешімі: Минимальное: 1

Введите целое число  
2  
Введите целое число  
5  
Введите целое число  
1

### Пайдалануға ұсынылатын Әдебиеттер: [1-20]

Тақырып Циклдермен жұмыс

### Тапсырмалар

Келесі сандар қатарының n элементтерінің қосындысын табыңыз: 1 -0.5 0.25 -0.125 ... n. (N) элементтер саны пернетақтадан енгізіледі. Экранда қатардың әр мүшесін және оның сомасын көрсетіңіз.

**Әдістемелік нұсқаулар:**

```
n=int(input('Қатар элементтерін енгізіңіз:'))
```

```
x=1
```

```
s=0
```

```
print(x)
```

```
for i in range(n):
```

```
    s+=x
```

```
    x/=-2
```

```
print(x)
```

```
print('Қатар элементтері қосындысы:',s)
```

**Шешімі:**

```
Введите количество элементов последовательности: 5
```

```
1
```

```
-0.5
```

```
0.25
```

```
-0.125
```

```
0.0625
```

```
-0.03125
```

```
Сумма ряда: 0.6875
```

**Пайдалануға ұсынылатын Әдебиеттер: [1-20]**

**Тақырып** Тізімдермен жұмыс. Python-дағы операциялардың тізімі

**Тапсырмалар**

Бір өлшемді массивті қолдана отырып, тапсырманы шешу. Жолдың солдан солға және солдан оңға қарай бірдей оқылатынын тексеріңіз.

**Әдістемелік нұсқаулар:**

```
s=input('Введите строку \n')
```

```
flag=1
```

```
string='',
```

```
for l in range(len(s)):
```

```
    if s[l]!='':
```

```
        string+=s[l]
```

```
print(string)
```

```
for l in range(len(s)//2):
```

```
if string[i]!=string[-i-1]:
```

```
    flag=0
```

```
break
```

```
if flag: print('Палиндром')
```

```
else: print('не палиндром')
```

```
Введите строку
```

```
а роза упала на лапу азора
```

```
арозаупаланалапуазора
```

```
Палиндром
```

**Пайдалануға ұсынылатын Әдебиеттер: [1-20]**

**Тақырып** Процедуралар және функциялар

**Тапсырмалар**

Бір өлшемді массивті енгізіңіз ұзындығы n. Ондағы бірінші және соңғы элементтерді ауыстырыңыз. Пернетақтадан массивтің ұзындығын және оның элементтерін енгізіңіз. Бағдарламада массив элементтерін ауыстыру процедурасын сипаттаңыз. Түпнұсқа және алынған массивтерді басып шығарыңыз.

**Әдістемелік нұсқаулар:**

```
def zam(x):
```

```
    tmp=x[0]
```

```
    x[0]=x[len(x)-1]
```

```
    x[len(x)-1]=tmp
```

```
A=[]
```

```
m=int(input('Введите длину массива:'))
```

```
for i in range(m):
```

```
    print('Введите ',i,' элемент массива')
```

```
    A.append(input())
```

```
print(A)
```

```
zam(A)
```

```
print(A)
```

Шешімі:

```

Введите длину массива: 5
Введите 0 элемент массива
0
Введите 1 элемент массива
1
Введите 2 элемент массива
2
Введите 3 элемент массива
3
Введите 4 элемент массива
4
[0, 1, 2, 3, 4]
[4, 1, 2, 3, 0]

```

### Пайдалануға ұсынылатын Әдебиеттер: [1-20]

**Тақырып** Екі өлшемді массивтермен жұмыс

**Тапсырмалар**

1. [6] әдебиеттегі 56 беттегі 3,4,5,6,7 –ші тапсырманы орындау керек.

**Әдістемелік нұсқаулар:** Тапсырманы орындау үлгісін [12] әдебиеттегі, 58-62 беттен қараңыз

### Пайдалануға ұсынылатын Әдебиеттер: [1-20]

**Тақырып** Графикалық интерфейсі бар қосымшаларды әзірлеу. Тасбақа көмегімен сурет салу

**Тапсырмалар**

"Тасбаканың" көмегімен шаршы және оған іштей сызылған шеңбер салыңыз.

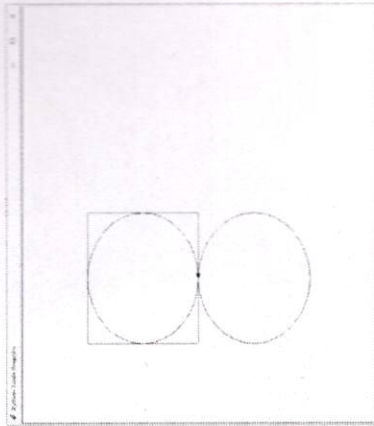
**Әдістемелік нұсқаулар:**

```

from turtle import *
t = Turtle()
t.screen.setup(800, 800)
def sq_cr(side):
    for i in range(4):
        t.left(90)
        t.fd(side)
        t.bk(side / 2)
        t.circle(side / 2, 360)
        t.left(180)
        t.circle(side / 2, 360)
    sq_cr(250)
t.screen.exitonclick()

```

t.screen.mainloop()



### Пайдалануға ұсынылатын Әдебиеттер: [1-20]

**Тақырып** Tkinter модулімен жұмыс істеу негіздері

**Тапсырмалар**

Қарапайым виджеттерді құруға арналған бағдарлама кодын жазыңыз: батырма, ауыстырғыштар, флажок және т.б.

**Әдістемелік нұсқаулар:**

**Батырма коды:**

```

from tkinter import *
root = Tk()
but = Button (root,
               text="Это кнопка", #надпись на кнопке
               width=30,height=5, #ширина и высота
               bg="white",fg="blue") #цвет фона и надписи
but.pack()
root.mainloop()
Ауыстырғыштар құру коды:
var=IntVar()
var.set(1)
rad0 = Radiobutton(root,text="Первая",
                   variable=var,value=0)
rad1 = Radiobutton(root,text="Вторая",
                   variable=var,value=1)
rad2 = Radiobutton(root,text="Третья",
                   variable=var,value=2)
Флажоктар құру:
c1 = IntVar()

```

```

c2 = IntVar()
che1 = Checkbutton(root,text="Первый флажок",
variable=c1,onvalue=1,offvalue=0)
che2 = Checkbutton(root,text="Второй флажок",
variable=c2,onvalue=2,offvalue=0)
ListBox тізім құру коды:
r = ['Linux','Python','Tk','Tkinter']
lis = Listbox(root,selectmode=SINGLE,height=4)
for i in r:
lis.insert(END,i)

```

### Пайдалануға ұсынылатын Әдебиеттер: [1-20]

**Тақырып Деректерді** визуализациялау. Диаграммалар тұрғызу

#### Тапсырмалар

1.  $y=x$  және  $y=x^2$  графиктерін сызыңыз.
2. Дөңгелек диаграмма тұрғызыңыз.

#### Әдістемелік нұсқаулар:

```

# Линейная зависимость
x = np.linspace(0, 10, 50)
y1 = x
# Квадратичная зависимость
y2 = [i**2 for i in x]
# Построение графиков
plt.figure(figsize=(9, 9))
plt.subplot(2, 1, 1)
plt.plot(x, y1) # построение графика
plt.title("Зависимости: y1 = x, y2 = x^2") # заголовок
plt.ylabel("y1", fontsize=14) # ось ординат
plt.grid(True) # включение отображение сетки
plt.subplot(2, 1, 2)
plt.plot(x, y2) # построение графика
plt.xlabel("x", fontsize=14) # ось абсцисс
plt.ylabel("y2", fontsize=14) # ось ординат
plt.grid(True) # включение отображение сетки
vals = [24, 17, 53, 21, 35]
2-тапсырма шешімі:

```

```

labels = ["Ford", "Toyota", "BMW", "AUDI", "Jaguar"]
fig, ax = plt.subplots()
ax.pie(vals, labels=labels)
ax.axis("equal")

```

### Пайдалануға ұсынылатын Әдебиеттер: [1-20]

**13 Жазбаша жұмыстар және курстық жұмыстар тақырыптары**

#### Рефераттар тақырыптары

1. Бағдарламалаудың алгоритмдік тілдері. Даму тарихы.
2. Бейсик бағдарламалау тілі
3. Си бағдарламалау тілі
4. Құрылымдық бағдарламалау
5. Функциональды бағдарламалау
6. Логикалық бағдарламалау
7. Объектілі-бағытталған бағдарламалау
8. Python тілінің графикалық мүмкіншілігі
9. Сорттау алогритмі және оларға талдау
10. Мәліметтердің динамикалық құрылымы

#### Бақылау жұмыстар тақырыптары

1. Сзықтық алгоритмдер
2. Шартты алгоритмдер
3. Циклдік алгоритмдер
4. Массивтерді өңдеу
5. Жазбалармен және файлдармен жұмыс

#### Пайдалануға ұсынылатын әдебиеттер:

1. Свердлов С. З. Языки программирования и методы трансляции. – С-Пб.: БХВ-Петербург, 2014 г.
2. О. Л. Голицына, И. И. Попов Основы алгоритмизации и программирования. – М.: Форум, 2016 г.
3. Колдаев В. Д. Основы алгоритмизации и программирования. – М.: Форум, Инфра-М, 2016 г.
4. Гвоздева В. А. Введение в специальность программиста, 2-ое издание. – М.: Форум, Инфра-М, 2015 г.

5. Миронченко А.С. Императивное и объектно-ориентированное программирование на Turbo Pascal и Delphi. – Одесса:ВМВ, 2015 г.

#### Курстық жұмыстар тақырыптары

- Деректер құрылымы және алгоритмдер түсінігі
- Деректер құрылымын жіктеу.
- Мәліметтер құрылымының классификациясы Оларға жүргізілетін операциялар
- Python тіліне кіріспе
- Айнymалы және арифметикалық өрнектер
- Шартты операторлар
- Циклдар.
- Шартсыз көшу операторы, белгі
- Стандартты Python пакеті
- Программа құрылымы.Енгізу және шығару
- Блок. Тармақталу. Таңдау нұсқауы
- Файлдармен енгізу-шығару операциялары
- Жолдар Оларға қолданылатын операциялар
- Тізімдер
- Жиындар
- Массив. Массивтерге қолданылатын операциялар.
- Жазулар
- Сөздіктер
- Итерация және циклдер
- Функциялар
- Генераторлар. Қосалқы бағдарламалар
- Объектілер мен кластар.
- Ерекшелік жағдайлар. Модульдер
- Іздеу және таңдау алгоритмдері
- Көрсеткіштік тип. Динамикалық айнымалылар
- Сұрыптау алгоритмдері

#### Рефераттар жұмыстар тақырыптары

1. Turbo Profiler программасы арқылы TP-ғы программаның әр үзінділерінің тиімділігін бағалау
2. Resource Workshop программалау пакеті
3. Структурлық программалаудың теориясы мен тәжірибесі
4. TP-ғы анимация
5. Паскальдағы мәліметтер қоры
6. Паскальдағы эксперттік жүйе
7. Ханой ғимарат есебі.
8. 8 ферз туралы есеп
9. Үй бухгалтериясы
10. Өмірді модельдеу
11. Статистика бойынша пакет
12. Инвестицияларды модельдеу
13. Файлдарды ығыстыру
14. Фибоначчи сандары
15.  $f(x) = 0$  теңдеуінің шешу әдістері
16. Қыйсықтың ұзындығын есептеу
17. Ең кіші квадраттар әдісі

#### Пайдалануға ұсынылатын әдебиеттер:

1. Свердлов С. З. Языки программирования и методы трансляции. – С-Пб.: БХВ-Петербург, 2023 г.
2. О. Л. Голицына, И. И. Попов Основы алгоритмизации и программирования. – М.: Форум, 2023 г.
3. Колдаев В. Д. Основы алгоритмизации и программирования. – М.: Форум, Инфра-М, 2023 г.
4. Гвоздева В. А. Введение в специальность программиста, 2-ое издание. – М.: Форум, Инфра-М, 2023 г.
5. Миронченко А.С. Императивное и объектно-ориентированное программирование на Turbo Pascal и Delphi. – Одесса:ВМВ, 2023 г.

#### 14. Баға туралы ақпарат

##### Студенттің білімін бағалайтын рейтинг-шкала

Рейтинг	Балдар
Ағымдық	50 - 100
Аралық	50 - 100
Қорытынды	50 - 100
<b>Барлығы:</b>	<b>50 - 100</b>

##### Студенттердің білімін бағалау критерийі

Рейтинг (әріптік эквивалент)	Критерийлер
0-49 (F)	Материалды толық меңгермеу; оқу бағдарламасының негізгі материалын игере алмау және дәрекі қателер жіберу.
50-74 (D, D+, C-, C, C+)	Жауабы негізінде дұрыс, бірақ толық емес; материалды жеткілікті түрде меңгермеу; материалды түсіндіруде теория жағынан кейбір қателер жіберу; материалды тек репродуктивті түрде түсіну; өз бетінше нәтиже шығара алмау.
75-89 (B-, B, B+)	Жауабы толық, дұрыс; материалдың негізгі идеяларын, түсініктерін және теңдеулер түрлерін білу; дәлелдей, нәтижелер шығара білу; бірақ жауапта дәлдік жоқ.
90-100 (A-, A)	Жауабы толық, дұрыс; материалды толық меңгеру; дәлелдеулер мен нәтижелер шығара білу.

##### 15. Курс бойынша емтихан сұрақтары

- Деректер құрылымы және алгоритмдер түсінігі
- Деректер құрылымын жіктеу.
- Мәліметтер құрылымының классификациясы Оларға жүргізілетін операциялар
- Python тіліне кіріспе
- Айнымалы және арифметикалық өрнектер
- Шартты операторлар
- Циклдар.
- Шартсыз көшу операторы, белгі

- Стандартты Python пакеті
- Программа құрылымы.Енгізу және шығару
- Блок. Тармақталу. Таңдау нұсқауы
- Файлдармен енгізу-шығару операциялары
- Жолдар Оларға қолданылатын операциялар
- Тізімдер
- Жиындар
- Массивы. Массивтерге қолданылатын операциялар.
- Жазулар
- Сөздіктер
- Итерация және циклдер
- Функциялар
- Генераторлар. Қосалқы бағдарламалар
- Объектілер мен кластар.
- Ерекшелік жағдайлар. Модульдер
- Іздеу және таңдау алгоритмдері
- Көрсеткіштік тип. Динамикалық айнымалылар
- Сұрыптау алгоритмдері

##### Мазмұны

1. Күндізгі оқыту нысанының жұмыс оқу жоспары.....	3
2. Оқытушы туралы мәліметтер:.....	4
3. Пререквизиттері: .....	4
4. Постреквизиттері:.....	4
5. Пәннің қысқаша мазмұны.....	5
6. Пән бойынша тапсырмаларды орындау және өткізу кестесі.....	6
7. Пәннің оқу- әдістемелік картасымен қамтамасыз етілуі.....	7
8. Лекциялық кешен (лекция тезисі).....	8
9. Семинарлық (практикалық) сабақтар жоспары .....	32
10. Лабораториялық сабақтар жоспары.....	39
11. Оқытушының жетекшілігімен орындалатын студенттердің өзіндік жұмыстары бойынша өткізілетін сабақтардың жоспары.....	39
12. Студенттердің өзіндік жұмыстары бойынша сабақтар жоспары .....	48
13. Жазбаша жұмыстар және курстық жұмыстар тақырыптары.....	56
14. Баға туралы ақпарат .....	59
15. Курс бойынша емтихан сұрақтары .....	59



«Алгоритмдеу және программалау тілдері»  
пәні бойынша

6B01505-Информатика білім беру бағдаламасы

### ОҚУ БАҒДАРЛАМАСЫ (SYLLABUS)

**Кельдибекова Алия Болатовна**  
**Каменова Шынар Кайкеновна**

---

Басуға — 20 ж.Қол қойылды.Пішімі 60x84 1/16. Кітап-журнал қағазы.  
Көлемі — сс.-б.т. Таралымы \_\_\_\_\_ дана. Тапсырыс № \_\_\_\_\_

---

Е.А.Бокетов атындағы Караганды университетінің баспасы  
100012, Караганды қ., Гоголь к-сі, 38