

3.1-тема: Python программалоо тили

- 1) 30 окуучунун ичинен сабакта 24тү эле олтурат. Бул класстагы катышуу пайызын эсептөөчү программаны түзгүлө.

Чыгаруу:

```
print(24 / 30 * 100)
```

- 2) Убакыт интервалы саат, минута жана секунда менен берилген. Ошол эле интервалды секундада гана көрсөтө турган программаны түзгүлө.

Чыгаруу:

```
h = int(input())
```

```
m = int(input())
```

```
s = int(input())
```

```
print(h * 60 * 60 + m * 60 + s)
```

3.2-тема: Маалыматтардын тиби жана алар менен болгон амалдар

- 1) **a** өзгөрмөсүнө 23, **b**га 7.5, **c**га «Hello» маанисин ыйгаргыла.

Чыгаруу:

```
a = 23
```

```
b = 7.5
```

```
c = 'Hello'
```

- 3) Мурунку **a** өзгөрмөсүнүн маанисин 2.5 эсеге көбөйтүү менен өзгөрткүлө, жыйынтыгын **d** өзгөрмөсүнө жазып, жакынкы бүтүн санга чейин тегеректегиле.

Чыгаруу:

```
a = float(input('a='))
```

```
d = 2.5 * a
```

```
print(round(d))
```

- 3) Температура Цельсий градуста берилген. Берилген температураны Фаренгейт градуста туюндуруучу программаны түзгүлө. Туюндуруу үчүн формула – $t(F) = 9/5 * t(C) + 32$.

Чыгаруу:

```
c = float(input('c='))
```

```
f = 9 / 5 * c + 32
```

```
print(f)
```

- 4) Туюнтманын маанисин эсептөөчү программаны жазгыла:

$$\frac{(a+b)*c}{x^2+y^2}$$

Чыгаруу:

```
a, b, c, x, y = map(float, input().split())
z = ((a + b) * c) / (x ** 2 + y ** 2)
print(z)
```

3.3-тема: Шарттуу операторлор

1) Үч сан берилди. Алардын ичинен канча сан бирдей экенин аныктоочу программа түзгүлө.

Чыгаруу:

```
a, b, c = map(float, input().split())
if a == b:
    if b == c:
        print(3)
    else:
        print(2)
else:
    if b == c or a == c:
        print(2)
    else:
        print(0)
```

2) Натуралдык сан берилди. Бул сан: а) жуп экенин, б) 3кө эселүү экенин аныктоочу программаны түзгүлө.

Чыгаруу:

n = int(input())	n = int(input())
if a % 2 == 0:	if a % 3 == 0:
print('Ооба')	print('Ооба')
else:	else:
print('Жок')	print('Жок')

3) Киргизилген айдын номерине жараша жыл мезгилинин атын чыгарып бере турган программаны жазгыла. Эгерде туура эмес номер киргизилсе ката жөнүндө билдирүүнү чыгарсын.

Чыгаруу:

```
n = int(input('Айдын номерин киргиз: '))
if n < 1 or n > 12:
    print('Ката')
elif n == 12 or n < 3:
    print('Кыш')
elif n < 6:
    print('Жаз')
elif n < 9:
    print('Жай')
else:
    print('Күз')
```

4) Сомду доллар жана евро валюталарына алмаштыруучу программа жазгыла.

Чыгаруу:

```
c = int(input('Сомду киргиз: '))
print('Сомду долларга алмаштырсан 0 санын, ал эми еврого алмаштырсан 1 санын танда.')
n = int(input())
```

```
if n == 0:
    k = float(input('Доллардын курсун киргиз: '))
    print(c / k, 'доллар')
elif n == 1:
    k = float(input('Евронун курсун киргиз: '))
    print(c / k, 'евро')
else:
    print('0 же 1 санын тандоо керек')
```

3.4-тема: while жана for циклдери

1) **a** жана **b** бүтүн сандарын алган жана **адан b**га чейинки диапазондогу бардык натуралдык сандардын квадраттарын чыгара турган программаны түзгүлө.

Чыгаруу:

```
a, b = map(int, input().split())
```

```
for i in range(a, b + 1):
```

```
    print(i * i, end = ' ')
```

2) Натуралдык сан берилген. Ал сандын цифраларынын суммасын чыгарып берген программаны жазгыла.

Чыгаруу:

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

```
s = 0
```

```
while n != 0:
```

```
    s += n % 10
```

```
    n //= 10
```

```
print(s)
```

3) Узундугу 23 метр болгон тактай берилген. Бул тактайдан узундуктары 1,5 м жана 2 м болгон бүтүн сандагы канча минималдык кесиндини даярдоого боло турганын эсептөөчү программаны түзгүлө.

Чыгаруу:

```
s = 23 * 10
```

```
a = 15
```

```
b = 20
```

```
mn = s
```

```
for i in range(s // b + 1):
```

```
    j = (s - i * b) // a
```

```
    if 20 * i + j * 15 == s:
```

```
        mn = min(mn, i + j)
```

```
print(mn)
```

8-класс

3.1-тема: Татаал шарттар: and, or, not

1) and операторунун жардамы менен бирөөсү чындыкты, экинчиси – жалганды көрсөткөн эки татаал логикалык туюнтманы түзгүлө.

$(5 > 3)$ and $(20 \% 4 == 0)$

$(4 < 2)$ and $(2 * 2 == 4)$

2) Жогорудагы маселени **or** операторун колдонуп аткаргыла.

$(7 > 5)$ or $(3 != 3)$

$(2 == 4)$ or $(33 \% 3 != 0)$

3.2-тема: Тизмелер, кортеждер жана сөздүктөр

1) 5 сандан турган тизме түзгүлө. Экранга алардын суммасын, максималдык жана минималдык элементтерин чыгаргыла. Чыгаруу үчүн Python тилинде камтылган `sum()`, `max()` жана `min()` функцияларын колдонула.

```
a = [23, -10, 2, 123, -45, 67]
```

```
print(sum(a), max(a), min(a))
```

2) 5 элементтен турган тизмени түзгүлө. Тизменин биринчи жана акыркы элементин чыгаргыла.

```
a = [2, -3, 7, 12, 321]
```

```
print(a[0], a[-1])
```

3) `a = [1, 2, 3, 5, 8, 13, 21, 34, 55, 89]` тизмеси берилди. Экранга `for` циклин колдонуп тизменин так элементтерин чыгаргыла.

```
a = [1, 2, 3, 5, 8, 13, 21, 34, 55, 89]
```

```
for el in a:
```

```
    if el % 2 == 1:
```

```
        print(el, end = ' ')
```

4) 3 тапшырмада берилген тизмеден: а) 5 жана 34 элементтерин өчүргүлө; б) тизмеге 6 жана 120 сандарын кошкула.

```
a = [1, 2, 3, 5, 8, 13, 21, 34, 55, 89]
```

```
a.remove(5)
```

```
a.remove(34)
```

```
print(*a)
```

```
a = [1, 2, 3, 5, 8, 13, 21, 34, 55, 89]
```

```
a.append(6)
```

```
a.append(120)
```

```
print(*a)
```

3.3-тема:

Циклдик алгоритмдер

1) 12, 34, 8, 15, 9, 26 сандарынан турган тизме берилген. Экранга төмөнкү сандарды чыгаргыла: а) **n** санынан кичине; б) биринчи саны **n**ден чоң; в) **n** санынан чоң бардык сандардын суммасын.

```
a = [12, 34, 8, 15, 9, 26]
```

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

```
s = 0
```

```
for el in a:
```

```
    if el < n:
```

```
        s += el
```

```
print(s)
```

```
a = [12, 34, 8, 15, 9, 26]
```

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

```
s = 0
```

```
for el in a:
```

```
    if el > n:
```

```
        s += el
```

```
        break
```

```
print(s)
```

```
a = [12, 34, 8, 15, 9, 26]
```

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

```
s = 0
```

```
for el in a:
```

```
    if el > n:
```

```
        s += el
```

```
print(s)
```

2) Төмөнкү программанын иштөөсүнүн жыйынтыгында алынган у өзгөрмөсүнүн маанисин жазгыла:

```
y = 5
```

```
for i in range(2,6):
```

```
    y = y + 4 * i
```

```
print(y)
```

```
y = 5
```

```
i = 2
```

```
i = 3
```

```
i = 4
```

```
i = 5
```

```
y = 5 + 4 * 2 = 13
```

```
y = 13 + 4 * 3 = 25
```

```
y = 25 + 4 * 4 = 41
```

```
y = 41 + 4 * 5 = 61
```

Жообу 61

3) Берилген код боюнча алгоритмдин ар бир кадамындагы өзгөрмөлөрдүн маанилерин таблицага жазгыла:

k=4, p=1040, m=2

while p != m*m:

 k=k+1

 p=p-4

 m=m*2

print (k)

*3) Берилген код боюнча алгоритмдин ар бир кадамындагы өзгөр-
мөлөрдүн маанилерин таблицкага жазгыла:*

k=4 p=1040 m=2

while p != m*m:

 k=k+1

 p=p-4

 m=m*2

print (k)

k	p	m	m*m

k	p	m	m * m
4	1040	2	4
5	1036	4	16
6	1032	8	64
7	1028	16	256
8	1024	32	1024

Жообу: 8

4) Майнкрафттын каарманы Алекстин минутасына 4 минерал чыгара турган машинасы бар. Ар бир 100 минералга дал ушундай эле минутасына 4 минерал чыгарган машина курууга болот. Бир сааттан кийин Алекстин канча машинасы боло тургандыгын аныктоочу программаны жазгыла.

t = 60

cnt = 1 # Машинанын саны

s = 0 #Минералдын саны

for i in range(t):

 s += 4 * cnt

 if s >= 100:

 cnt += 1

 s -= 100

print(cnt)

3.4-тема:

Камтылган шарттуу амалдар жана циклдер

1) Колдонуучудан санды киргизүү суралсын. Эгер ал сан оң болсо, анда экранга 1 цифрасы чыгарылсын. Эгерде сан терс болсо, анда -1 цифрасы, эгер киргизилген сан 0 болсо, анда экранга 0 чыгарылсын. Коддо көп тармактанган шарттуу операторду колдонула.

n = float(input())

if n > 0:

```
print(1)
elif n < 0:
    print(-1)
else:
    print(0)
```

2) 1ден 15ке чейинки сандардын тизмесинен 3кө бөлүнгөн бардык сандарды алып салган циклдик программаны жазгыла. **Continue** операторун колдонула.

```
for i in range(1, 16):
    if i % 3 == 0:
        continue
    print(i, end = ' ')
```

3) 2чи тапшырмадагы сандардын тизмесинен 7ден кичине болгон сандарды экранга чыгаргыла. **break** операторун колдонула.

```
for i in range(1, 16):
    if i == 7:
        break
    print(i, end = ' ')
```

4) 5 кг, 10 кг жана 15 кг алма бата турган ящиктер берилди. 100 кг алманы бөлүштүрүү үчүн канча ар кандай өлчөмдөгү ящик керек боло тургандыгын аныктоочу программаны түзүү керек.

```
s = 100
a = 5
b = 10
c = 15
for i in range(s // a + 1):
    for j in range(s // b + 1):
        for k in range(s // c + 1):
            if i * a + b * j + k * c == s:
                print(i, j, k)
```

Оптималдуу варианты:

$$5 * i + 10 * j + 15 * k = 100$$

$$i + 2 * j + 3 * k = 20$$

$$i = 20 - 2 * j - 3 * k \quad 0 \leq j \leq 10, 0 \leq k \leq (20 - 2 * j) // 3$$

```
for j in range(11):
    for k in range((20 - 2 * j) // 3 + 1):
        i = 20 - 2 * j - 3 * k
        print(i, j, k)
```

3.5-тема:

Функциялар

1) 3 берилген санды өсүү тартибинде экранга чыгаруучу функцияны жазгыла.

```
def sort_three(a, b, c):  
    if a > b:  
        a, b = b, a  
    if b > c:  
        b, c = c, b  
    if a > b:  
        a, b = b, a  
    return a, b, c  
x, y, z = map(float, input().split())  
x, y, z = sort_three(x, y, z)  
print(x, y, z)
```

2) Эки натуралдык сандын эң чоң жалпы бөлүүчүсүн табуучу функцияны жазгыла.

```
from math import gcd  
def nok(a, b):  
    return a * b // gcd(a, b)  
n, m = map(int, input().split())  
print(nok(n, m))
```

3) Эки натуралдык сан берилген. Алардын кайсынысында: а) цифралар көп экендигин; б) цифралардын суммасы чоң экендигин аныктоочу программа түзгүлө.

<pre>def count_digit(n): a = [] while n != 0: a.append(n % 10) n //= 10 s = 0 for i in range(10): if i in a: s += 1 return s a, b = map(int, input().split()) s1 = count_digit(a) s2 = count_digit(b) if s1 > s2: print('Цифралардын саны биринчи санда көп') elif s1 == s2: print('Цифралардын саны барабар') else: print('Цифралардын саны экинчи санда көп')</pre>	<pre>def sum_digit(n): s = 0 while n != 0: s += n % 10 n //= 10 return s a, b = map(int, input().split()) s1 = sum_digit(a) s2 = sum_digit(b) if s1 > s2: print('Биринчи санда цифралардын суммасы чон') elif s1 == s2: print('Цифралардын суммасы барабар') else: print('Экинчи санда цифралардын суммасы чон')</pre>
----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

4) Эки үч бурчтуктун жактары берилген. Алардын: а) периметрлеринин; б) аянттарынын суммасын таап берүүчү программаны түзгүлө

<pre>def per(a, b, c): return a + b + c a1, b1, c1 = map(float, input().split()) a2, b2, c2 = map(float, input().split()) print(per(a1, b1, c1) + per(a2, b2, c2))</pre>	<pre>def per(a, b, c): return a + b + c def area(a, b, c): p = per(a, b, c) / 2 s = (p * (p - a) * (p - b) * (p - c)) ** 0.5 return s a1, b1, c1 = map(float, input().split()) a2, b2, c2 = map(float, input().split()) print(area(a1, b1, c1) + area(a2, b2, c2))</pre>
----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

3.6-тема:

Массивдер

1) Тизмени толтуруу генераторун колдонуп, 1ден 30га чейинки сандардын тизмесин түзгүлө. Экранга **a** массивиндеги жуп жана так сандардын санын чыгаргыла.

```
a = [ i + 1 for i in range(30)]
s1 = 0
s2 = 0
for el in a:
    if el % 2 == 0:
        s1 += 1
    else:
        s2 += 1
print(s1, s2)
```

2) Мурунку **a** тизмесинен экранга **b** тизмесине төмөнкүлөрдү чыгаруучу программа түзгүлө: а) 3кө бөлүнө турган бардык сандарды; б) так сандарды.

```
from random import randint
n = int(input('Тизмедеги элементтердин санын
киргиз: '))
a = [randint(1, 50) for i in range(n)]
print(*a)
b = [el for el in a if el % 3 == 0]
print(*b)
```

```
from random import randint
n = int(input('Тизмедеги элементтердин санын
киргиз: '))
a = [randint(1, 50) for i in range(n)]
print(*a)
b = [el for el in a if el % 2 == 1]
print(*b)
```

3) Тизменин эки кошуна элементинин (**a[0]** жана **a[1]**, **a[2]** жана **a[3]** ...ж.б.) ордун алмаштыра турган программаны түзгүлө.

```
from random import randint
n = int(input('Тизмени санын киргиз: '))
a = [randint(-20, 50) for i in range(n)]
print(*a)
for i in range(0, n - 1, 2):
    a[i], a[i + 1] = a[i + 1], a[i]
print(*a)
```

4) Массивди 100 кокустук сан менен толтургула. Алынган тизмени экранга бир катарда 10 элементтен кылып чыгаргыла. Тизмени чыгаруу үчүн өзүнчө функцияны түзгүлө, аргумент катары ал тизмени кабыл алыш керек.

```
def pr(a):
    for i in range(100):
        print(a[i], end = ' ')
        if i % 10 == 9:
            print()
```

```
from random import randint
a = [randint(10, 99) for i in range(100)]
pr(a)
```

5) Дене тарбия сабагынан окуучулардын бойлорун тизмеге жазышты. Бул тизмеден бою эң узун жана эң кыска окуучуларды тапкыла.

```
from random import randint
n = int(input('Окуучулардын санын киргиз: '))
a = [randint(120, 190) for i in range(n)]
print(*a)
print(max(a), min(a))
```

3.7-тема:

Саптар жана алар менен болгон амалдар

1) Өзүңөрдүн атыңарды киргизгиле: а) экранга 1чи жана 3-символду чыгаргыла; б) силердин атыңардын 1чи жана акыркы символдору дал келээрин аныктагыла.

```
name = input('Атыңды киргиз: ')
print(name[0], name[2])
```

```
name = input('Атыңды киргиз: ')
if a[0] == a[-1]:
    print('Дал келет')
else:
    print('Дал келбейт')
```

2) Клавиатурадан бир нече сөз киргизүүнү жана ошол сөздөрдүн ичинен эң кыска сөздүн узундугун аныктоочу программа түзгүлө.

```
n = int(input('Канча сөз киргизесиз?'))
mn = len(input())
for i in range(n - 1):
    s = len(input())
    if s < mn:
        mn = s
print(mn)
```

3) isdigit() саптык методу сап жалаң гана цифралардан тураарын текшерип берет. Киргизүүдө программа эки бүтүн санды сураган жана алардын суммасын чыгарып берүүчү программаны түзгүлө. Туура эмес киргизүү учурунда программа ката деп токтоп калбастан, кайрадан эле сандарды сурап тургандай кылыш керек.

```
while True:
    a, b = input().split()
    if a.isdigit() and b.isdigit():
        break
print(int(a) + int(b))
```

3.8-тема:

Саптарды форматтоо

Колдонуучудан логин жана паролду сураган шаблонду жазгыла. Эгерде логин же пароль туура эмес болуп калса, «мындай логин жана пароль табылган жок» деген билдирүүнү

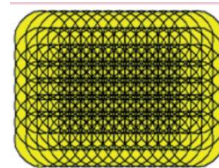
чыгарсын. Ал эми логин жана пароль туура болсо, анда колдонуучунун атын көрсөткөн саламдашууну чыгаргыла.

Шаблон кандай формада болуш керек, түшүнбөдүм.

3.9-тема:

Python тилинде графика менен иштөө

1) Turtle модулунун жардамында айланалардан килем жасагыла. Мында айланалар бир түс менен, ал эми килемдин фону башка түс менен боёлсун.



```
import turtle as t
t.screensize(800, 600)
t.reset()
t.bgcolor('yellow')
t.up()
x1 = -200
y1 = 100
r = 60
t.color('black')
t.speed(10)
for y in range(7):
    t.up()
    y2 = y1 - r // 4 * 3 * y
    t.goto(x1, y2)
    for x in range(1, 18):
        t.pd()
        x2 = x1 + x * 2 * r // 5
        t.goto(x2, y2)
        t.circle(r)
```

2) Tkinter модулунун жардамында холстко жоондугу жана түсү кокустан тандалган сызыктарды чыгаруучу программаны түзгүлө.

```
from random import randint
from tkinter import *
size = 800
tk = Tk()
canv = Canvas(tk, width = size, height = size)
canv.pack()
for i in range(1000):
    x1, y1, x2, y2 = randint(10, size - 50), randint(10, size - 50), randint(10, size - 50), randint(10, size - 50)
    w = randint(1, 5)
    r = randint(1, 255) / 255
    color = "#%02X%02X%02X" % (randint(0, 255), randint(0, 255), randint(0, 255))
    canv.create_line(x1, y1, x2, y2, width = w, fill = color)
tk.update()
```

```
tk.mainloop()
```

9-класс

3.1-тема:

Рекурсия

1) n берилген санынын цифраларынын суммасын чыгаруучу рекурсивдүү функцияны жазгыла.

```
def sum_digit(n):  
    if n < 10:  
        return n  
    return n % 10 + sum_digit(n // 10)  
print(sum_digit(int(input('Санды киргиз: '))))
```

2) Алынган: сандын а) жуптугун; б) тактыгын аныктаган рекурсивдүү функцияларды жазгыла.

```
def check(n):  
    if n < 2:  
        return n % 2 == 0  
    return check(n - 2)  
n = int(input('Санды киргиз: '))  
if check(n):  
    print('Жуп сан')  
else:  
    print('Жуп эмес сан')
```

```
def check(n):  
    if n < 2:  
        return n % 2 != 0  
    return check(n - 2)  
n = int(input('Санды киргиз: '))  
if check(n):  
    print('Так сан')  
else:  
    print('Так эмес сан')
```

3) Эң кичине жалпы бөлүүчүнү (ЭКЖБ) табуу үчүн рекурсивдүү функцияны жазгыла.

```
def lcm(a, b):  
    global mult  
    mult += b  
    if mult % a == 0 and mult % b == 0:  
        return mult  
    lcm(a, b)  
    return mult  
mult = 0  
a = int(input('1-санды киргиз: '))  
b = int(input('2-санды киргиз: '))  
print(lcm(min(a, b), max(a, b)))
```

4) Арифметикалык прогрессиянын айырмасы жана биринчи мүчөсү берилген. Прогрессиянын биринчи n мүчөлөрүнүн суммасын табуучу рекурсивдүү функцияны жазгыла.

```
def suma(a, n):  
    if n == 1:  
        return a  
    return a + suma(a + d, n - 1)  
a1 = int(input())  
d = int(input())  
n = int(input())
```

```
print(sum_aprog(a1, n))
```

Эгерде арифметикалык прогрессияны айырмасы, биринчи мүчөсү анык сандар болсо, `int` дин ордуна `float` деп жазуу керек.

5) Геометриялык прогрессиянын жалпы бөлүүчүсү жана биринчи мүчөсү берилген. Прогрессиянын биринчи `n` мүчөлөрүнүн суммасын табуучу рекурсивдүү функцияны жазгыла.

```
def sum_gprog(b, n):  
    if n == 1:  
        return b  
    return b + sum_gprog(b * q, n - 1)  
b1 = int(input())  
q = int(input())  
n = int(input())  
print(sum_gprog(b1, n))
```

Эгерде геометрия прогрессияны бөлүмү, биринчи мүчөсү анык сандар болсо, `int` дин ордуна `float` деп жазуу керек.

3.2-тема:

Тизмелерди иштетүү алгоритмдери

1) Тизмени $(0, 20)$ интервалындагы кокустук сандар менен толтургула. `x` санын киргизгиле жана `x`ке барабар бардык маанилерди тапкыла.

Эскертүү (`x` ке барабар бардык маанилерди дебестен индекстерин чыгаргыла десе туура болмок)

```
from random import randint  
n = int(input('Тизменин элементтери канча сандан турсун'))  
a = [randint(1, 19) for i in range(n)]  
print(*a)  
x = int(input('x ке маани бер: '))  
for i in range(n):  
    if a[i] == x:  
        print(i, end = ' ')
```

2) `N` элементтен турган сан маанисиндеги бир өлчөмдүү массив берилген. Массивдин элементтерин оң жакка айланма жылдырууну аткар, б.а. `a(1) --> a(2)`; `a(2) --> a(3)`; ... `a(n) --> a(1)`.

```
from random import randint  
n = int(input('n ге маани бер: '))  
a = [randint(-20, 30) for i in range(n)]  
print(*a)  
x = a[-1]  
for i in range(n - 1):  
    a[i + 1] = a[i]  
a[0] = x
```

```
print(*a)
```

Үзүмдөрдү колдонсок

```
from random import randint
n = int(input('n ге маани бер: '))
a = [randint(-20, 30) for i in range(n)]
print(*a)
a = a[-1] + a[1:]
print(*a)
```

3) Кокустук сандардын тизмеси берилген. Тизменин бардык так сандарын нөлдөр менен алмаштыргыла жана алардын санын чыгаргыла.

```
from random import randint
n = int(input('Тизменин элементтери канча сандан турсун'))
a = [randint(1, 200) for i in range(n)]
print(*a)
s = 0
for i in range(n):
    if a[i] % 2 == 1:
        a[i] = 0
        s += 1
print(*a)
print(s)
```

4) Тизмени кокустук сандар менен толтургула жана тизменин x жана y индекстери менен белгиленген үзүмдөгү элементтери үчүн реверс жүргүзгүлө.

```
from random import randint
n = int(input('Тизменин элементтери канча сандан турсун? '))
a = [randint(1, 200) for i in range(n)]
print(*a)
x, y = map(int, input().split())
a = a[:x] + a[x:y][::-1] + a[y:]
print(*a)
```

3.3-тема:

Тизмелерди сорттоо

1) n элементтен турган бир өлчөмдүү сандык маанидеги массив берилген. Баштапкы массивдин элементтеринен эки жаңы массив түзгүлө. Биринчисине 3кө бөлүнгөн гана сандар, ал эми экинчисине 5ке бөлүнгөн гана сандар кирсин.

```
from random import randint
n = int(input('Тизменин элементтери канча сандан турсун? '))
a = [randint(1, 200) for i in range(n)]
print(*a)
b = []
c = []
for el in a:
    if el % 3 == 0:
        b.append(el)
```

```
if el % 5 == 0:  
    c.append(el)  
print(*b)  
print(*c)
```

2) Биринчи суроодогу программаны уланткыла жана 3кө бөлүнгөн сандарды өсүү тартибинде, ал эми 5ке бөлүнгөн сандарды кемүү тартибинде жайгаштыра тургандай кылып программаны толуктагыла.

```
from random import randint  
n = int(input('Тизменин элементтери канча сандан турсун? '))  
a = [randint(1, 200) for i in range(n)]  
print(*a)  
b = []  
c = []  
for el in a:  
    if el % 3 == 0:  
        b.append(el)  
    if el % 5 == 0:  
        c.append(el)  
b.sort()  
c.sort(reverse = True)  
print(*b)  
print(*c)
```

3) Бүтүн сандар менен тизме берилген. Тизмедеги эң кичине жана эң чоң сандын көбөйтүндүсүнүн жыйынтыгын чыгаруучу программаны түзгүлө.

```
from random import randint  
n = int(input('Тизменин элементтери канча сандан турсун? '))  
a = [randint(1, 200) for i in range(n)]  
print(*a)  
print(min(a) * max(a))
```

4) Колдонуучудан сапка бир нече сөз киргизүүсүн сураган программаны жазгыла. Сөздөрдү алардын узундугунун өсүү тартиби боюнча сорттогула.

```
n = int(input('Канча сөз киргизсениз? '))  
a = []  
for i in range(n):  
    a.append(input())  
a.sort(key = lambda a: len(a))  
print(*a)
```

3.4-тема:

Матрицалар

1) Эки өлчөмдүү массив берилген. Экранга төмөнкүлөрдү чыгаргыла: а) экинчи саптагы бардык элементтерди; б) экинчи мамычадагы бардык элементтерди.

```
from random import randint  
n = int(input())
```

```
from random import randint  
n = int(input('Саптын санын киргизгиле: '))
```

```
m = int(input())
a = [[randint(-20, 50) for j in range(m)] for i in
range(n)]
print(*a[1])
```

```
m = int(input('Мамычанын санын киргизгиле : '))
a = [[randint(-20, 50) for j in range(m)] for i in
range(n)]
for i in range(n):
    print(a[i][1], end = ' ')
```

2) Эки өлчөмдүү массив берилген. Массивдин каалагандай эки элементинин ордуларын алмаштырган программаны жазгыла.

```
from random import randint
```

```
n = int(input('Саптын санын киргизгиле: '))
```

```
m = int(input('Мамычанын санын киргизгиле : '))
```

```
a = [[randint(-20, 50) for j in range(m)] for i in range(n)]
```

```
for row in a:
```

```
    print(*row)
```

```
i1 = int(input('1-элементтин сабынын индексинин номерин киргиз: '))
```

```
j1 = int(input('1-элементтин мамычасынын индексинин номерин киргиз: '))
```

```
i2 = int(input('2-элементтин сабынын индексинин номерин киргиз: '))
```

```
j2 = int(input('2-элементтин мамычасынын индексинин номерин киргиз: '))
```

```
a[i1][j1], a[i2][j2] = a[i2][j2], a[i1][j1]
```

```
for row in a:
```

```
    print(*row)
```

3) n саптан жана m мамычадан турган тик бурчтуу a матрицасын кокустук сандар менен толтургула. Массивдин элементтеринин орточо арифметикалык маанисин тапкыла.

```
from random import randint
```

```
n = int(input('Саптын санын киргизгиле: '))
```

```
m = int(input('Мамычанын санын киргизгиле : '))
```

```
a = [[randint(-20, 50) for j in range(m)] for i in range(n)]
```

```
s = 0
```

```
for row in a:
```

```
    s += sum(row)
```

```
print(s / (n * m))
```

4) Матрицанын ар бир сабы үчүн табылган орточо маанилеринин эң чоңун аныктагыла.

```
from random import randint
```

```
n = int(input('Саптын санын киргизгиле: '))
```

```
m = int(input('Мамычанын санын киргизгиле : '))
```

```
a = [[randint(-20, 50) for j in range(m)] for i in range(n)]
```

```
b = []
```

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

```
    s = 0
```


Кошмурзаев Исмаил.

Ноокат району. М.Гапаров атындагы жатак-гимназиянын информатика мугалими

```
for j in range(n):  
    s += a[j][i]  
    b.append(s / n)  
print(*b)  
print(max(b))
```