



Әл-Фараби атындағы Қазақ ұлттық университеті  
Механика-математика факультеті



**Функциялар мен модульдер**

Темирбеков Нурлан Муханович ф-м.ғ.д., профессор

## Жоспар

1. Функциялар мен модульдер.
2. Математикалық модульдер.
3. NumPy модулі.
4. Matplotlib.pyplot арқылы график құру.

## Мақсаты

Python бағдарламалық тілінің негізгі құрылымы  
функциялар және қолданбалы модулдармен танысу,  
оны қолдану жолдарын талдау.

## Функциялар

Python-да функциясының құрылымы

```
Def func_name(param1, param2, . . .):  
    операторлар  
return қайтаратын_мәндер
```

Келесі функция  $f(x)$  санының алғашқы екі туындысын ақырлы айырымдар арқылы есептейді:

```
from math import atan
```

```
def derivatives(f,x,h=0.0001): # h әдепкі мәні бар  
    df =(f(x+h) - f(x-h))/(2.0*h)  
    ddf =(f(x+h) - 2.0*f(x) + f(x-h))/h**2  
    return df,ddf  
df,ddf = derivatives(atan,0.5) # әдепкі h мәнін  
пайдаланады  
print('Бірінші туынды =',df)  
print('Екінші ретті туынды =',ddf)
```

Бағдарламаның нәтижесі

Бірінші туынды = 0.7999999995730867

Екінші ретті туынды = -0.6399999918915711

## Функциялар мен модульдер

Мысалы: `def func(x1,x2,*x3)`.

$x_1$  және  $x_2$  позициялық параметрлер,

$x_3$  – артық параметрлерді қамтитын еркін ұзындықты кортежі.

Функцияны `def func(a,b,c,d,e)` түрінде шақыру параметрлер арасындағы келесі сәйкестікке әкеледі:

$$a \leftrightarrow x_1, b \leftrightarrow x_2, (c, d, e) \leftrightarrow x_3$$

Функция формуламен болғанда, `lambda` операторы пайдаланылады

```
func_name = lambda param1,param2,...:expression
```

Мысал:  $f(x, y) = x^2 + y^2$  функциясын анықтайтын бағдарлама жазыңыз:

```
>>> c = lambda x,y : x**2 + y**2
>>> print(c(3,4))
25
```

**Модуль** – бұл функциялар орналасқан жай ғана файл; модульдің аты файлдың аты болып табылады.

Модульді бағдарламаға `from module_name import *` оператор арқылы жүктеуге болады

## Математикалық модульдер

`math` модульдегі функцияларға қол жеткізудің жолдары

```
from math import func1,func2,...
```

```
>>> from math import log,sin
```

```
>>> print(log(sin(0.5)))
```

```
-0.735166686385
```

```
import math
```

```
math.func1
```

```
math.func2
```

```
>>> import math
```

```
>>> print(math.log(math.sin(0.5)))
```

```
-0.735166686385
```

```
import math as m
```

```
m.func1
```

```
m.func2
```

```
>>> import math as m
```

```
>>> print(m.log(m.sin(0.5)))
```

```
-0.735166686385
```

```
>>> import math
```

```
>>> dir(math)
```

```
'acos', 'acosh', 'asin', 'asinh', 'atan', 'atan2',  
'atanh', 'ceil', 'comb', 'copysign', 'cos', 'cosh',  
'degrees', 'dist', 'e', 'erf', 'erfc', 'exp', 'expm  
1', 'fabs', 'factorial', 'floor', 'fmod', 'frexp', 'fs  
um', 'gamma', 'gcd', 'hypot', 'inf', 'isclose',  
'isfinite', 'isinf', 'isnan', 'isqrt', 'ldexp', 'lgam  
ma', 'log', 'log10', 'log1p', 'log2', 'modf', 'na  
n', 'perm', 'pi', 'pow', 'prod', 'radians', 'remai  
nder', 'sin', 'sinh', 'sqrt', 'tan', 'tanh', 'tau', 'tr  
unc']
```

`cmath` модулі комплекс сандарды қабылдайды.

## NumPy модулі

Модуль тізімдерге ұқсас массив нысандарын ұсынады, бірақ модульде қамтылған көптеген функциялар арқылы түрлендіріп басқаруға болады.

Тізімді массивке айналдыру	<code>array(list, type)</code>	<pre>&gt;&gt;&gt; from numpy import array &gt;&gt;&gt; a = array([[2.0, -1.0],[-1.0, 3.0]]) &gt;&gt;&gt; print(a) [[ 2. -1.] [-1. 3.]</pre>
Элементтері нөл, $n \times m$ массив	<code>zeros((n, m),type)</code>	<pre>&gt;&gt;&gt; from numpy import * &gt;&gt;&gt; print(zeros((2,2))) [[ 0. 0.] [ 0. 0.]</pre>
Бір өлшемді массив	<code>arange(from,to,increment)</code>	<pre>&gt;&gt;&gt; from numpy import * &gt;&gt;&gt; print(arange(2,10,2)) [2 4 6 8] &gt;&gt;&gt; print(arange(2.0,10.0,2.0)) [ 2. 4. 6. 8.]</pre>

## Numru модулі

*Массивтердегі амалдар*

```
>>> from numpy import array
>>> a = array([0.0, 4.0, 9.0, 16.0])
>>> print(a/16.0)
[ 0.  0.25  0.5625  1. ]
>>> print(sqrt(a))
[ 1.  2.  3.  4.]
>>> print(sin(a))
[ 0.8414 -0.7568  0.4121 -0.2879]
```



## Matplotlib.pyplot арқылы график құру

```
import matplotlib.pyplot as plt
```

```
from numpy import arange,sin,cos
```

```
x = arange(0.0,6.2,0.2)
```

```
plt.plot(x,sin(x),'o-',x,cos(x),'s-')
```

```
plt.xlabel('x') # x осіне белгі қосу
```

```
plt.legend(('синус','косинус'),loc = 0) # loc.3 легенданы қосу
```

```
plt.grid(True) # координаталық торды қосу
```

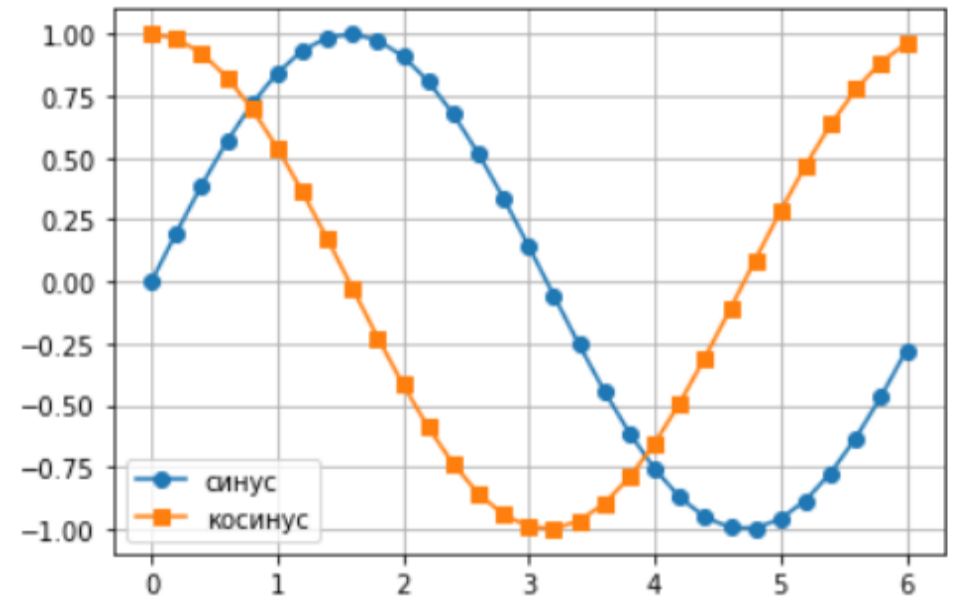
```
plt.savefig('testplot.png',format='png') # png сақтау
```

```
plt.show() # суретті экранда көрсету
```

Легенданы орналастыру (loc) кодтары

0	"Ең жақсы" орын
1	Жоғарғы оң жақ
2	Жоғарғы сол жақ

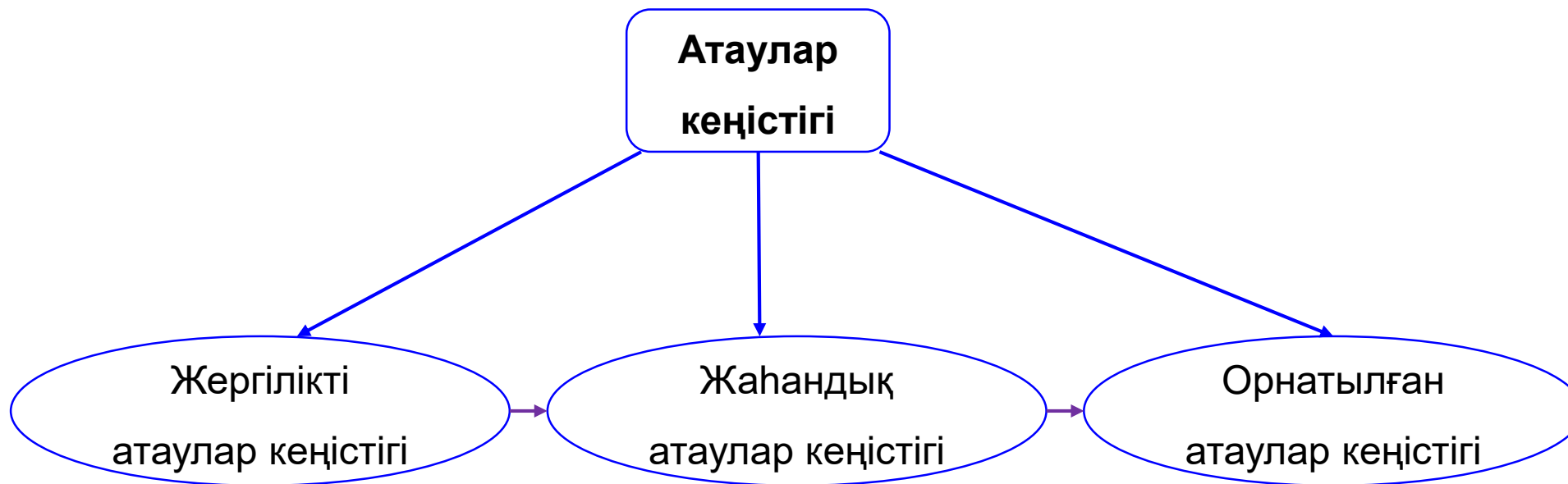
3	Төменгі сол жақ
4	Төменгі оң жақ



Жол және маркер мәнерлерінің таңбалары

'-'	Тұтас сызық
'--'	Үзік сызық
'-.'	Сызық-нүкте сызығы
':'	нүктелі сызық
'o'	Шеңбер маркері
's'	Шаршы маркері
'h'	Алтыбұрышты маркер
'x'	x маркері

## Айнымалылар әрекетінің аймағы



## Қорытынды

1. Функцияны, модульді құру.
2. Math, Cmath модульдері.
3. NumPy модулімен жұмыс.
4. Matplotlib.pyplot арқылы график салу.

## Пайдаланылган әдебиеттер тізімі

1. Jaan Kiusalaas. Numerical methods in engineering with Python. Cambridge University Press. ISBN 978-1-107-03385
2. Вабищевич П.Н. Численные методы: Вычислительный практикум. — М.: Книжный дом «ЛИБРОКОМ», 2010. — 320 с.
3. Киреев В. И., Пантелеев А. В. Численные методы в примерах и задачах: Учебное пособие. —СПб.: Издательство «Лань», 2015. — 448 с.