Computer Science I

Computer Science Part 2 —>

Kolokoltseva Tatiana GorbunovaTN@mgsu.ru

Num	First and last name
1	Акакпо Мофудукпе Маттье
2	Аль Дин А Б М Моснадул
3	Арему Даниел Олаофе
4	Аслонов Шавкат Шарофджонович
5	Ахмед Мухаммад Фахим
6	Бадхон Мустафизур Рахман
7	Банда Форчун
8	Гаффоров Темурбек Муроджонович
9	Гембицкий Тимофей Михайлович
10	Даиро Джеймс Толувани
11	Зеленщикова Дарья Александровна
12	Ибрагим Муххамад Амир
13	Ибрахим Абдулкуддус Таийе
14	Кабир Мд Тарек
15	Калито Калито
16	Кумар Принс
17	Миа Ракиб

18	Мумба Деррик
19	Одрохо Ишрак Бин Насим
20	Оланреважу Шериффдин Оламилекан
21	Ондала Нандипха
22	Ондала Хилия
23	Пири Раджени
24	Рахаман Садекур
25	Ридой Раззак Рахман
26	Ричард Симеон Онучеожо
27	Фарид Музамиль
28	Фейсал Мд Эарм
29	Хамид Мухаммад Наср Уллах
30	Чабу Абель
31	Чикма Клин
32	Чаурасия Бикаш Прасад
33	Чола Стифен
34	Игбонекву Эммануэль

Num	First and last name	1	2	3	5	6	8	9	10	12	13	Homework	credit
1	Акакпо Мофудукпе Маттье	1/1	2	3	5	6	8						
2	Аль Дин А Б М Моснадул												
3	Арему Даниел Олаофе	1/1	2	3	5	6	8	9	10	12	13		
4	Аслонов Шавкат												

	Шарофджонович												
5	Ахмед Мухаммад Фахим	1	2	3	5	6	8						
6	Бадхон Мустафизур Рахман												
7	Банда Форчун												
8	Гаффоров Темурбек Муроджонович												
9	Гембицкий Тимофей Михайлович	1	2	3	5	6							
10	Даиро Джеймс Толувани	1/1	2	3	5	6	8						
11	Зеленщикова Дарья Александровна	1/1	2	3	5	6	8						
12	Ибрагим Муххамад Амир	1/1	2	3	5	6	8						
13	Ибрахим Абдулкуддус Таийе	1/1	2	3	5	6							
14	Кабир Мд Тарек												
15	Калито Калито	1	2										
16	Кумар Принс	1/1	2	3	5	6							
17	Миа Ракиб												
18	Мумба Деррик	1/2	2	3	5	6							
19	Одрохо Ишрак Бин Насим												
20	Оланреважу Шериффдин Оламилекан	1/1	2	3	5	6							
21	Ондала Нандипха	1/	2										
22	Ондала Хилия	1/	2	3									
23	Пири Раджени	1/1	2		5								
24	Рахаман Садекур												
25	Ридой Раззак Рахман												
26	Ричард Симеон Онучеожо												
2	Фарид Музамиль	1/1	2	3	5	6							

7													
28	Фейсал Мд Эарм												
29	Хамид Мухаммад Наср Уллах	1	2	3									
30	Чабу Абель												
31	Чикма Клин												
32	Чаурасия Бикаш Прасад	1/1	2	3	5	6	8						
33	Чола Стифен	1	2	3									
34	Игбонекву Эммануэль	1/1	2	3	5	6	8						

WORKBOOK

Practical work 1

Practical work 2 only 1-st part

Practical work 3

Practical work 4 We don't do this job

Practical work 5

Practical work 6

Practical work 7 We don't do this job

Practical work 8

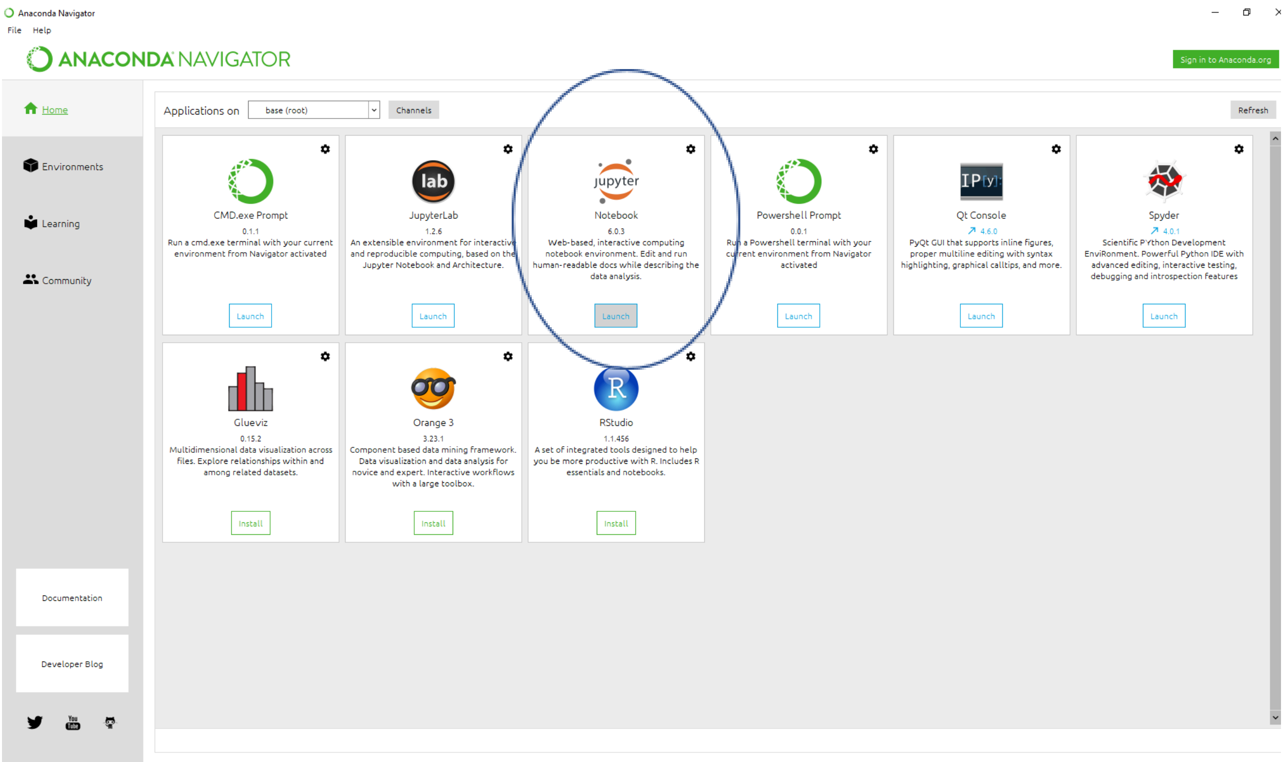
Practical work 9

Practical work 10

Practical work 11 We don't do this job

Python

install Anaconda [Installation — Anaconda documentation](#)



useful python links

https://www.w3schools.com/python/default.asp

https://docs.python.org/3/library/math.html#power-and-logarithmic-functions

1 lesson 09.02.2026

[Lecture 1](#)

Practical work 1

Programming Arithmetic Expressions

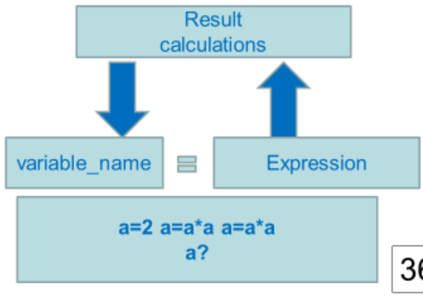
Assignment operator

In the system, you can assign specific values to variables using the assignment operator:

variable_name= value

variable_name= Expression

The assignment operator operates from right to left. First, the expression to the right of the assignment sign is evaluated, and then the result of the action is always stored in a variable.



36

Arithmetic operations

What should be done?	Operator in Python	Example
add	+	2 + 3 = 5
Subtract	-	5 - 2 = 3
Multiply	*	2 * 3 = 6
Divide	/	6 /4 =1.5
Get integer part of division	//	7 // 3 = 2
Get the remainder of a division	%	7 % 3 = 1
Raise to a power	**	2 ** 3 = 8

Library math

You can connect not the entire module, but some part of it. For example, you need to use only one function from the mathematical library math. If you connect the entire library, then more than 40 functions will be added that will take up space. To add some part to the project, use the **from** keyword:

```
from <plugin name> import <function name>
```

Например.

```
from math import sin # only sin is connected
y = sin(5) # operation completed
x = cos(5) # error cos function not connected

from math import * # all math functions included
```


1) $|r|^{5xy} + tg\,3k$ при $k = 2;\ r = 2;\ x = 2;\ y = 1$

```
# 1
k=2
r=2
x=2
y=1
r1=abs(r)**(5*x*y)+tan(3*k)
```

r1= 1023.7089938086152

\# 2 2) $\sqrt{\ln^2 x + 1} + 3\sqrt[3]{x}$ при $x = 0{,}5$

```
x=.5
r2=sqrt(log(x)**2+1)+3*x**(1/3)
print('r2=',r2)
```

r2= 3.597840257340031

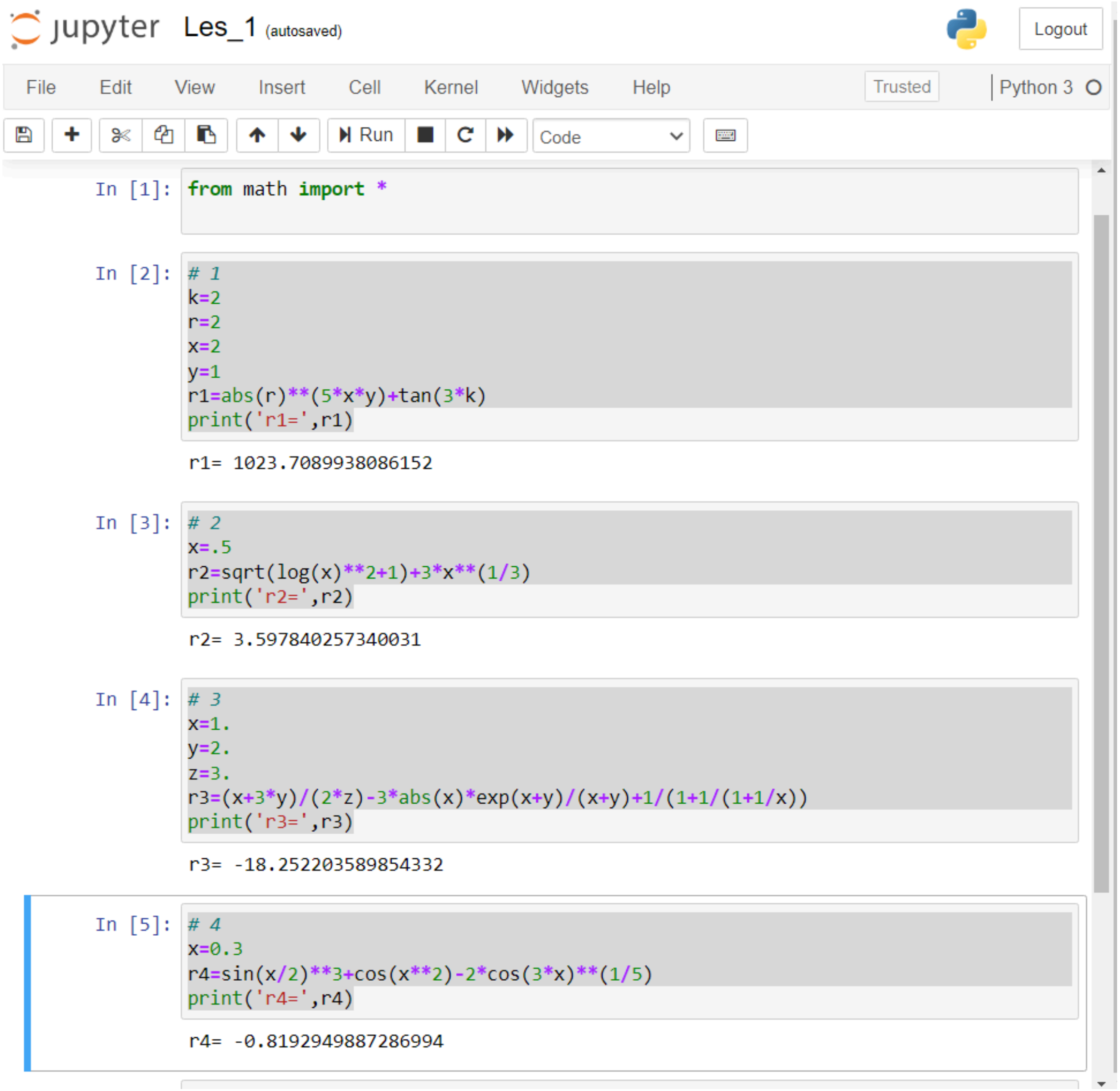
3) $$\frac{x+3y}{2z}-\frac{3|x|e^{x+y}}{x+y}+\frac{1}{1+\frac{1}{1+\frac{1}{x}}}$$ при $x=1;\ y=2;\ z=3$

 # 3

r3= -18.252203589854332

4) $$\sin^3\frac{x}{2}+\cos x^2-2\sqrt[5]{\cos 3x}$$ при $x=0{,}3$

```
# 4
x=0.3
r4=sin(x/2)**3+cos(x**2)-2*cos(3*x)**(1/5)
print('r4=',r4)

r4= -0.8192949887286994
```




```
In [1]: from math import *

In [2]: # 1
        k=2
        r=2
        x=2
        y=1
        r1=abs(r)**(5*x*y)+tan(3*k)
        print('r1=',r1)

        r1= 1023.7089938086152

In [3]: # 2
        x=.5
        r2=sqrt(log(x)**2+1)+3*x**(1/3)
        print('r2=',r2)

        r2= 3.597840257340031

In [4]: # 3
        x=1.
        y=2.
        z=3.
        r3=(x+3*y)/(2*z)-3*abs(x)*exp(x+y)/(x+y)+1/(1+1/(1+1/x))
        print('r3=',r3)

        r3= -18.252203589854332

In [5]: # 4
        x=0.3
        r4=sin(x/2)**3+cos(x**2)-2*cos(3*x)**(1/5)
        print('r4=',r4)

        r4= -0.8192949887286994
```

To complete tasks, you need to create a separate notebook in which to write down the task, solution (program code) and code results.

In your notebook I will note that I saw your program code and it works correctly.

The work will be considered fully completed if you answer all my questions. There are only 10 jobs and all of them must be completed.

Homework

Exercise 1

Write expression evaluation in python

1.	1)	z^{5y}	y=2; z=2	2)	$\sqrt{e^{\sin x}+1}-\cos^3\frac{x}{3}$	x=0,5
	3)	$\frac{5{,}2x}{2\lvert y\rvert}-\frac{4\ln x^2}{5\operatorname{tg}x}$	x=1 y=1,5	4)	$\frac{\operatorname{arctg}\sqrt[3]{x+1}}{x+1{,}3}+3^x$	x=0,3
2.	1)	$x^{y^z}+0{,}3y$	x=2; y=2; z=2	2)	$\sqrt[5]{\ln^2 x+1}+4e^{\sin x}$	x=0,5
	3)	$1+\lvert x\rvert+\frac{x^2+\sqrt{x+1}}{2\cdot 3x}$	x=1,5	4)	$\cos^3 x^2+\frac{\arcsin x^2}{1+\frac{x}{x+1}}$	x=0,2

3.	1)	$(x^y)^{tz}-e^{3x}$	x=2; y=2 z=2; t=2	2)	$\sqrt{0{,}3tx}+\operatorname{ctg}^2\frac{x}{2}$	x=2; t=2
	3)	$\sqrt[7]{\frac{x+3}{3x}}+\cos^3 5x$	x=0,05	4)	$\frac{8\lvert xy\rvert}{3tz}-\ln^3(x+1)$	x=1; y=2 z=1; t=3
4.	1)	$z^{3x}+3x^z-0{,}3$	x=2; z=3	2)	$\sqrt{\ln\lvert\sin^3 x\rvert+1}-e^{-x}$	x=1
	3)	$\frac{0{,}3\cos^2 x^2+1}{2xy}+6$	x=2; y=2	4)	$\frac{\operatorname{arctg} 2x+7}{x+4{,}2}+\sqrt[3]{x}$	x=3

5.	1)	$z^{3x^5}+\ln^2(x+1)$	x=1,5 z=0,2	2)	$\sin^2\lvert x\rvert+\arccos\sqrt[3]{x+1{,}2}$	x=–0,6
	3)	$\frac{x+3yt-4}{0{,}3xyt}+e^{x-1}$	x=1; y=2 t=2	4)	$\frac{\operatorname{ctg}3x-7{,}2}{x+1}-\sqrt{x+0{,}2}$	x=2
6.	1)	$x^{x^x}+(x^x)^x+0{,}04$	x=2	2)	$e^{3x^2+4}-\lvert x\rvert^3+\ln^2 x$	x=0,4
	3)	$\sqrt[3]{\frac{x+1}{x+2}}+\arcsin\sqrt{x}$	x=0,5	4)	$\frac{x+5-3y}{3xyz}+\operatorname{tg}^3 x^2$	x=1; y=2 z=4
7.	1)	$(y^{2z})^3+\ln^3(x+1)$	x=2; y=1 z=2	2)	$\frac{x}{2}+\cos^3 x^3-e^{-3x}$	x=0,3
	3)	$\frac{x+2(x-1)^2}{3xt}-\sqrt{\sin\frac{x}{3{,}3}}$	x=3; t=2	4)	$\frac{\operatorname{arctg}\sqrt[3]{x}-5}{\lvert x\rvert-\frac{x}{x+1}}-3{,}7y$	x=2; y=3

8.	1)	$y^{3^x} - x^3 + e^{\frac{-x}{3}}$	x=2; y=2	2)	$e^{x^2-1} - 2\ln\lvert x+1\rvert - \frac{3}{xy}$	x=2; y=3
	3)	$0{,}8\left(\sin^2\frac{x}{3} - \frac{x+2}{x+1}\right)^3$	x=–2	4)	$\frac{\cos^3 3x^2 + \sqrt{x}}{x+4y} - \sqrt[3]{\frac{x+1}{x-1}}$	x=2; y=3
9.	1)	$4^{x^2} + \sin^2\frac{3x}{7y} + 0{,}3$	x=2; y=3	2)	$\ln^2 x - \lvert\cos(x+3)\rvert$	x=1
	3)	$\frac{\arcsin^3 x + 1 - x}{3x}$	x=0,3	4)	$\frac{\sqrt{x} + \sqrt[3]{x-1} + e^{-3x}}{x + 3{,}5x^2}$	x=2
10.	1)	$(x^y)^3(y^x)^2 + 0{,}7$	x=2; y=2	2)	$\sqrt{x}\sin^2 x + \cos\frac{x^2}{2}$	x=3
	3)	$\frac{\lvert x\rvert - \ln(x+1)}{e^{-x} + 4{,}7x}$	x=4	4)	$\frac{\sqrt[3]{x+7-4y}}{5xy} + \operatorname{arctg}\frac{x+7}{x-4}$	x=2; y=1
11.	1)	$(x^2)^{y^2} - e^{-xy} + 7{,}3$	x=2; y=2	2)	$\sqrt{\ln^2 x + 1} - 3\cos^7 x + 4$	x=0,01
	3)	$\frac{5 - \frac{\lvert x\rvert + \sqrt[3]{x}}{x}}{3 - \frac{x}{1 + \frac{x+1}{x}}}$	x=2	4)	$\frac{x^2+3}{x+2} - e^x\left(\frac{x-1}{x} + 1\right)^4$	x=4
12.	1)	$x^{y^2+4} - e^{-3x}$	x=2; y=3	2)	$2\sin^2\frac{x}{2} - \sin^3\sqrt{x}$	x=3
	3)	$\frac{\ln^3\lvert x\rvert + 4xy}{5xy}$	x=2; y=1	4)	$\frac{\arccos x - 7\ln x^2}{x + 7{,}3}$	x=0,5

Part 2: Programming formulas

Task 1.2. Compute the expression on the computer. The values of the initial data for debugging and manual calculation are indicated in parentheses.

Task options

1.	$\frac{\cos^3 x+3y}{1+2x+3y}$, где $x=s_2-4t$; $y=s_2/t$ ($s_2=12$; $t=3$)
2.	$\frac{u^{-v}+\sqrt{u^4+v^2}}{3u+v+1}$, где $u=a_4+a_4b$; $v=2a_4b$ $a_4=1$; $b=0$)
3.	$\frac{\cos^3 t-r}{5t+2r}$, где $t=4x_2-y/x_2$; $r=x_2+y$ ($x_2=1$; $y=4$)
4.	$\frac{(w-4p)(p^2-w)}{3w+4p}$, где $p=v_2 tgu$; $w=u+3v_2$ ($u=0$; $v_2=6$)
5.	$\frac{\ln\lvert x^2-3\rvert-4y}{x^2+1}$, где $x=2ab_5$; $y=5a-8b_5$ ($a=1$; $b_5=1$)
6.	$\frac{\sin\alpha+3e^{-s}}{1+tg^2\alpha}$, где $\alpha=u_2+v$; $s=2u_2$ ($u_2=0$; $v=0$)
7.	$\frac{\cos\beta-e^{-t}}{t+2\beta t}$, где $\beta=u_2-v$; $t=u_2v-1$ ($u_2=3$; $v=3$)
8.	$\frac{\sqrt[3]{\cos x+7}+4}{5x+t}$, где $x=g_1+3h_2$; $t=\cos^2(2g_1-6h_2)$ ($g_1=3$; $h_2=1$)
9.	$\frac{\sqrt{a^2+\lvert b\rvert}-1}{\lvert a\rvert+\lvert b\rvert}$, где $a=\cos t+s_1$; $b=6t-3s_1$ ($t=0$; $s_1=5$)
10.	$\frac{\sqrt[3]{\lvert u\rvert+2v}}{\cos^4 v+3u}$, где $u=9x-y_5$; $v=arctg\, y_5$ ($x=3$; $y_5=0$)
11.	$\frac{2\cos^3\alpha+3y}{2+3y}$, где $\alpha=s-4t_2$; $y=s/t_2$ ($s=4$; $t_2=1$)
12.	$\frac{ue^{-v}+\sqrt{u^4+3v^2}}{1+\lvert 4u+v\rvert}$, где $u=\alpha+4\beta_1$; $v=2\alpha$ ($\alpha=0$; $\beta_1=1$)
13.	$\frac{\cos^3 t-s}{5t+2s}$, где $t=4x_2-y/x_2$; $s=x_2+y$ ($x_2=1$;$y=4$)
14.	$\frac{(\beta-4p)(p^2-\beta)}{3\beta+4p}$, где $p=v\,tgu_5$; $\beta=u_5+3v$ ($u_5=0$; $v=1$)
15.	$\frac{\ln(x^2+1)-4u}{x^2+2}$, где $x=2ab_4$; $u=5a-8b_4$ ($a=0$; $b_4=1$)
16.	$\frac{\sin 2x+3e^{-s}}{1+arctg^2 4x}$, где $x=u+v_1$; $s=2u$ ($u=0$; $v_1=0$)

Example

$$\frac{\sqrt[3]{\sin^4 x + 2} - 1}{7t + 1}, \text{ где } x = g_2 + h_1; \quad t = \ln\cos^2(g_2 + h_1) \quad (g_2=0;\ h_1=0)$$

```
from math import *


h1 = float(input(“ Enter h1”))
g2 = float(input(“ Enter g2”))


t = log(cos(g2+h1)**2
x = g2+h1

res = ( (sin(x)**4+2)**(1/3)-1)/(7*t+1)
print(“ calculation result =  “, res)
```

---

2 lesson

3.03.2026

Practical work 2

Boolean expressions

Boolean expressions are expressions that evaluate to true or false 0 or 1.

Operator	Designation
>	more
>=	more or equal
<	less

Operator	Designation
<=	less or equal
!=	not equal
==	equal

A B A and B

True	True	True
True	False	False
False	True	False
False	False	False

A	B	A or B
True	True	True
True	False	True
False	True	True
False	False	False

https://docs.python.org/3/library/operator.html?highlight=operation%20logical#

1. x∈[-10,7] → -10≤x≤7 → x≤7 и x≥-10

   a) x>=-10 and x<=7

```
x=int(input(' Enter x'))
logic_1=x>=-10 and x<=7
print(' logic 1  ',logic_1)
```

2. x∈[-5,-1) U (6, 15]

   a) x>=-5 and x< -1 or x>6 and x<=15

Practical work #2

Part 1. Writing logical expressions in an algorithmic language

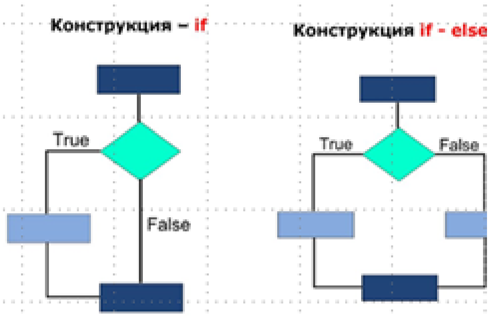
Task 2.1. Write down a logical expression for whether a point belongs to a given interval.

1.	а) $x \in [2;10)$; б) $x \in (-10;3] \cup [7;20)$.	2.	а) $x \in [-20;-10]$; б) $x \in (-3;-1] \cup [0;15]$.
3.	а) $x \in [-2;10)$; б) $x \in (-20;-7] \cup [-2;10]$.	4.	а) $x \in [-1;7)$; б) $x \in (-10;-3] \cup [0;13]$.
5.	а) $x \in (-3;5]$; б) $x \in [-13;-2) \cup [7;10]$.	6.	а) $x \in (5;10)$; б) $x \in [-5;-1] \cup [3;8)$.
7.	а) $x \in [-4;5)$; б) $x \in (-7;0] \cup (5;12)$.	8.	а) $x \in (-3;0)$; б) $x \in [3;6] \cup (10;20)$.
9.	а) $x \in [-18;-5]$; б) $x \in [-5;3) \cup [6;15]$.	10.	а) $x \in [-5;13)$; б) $x \in (-8;2] \cup (3;10)$.
11.	а) $x \in [-30;-10)$; б) $x \in (-10;-5] \cup [-2;2]$.	12.	а) $x \in [10;30)$; б) $x \in (-5;2] \cup (12;19)$.
13.	а) $x \in [-2;3]$; б) $x \in [-10;0) \cup [3;15]$.	14.	а) $x \in (5;8)$; б) $x \in [-15;-5) \cup [-3;15]$.

Part 2

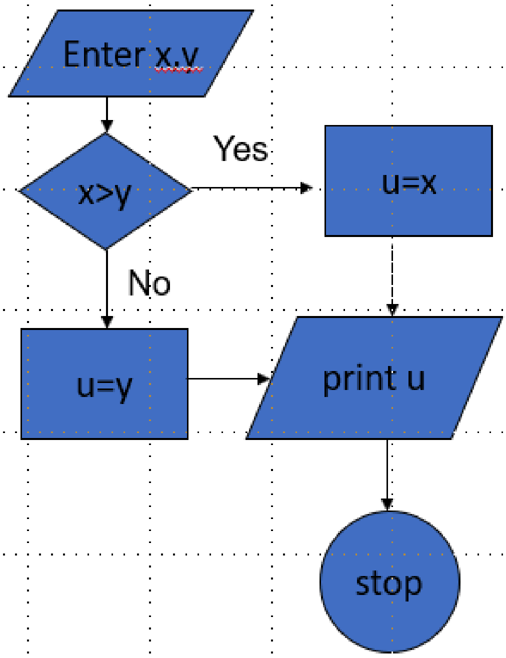
Determine the parity of a number

```
a = 3
if a % 2 == 0:
    print('The number is even')
else:
    print('The number is odd')

The number is odd
```



Practical work #3

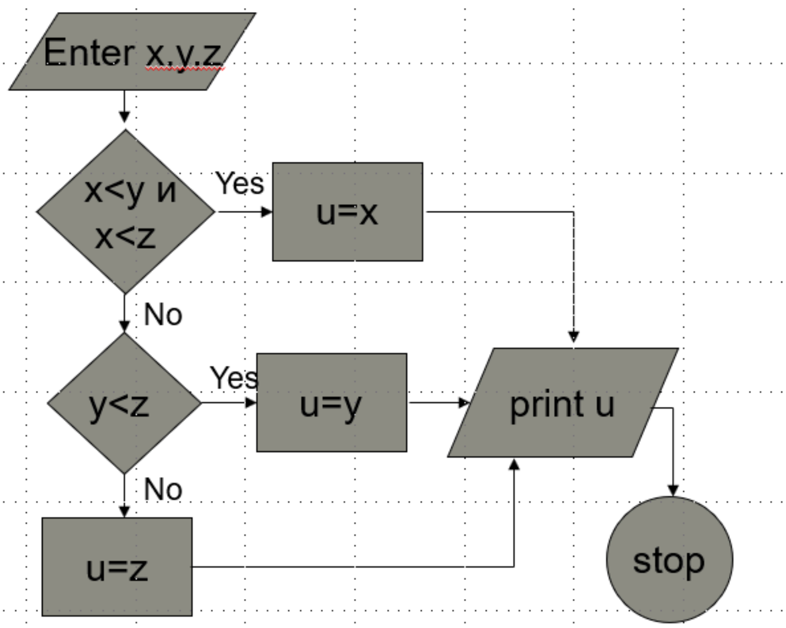
Example 1. Find and print $u=\max\{x,y\}$.




```
x=int(input('Enter x'))
y=int(input('Enter y'))
if x>y :
    u=x
else:
    u=y
print(' u= ',u)
```

Example 2

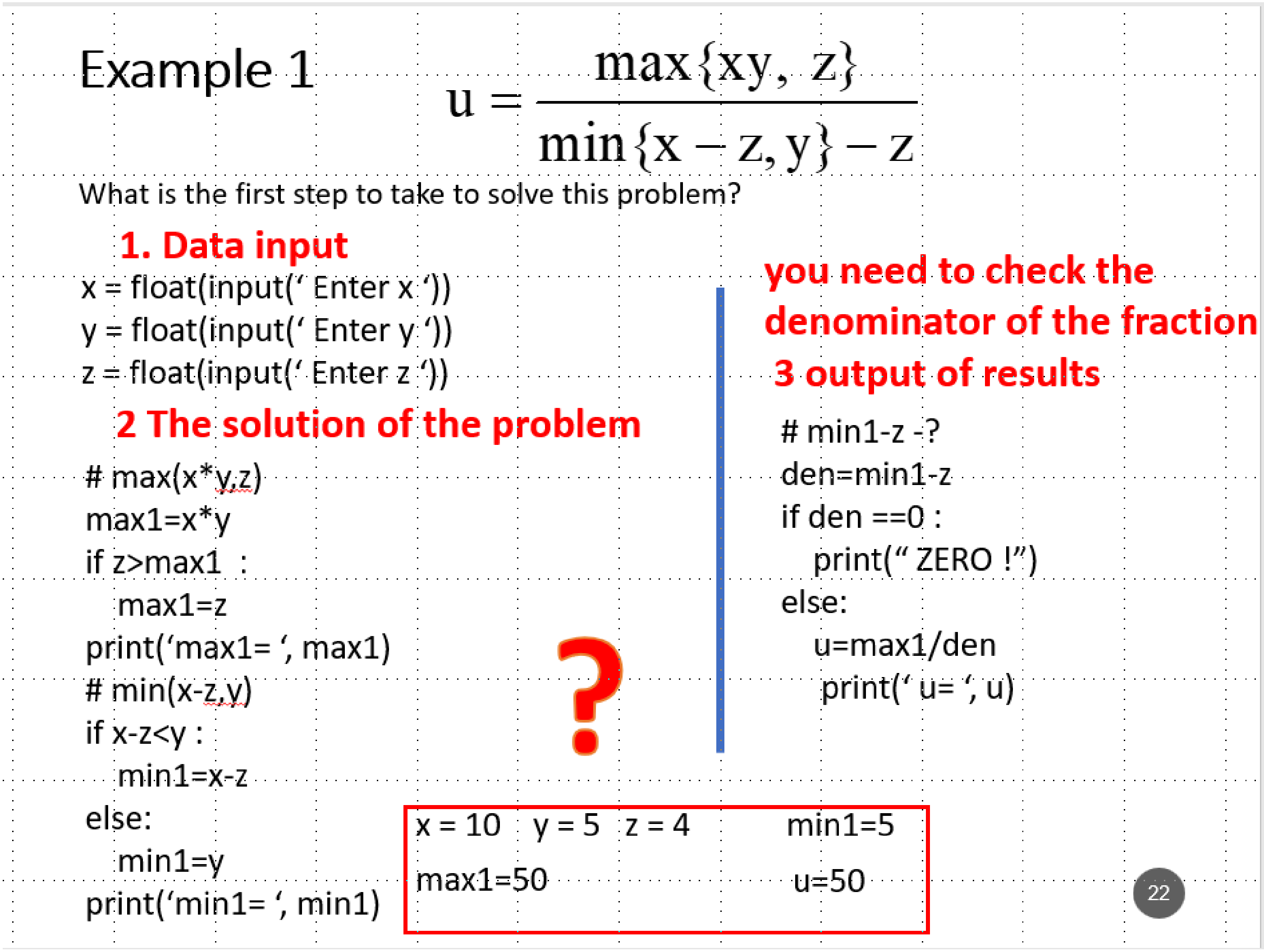
u=max{x,y,z}




```
x=int(input('Enter x'))
y=int(input('Enter y'))
z=int(input('Enter z'))
```

```
if x<y and x<z:
     u=x
elif y<z:
     u=y
else:
     u=z
print(` u= `,u)
```

Example 3



Example 1

$$u = \frac{\max\{xy,\ z\}}{\min\{x - z, y\} - z}$$

```
x = float(input(' Enter x '))
y = float(input(' Enter y '))
z = float(input(' Enter z '))
# max(x*y,z)
max1=x*y
if z>max1 :
   max1=z
print('max1= ', max1)
# min(x-z,y)
if x-z<y :
   min1=x-z
else:
   min1=y
print('min1= ', min1)
```

```
# min1-z -?
den=min1-z
if den ==0 :
   print(" ZERO !")
else:
   u=max1/den
    print(' u= ', u)
```

x = 2   y = 5   z = 1

min1=1

den=0
ZERO !

Task

1. **u=min{x, yz}/max{x+z,y} -?**
2. **u=max{x/z,y}/min{max{x,y}-2, z} -?**
3.

Homework 3

Find u and display all data

1. **u=min{(x+y+z)/x, (x+2y-z)/y}+ max{x,y,z}.**
2. **u= (min{x,y,z+x}+0.5)/max{x,y}.**
3. **u=1+min{(x+y+z)/ max{y,z},xyz}.**
4. **u=min{(x+y+z)/x, (x+2y-z)/y}+ max{x,y,z}**
5. **u = max{x+y,x/y}/min{x,y+z}.**

6. u=min{x+6y,x/y}/max{x+y,x+3y}

7. u = min{x+y,xy}/max{x,yz}

8. u=min{2x/max{y, (x+y-z)/5}, max{y, (x+y-z)/y}}

9. u=max{x+y+z,xyz}/min{2x+2y+2z,xyz}.

10. u=min{x,max{y,z}}/(max{x,y,z}-3) .

11. u=min{(x-y-z)/max{x,y},(x+yz)/max{y,z}}.

12. u= (min{x,y}+0.2)/(max{x,y}+min{x,y,z}).

13. u=max{(xy+z)/min{x,y+z}, (x+2yz)/min{x+y,z}}.

14. u=1+min{(x+y+z)/max{x+y,x+z,y+z},xyz}.

23.03.26

Practical work #5

Lecture 2

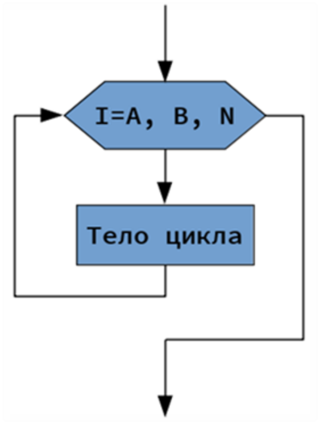
https://www.programiz.com/python-programming/online-compiler/

Lecture 2
https://drive.google.com/file/d/1PcSEvguy5ZrB6YqWFQikMHbN4VyBCsWQ/view?usp=sharing

Python
https://drive.google.com/file/d/1SeR5yL57VfLVqcRBWhWCuZjbwaLRVPq5/view?usp=sharing

Lesson

Loop



1. Imagine that we need to check for parity all numbers in the range from 30 to 41 (range(30,42)) inclusive and display the result for each of them on the screen Function range to write a range [30 to 42), the last number will not be considered.

```
for i in range(30, 42): #  loop
        # Loop body
        if i%2 == 0:
              print('Number', i, 'even')
        else:
              print('Number', i, 'odd')
```

2. Calculate the sum of the roots of numbers for a range of numbers[20, 29]

```
sum_squares = 0
for i in range(20, 30):
        sum_squares = sum_squares + i**(1/2)   #     #sum_squares += i**(1/2)
print(sum_squares) # output: 49.41177778422102
```

3 •Calculate sum and output results as a table

$$s = \sum_{k=1}^{n} u_k, \quad u_k = (-1)^k \frac{x^k}{k}, \quad n = 10$$

X=[0.1, 0.3, 0.4, 0.7, 1]

```
N=int(input(' Enter n '))

X=[0.1,  0.3,  0.4,  0.7,  1]

print(' ----------------------')

print(' |     x   |     s    |')

print(' ----------------------')


for x in X:

   s = 0

   u = -x

   for k in range(1,N+1):

      s =s+ (-1)**k*x**k/k

   print(" | %8.4f | %8.4f |" % (x,s))

print(' ----------------------')
```

```
In [11]: N=int(input(' Enter n '))
         X=[0.1,  0.3,  0.4,  0.7,  1]
         print(' -----------------------')
         print(' |      x    |     s    |')
         print(' -----------------------')

         for x in X:
             s = 0
             u = -x
             for k in range(1,N+1):
                 s =s+ (-1)**k*x**k/k
             print(" | %8.4f | %8.4f |" % (x,s))
         print(' -----------------------')

          Enter n 10
          -----------------------
          |      x    |     s    |
          -----------------------
          |   0.1000 |  -0.0953 |
          |   0.3000 |  -0.2624 |
          |   0.4000 |  -0.3365 |
          |   0.7000 |  -0.5295 |
          |   1.0000 |  -0.6456 |
          -------------------------
```

1. Given x, y, z   Count number of positive values and display them.
2. Given x, y, z   Count number of negative values and their sum.
3. Find the maximum negative number among three numbers x, y, z
4. Ten arbitrary values are entered in a loop. How many are negative among them?

   range(10) >>[0;10) >>[0,1,2,3,4,5,6,7,8,9]

Homework 5

Calculation of the amount.

Task 3. Calculate the sum $S = \sum_{k=1}^{n} u_k$ at n = 10 for given values of x equal to 0.1; 0.3; 0.4; 0.7; 1.0. Print the results in the form of a table, the type of which is given in the task.

Task option**: use the while construction.

1. $u_k = (-1)^k \frac{x^{2k}}{2k}$
`N      X      S`
`---------------------------`
`1     0.1    0.00`
`2     0.3    0.01`
`. . .`
`---------------------------`
	2. $u_k = (-1)^{k+1} \frac{x^{2k+1}}{2k+1}$
	`TABLE`
	`value     function`
	`---------------------------`
	`:   0.1   :   0.00   :`
	`:   0.3   :   0.01   :`
	`. . .`
	`---------------------------`

3. $u_k = (-1)^k \frac{x^{2k}}{k}$
`X1= 0.1    S1=  0.00`
`X2= 0.3    S2=  0.01`
`. . .`
`---------------------------`
	4. $u_k = (-1)^{k+1} \frac{x^{k+1}}{k(k+1)}$
	`X( 1)= 0.1    S( 1)=  0.00`
	`X( 2)= 0.3    S( 2)=  0.01`
	`. . .`

5. $u_k = (-1)^{k+1} \frac{x^k}{k}$

```
           TABLE
     value       value
       X           S
--------------------------
:     0.1    :    0.00    :
:     0.3    :    0.01    :
            . . .
--------------------------
```

6. $u_k = (-1)^k \frac{x^k}{(2k+1)(2k+3)}$

```
--------------------------
           TABLE
          RESULTS
--------------------------
     X= 0.1    S=  0.00
     X= 0.3    S=  0.01
            . . .
--------------------------
```

7. $u_k = (-1)^k \frac{x^k}{5^k}$

```
--------------------------
:    X= 0.1    S=  0.00    :
:    X= 0.3    S=  0.01    :
            . . .
--------------------------
```

8. $u_k = (-1)^{k+1} \frac{k^2 x^k}{2^k}$

```
           TABLE
          *******
     X= 0.1    S=  0.00
     X= 0.3    S=  0.01
            . . .
  **********************
```

9. $u_k = (-1)^k \frac{(k+1)x^k}{3^k}$

```
           TABLE
-------------------------
:     X     :     S     :
-------------------------
     0.1          0.00
     0.3          0.01
            . . .
```

10. $u_k = (-1)^k \frac{x^k}{(k+1)4^k}$

```
           TABLE
     value     sum
       X         S
      0.1       0.00
      0.3       0.01
            . . .
```

11. $u_k = (-1)^k \frac{x^{2k}}{(2k+1)(2k+5)}$

```
        VALUE RESULT
--------------------------
     0.1          0.00
     0.3          0.01
            . . .
--------------------------
```

12. $u_k = (-1)^k \frac{x^k}{(k+1)2^k}$

```
X( 1)= 0.1    S( 1)=  0.00
--------------------------
X( 2)= 0.3    S( 2)=  0.01
--------------------------
            . . .
--------------------------
```

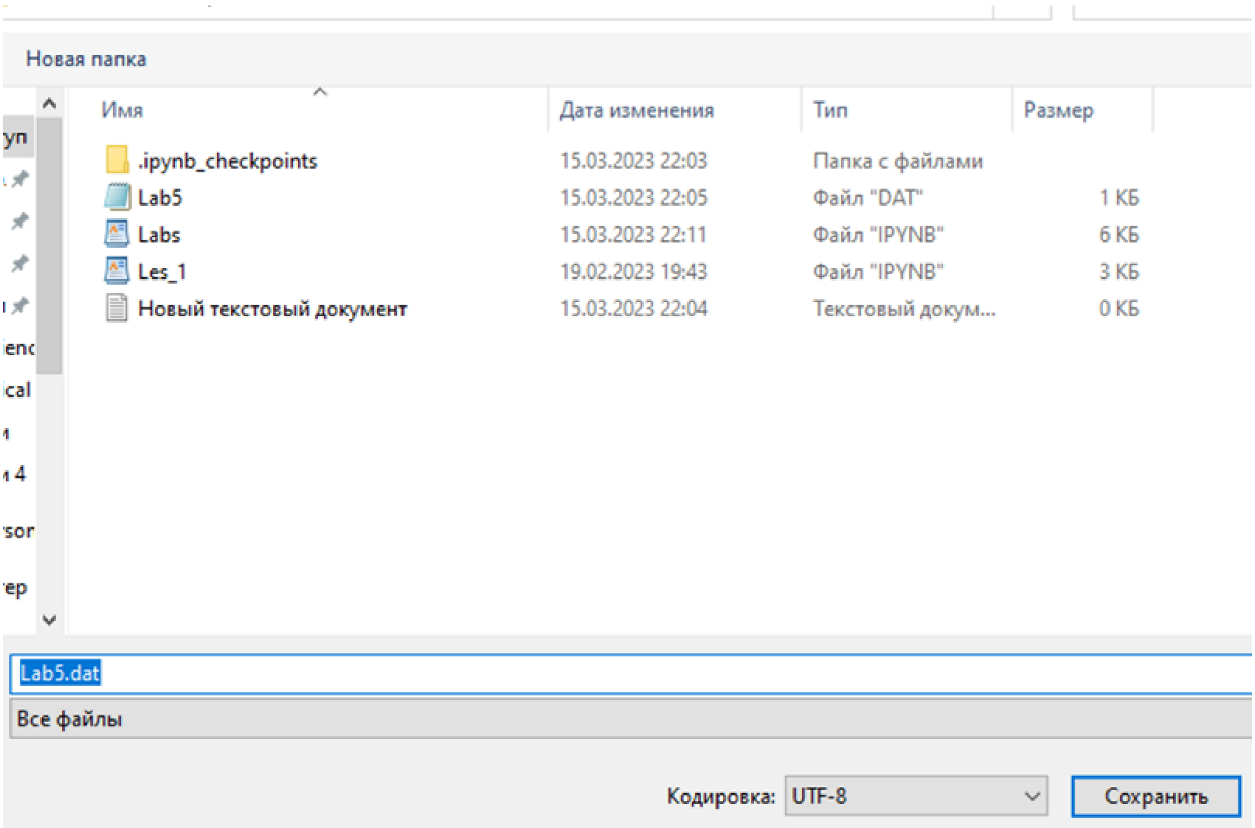
30.03.26

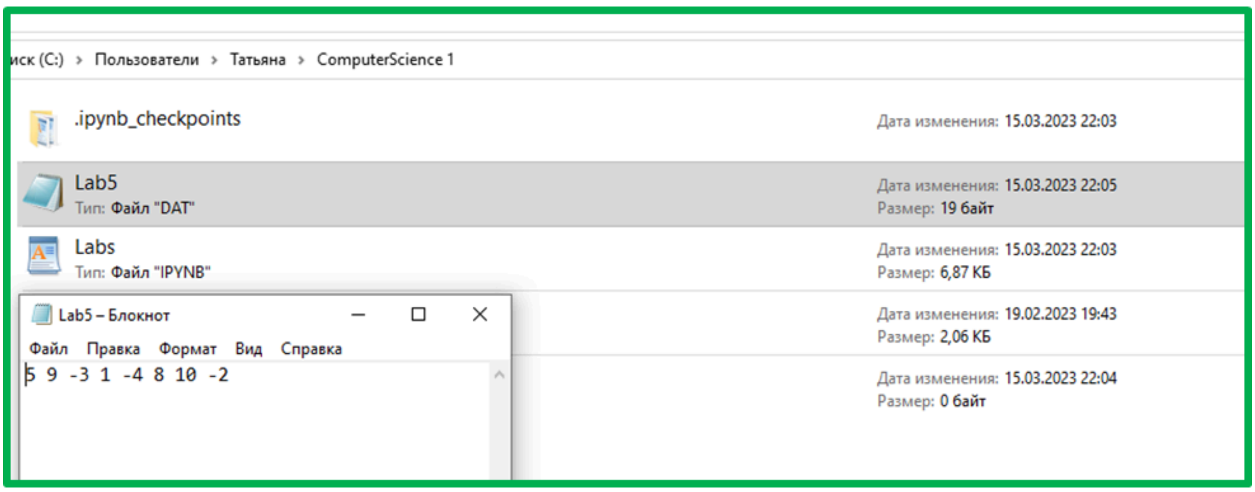
Practical work #6

[Lecture 3]

Task

The file contains an array. Find the value and index of the minimum element among positive ones





```
f=open('Lab5.dat', 'r')
line = f.read()       # read all data in one line
X=[]
for x in line.split():  # division of a string into elements relative to a space
    X.append(int(x))
print('array X')
print(X)
N=len(X)
R=1e25
k=-1
for i in range(N):
    if X[i]<=0:
        continue
    if X[i]<R:
        R=X[i]
        k=i
if k==-1:
    print ('No positive elements')
else:
    print ('X[%d] = %8.3f'%(k,R))
f.close()
```

```
array X
[5, 9, -3, 1, -4, 8, 10, -2]
X[3] =    1.000
```

Homework #6

One-dimensional array.

Read array from file

Task options

1. Given an array C(m), m <15. Print the numbers of those elements of the array that are less than the last one, and their number, as well as print the elements of array C.

2. Given an array A(m), m<12. Print the original array, as well as the value and number of its minimum positive element, as well as the number of positive elements.

3. Given an array A(m), m<15. Replace each negative element of the array with zero and calculate the sum S and the number K of the remaining positive elements. Print the original and converted arrays, as well as the values of S and K.

4. Given an array D(m), m<25. From array D, rewrite elements with odd numbers into array T. Print the elements of array D, array T, the sum and number of elements of array T.

5. Given an array A(m), m<20. Print the original array, as well as the value and number of its maximum negative element, as well as the number of negative elements.

6. Given an array B(m), m<15. All elements with even numbers are doubled, and all elements with odd numbers are replaced with zeros. Print the original and transformed arrays, as well as the number of negative elements in the transformed array.

7. Given an array C(m), m<17. Find the product of all elements preceding the first zero component and the sum of the subsequent ones. Print the original array and the resulting product and sum.

8. Given an array P(m), m<18. Calculate the value of K, equal to the number of negative elements, replacing these elements with zeros. Print the original array, the transformed array, and the value of K.

9. Given an array A(m), m<10. Find the number K of all elements preceding the first component greater than 2, and their product P. Print the original array and the values of K and P.

10. Given an array A(m), m<10. Determine the NM number of the first negative element and the NZ number of the first zero element. Print the original array and the NM and NZ values.

11. Given an array C(m), m<25. Add even and odd numbered elements separately. Print the larger amount and the original array.

12. Given an array C(m), m<30. Arrange its elements so that the positive ones come first, and then all the other elements. Print the original and converted arrays.

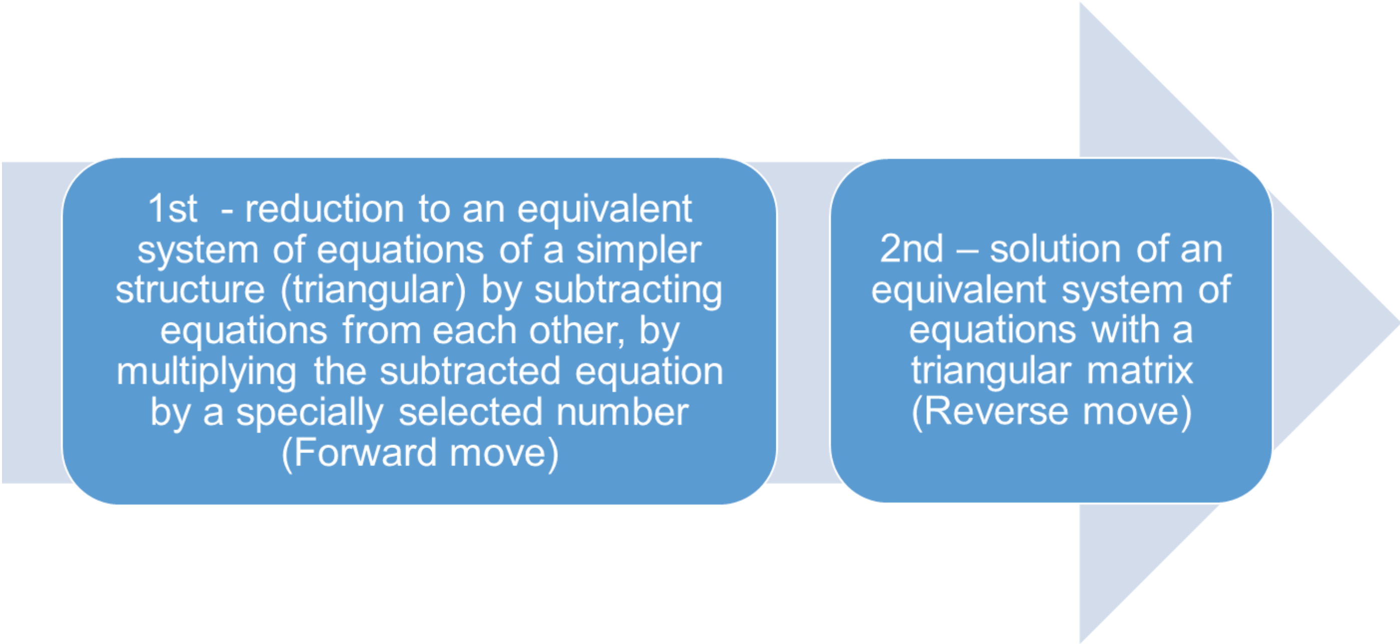
6.04.26

Practical work 8

Lecture 4_1

Lecture 4_2

The Gaussian method refers to exact methods used to solve jointly (consistent) defined systems



example of manual counting

$$\begin{cases} 2x_1 - 3x_2 + x_3 = -1 \\ x_1 + 2x_2 - 6x_3 = -10 \\ 5x_1 + x_2 + x_3 = 3 \end{cases}$$

The augmented matrix has the form

$$\left[\begin{array}{ccc|c} 2 & -3 & 1 & -1 \\ 1 & 2 & -6 & -10 \\ 5 & 1 & 1 & 3 \end{array}\right]$$

direct stage

At the **1st** step of forward stroke:

a) From the 2nd line subtract the 1st, multiplied by 1/2;

b) From the 3rd line subtract the 1st, multiplied by 5/2.

$$\left[\begin{array}{ccc|c} 2 & -3 & 1 & -1 \\ 1 & 2 & -6 & -10 \\ 5 & 1 & 1 & 3 \end{array}\right]$$

$$\left[\begin{array}{ccc|c} 2 & -3 & 1 & -1 \\ 0 & 7/2 & -13/2 & -19/2 \\ 0 & 17/2 & -3/2 & 11/2 \end{array}\right]$$

At the **2nd** step of forward stroke:

a) From the 3rd line subtract the 2nd, multiplied by 17/7.

$$\left[\begin{array}{ccc|c} 2 & -3 & 1 & -1 \\ 0 & 7/2 & -13/2 & -19/2 \\ 0 & 0 & 100/7 & 200/7 \end{array}\right]$$

Please note that in the manual method, calculations are performed with ordinary fractions in order not to lose accuracy, because the Gaussian method is an exact method

Reverse

The equivalent system with a triangular matrix has the form:

$$\begin{cases} 2x_1 - 3x_2 \quad + x_3 = -1 \\ \frac{7}{2}x_2 - \frac{13}{2}x_3 = -\frac{19}{2} \\ \frac{100}{7}x_3 = \frac{200}{7} \end{cases}$$

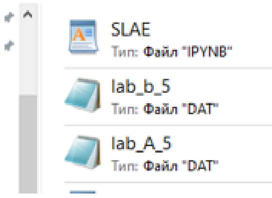
from the 3rd equations:	$x_3 = \frac{200}{7} \Big/ \frac{100}{7} = 2$
from the 2nd equation:	$x_2 = \left(-\frac{19}{2} + \frac{13}{2}2\right) \Big/ \frac{7}{2} = 1$
from 1st equation:	$x_1 = (-1 + 3 - 2)/2 = 0$

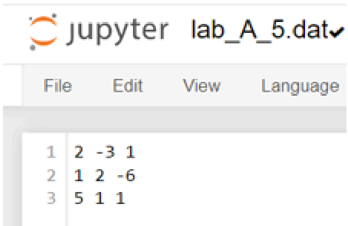
Result: $x_1 = 0, \quad x_2 = 1, \quad x_3 = 2$

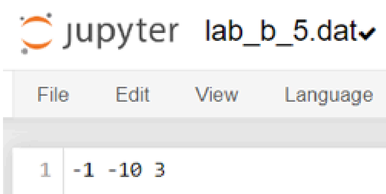
Solving SLAEs in Python

priority2030^ leaders are made, not born

```
In [1]: import numpy as np

In [2]: A=np.genfromtxt('lab_A_5.dat')
        print('array A')
        print(A)

array A
[[ 2. -3.  1.]
 [ 1.  2. -6.]
 [ 5.  1.  1.]]

In [4]: b=np.genfromtxt('lab_b_5.dat')
        print('vector B')
        print(b)

vector B
[ -1. -10.   3.]

In [5]: N=3
        x1 = np.linalg.solve(A,b) # x1= A\b
        print('')
        print('X1')
        for i in range (0,N) :
            print('%6.3f' %x1[i], end=' ')

X1
 0.000  1.000  2.000
```







Solving inverse A in Python

In [6]:
```
print('\nDet(A)=%6.3f'%np.linalg.det(A))
```

```
Det(A)=100.000
```

In [7]:
```
print('Inverse A')
print(np.linalg.inv(A))
```

```
Inverse A
[[ 0.08  0.04  0.16]
 [-0.31 -0.03  0.13]
 [-0.09 -0.17  0.07]]
```

In [9]:
```
x=np.linalg.inv(A).dot(b) #  x=Ainv*b
print('')
print('X1')
for i in range (0,N) :
    print('%6.3f' %x1[i], end=' ')
```

```
X1
 0.000  1.000  2.000
```

Solve SLAEs using the Gaussian method manually and in Python.

Find the inverse of a matrix using Python.

Even variants are solved on a computer using the Seidel method, and odd variants using the simple iteration method. Manual calculation is performed using two methods.

1 $\begin{cases} x_1+5x_2+x_3 = 11 \\ 4x_1-x_2+x_3 = 2 \\ x_1+x_2+6x_3 = 3 \end{cases}$ 1 2 0	**2** $\begin{cases} x_1+5x_2+x_3 = 12 \\ 4x_1-x_2+x_3 = 3 \\ 2x_1+x_2+8x_3 = 12 \end{cases}$ 1 2 1
3 $\begin{cases} x_1+5x_2+x_3 = 13 \\ 4x_1-x_2+x_3 = 4 \\ 3x_1+x_2+10x_3 = 25 \end{cases}$ 1 2 2	**4.** $\begin{cases} x_1+5x_2+x_3 = 14 \\ 4x_1-x_2+x_3 = 5 \\ 4x_1+x_2+12x_3 = 42 \end{cases}$ 1 2 3

5. $\begin{cases} x_1+5x_2+x_3 = 15 \\ 4x_1-x_2+x_3 = 6 \\ 5x_1+x_2+14x_3 = 63 \end{cases}$ 1 2 4	**6.** $\begin{cases} x_1+5x_2+x_3 = 16 \\ 4x_1-x_2+x_3 = 7 \\ 6x_1+x_2+16x_3 = 88 \end{cases}$ 1 2 5
7. $\begin{cases} x_1+5x_2+x_3 = 17 \\ 4x_1-x_2+x_3 = 8 \\ 7x_1+x_2+18x_3 = 117 \end{cases}$ 1 2 6	**8.** $\begin{cases} x_1+5x_2+x_3 = 18 \\ 4x_1-x_2+x_3 = 9 \\ 8x_1+x_2+20x_3 = 150 \end{cases}$ 1 2 7
9. $\begin{cases} x_1+5x_2+x_3 = 19 \\ 4x_1-x_2+x_3 = 10 \\ 9x_1+x_2+22x_3 = 187 \end{cases}$ 1 2 8	**10.** $\begin{cases} x_1+5x_2+x_3 = 20 \\ 4x_1-x_2+x_3 = 11 \\ 10x_1+x_2+24x_3 = 228 \end{cases}$ 1 2 9
11. $\begin{cases} x_1+5x_2+x_3 = 21 \\ 4x_1-x_2+x_3 = 12 \\ 11x_1+x_2+26x_3 = 273 \end{cases}$ 1 2 10	**12.** $\begin{cases} x_1+5x_2+x_3 = 22 \\ 4x_1-x_2+x_3 = 13 \\ 12x_1+x_2+28x_3 = 322 \end{cases}$ 1 2 11

13. $\begin{cases} x_1+5x_2+x_3 = 23 \\ 4x_1-x_2+x_3 = 14 \\ 13x_1+x_2+30x_3 = 375 \end{cases}$ 1 2 12	**14.** $\begin{cases} x_1+5x_2+x_3 = 24 \\ 4x_1-x_2+x_3 = 15 \\ 14x_1+x_2+32x_3 = 432 \end{cases}$ 1 2 13

16.04.26

Practical work 9

Lecture 5

Iterative methods

example solving a system of equations by iterative methods

$$\sum_{\substack{j=1 \\ j\neq i}}^{N} |a_{ij}| \le |a_{ii}|$$

$$\begin{cases} 4x_1 + x_2 + x_3 = 9 \\ x_1 + 2x_2 + 5x_3 = 20 \\ x_1 + 6x_2 - x_3 = 10 \end{cases}$$

Stage I - verification of convergence.

1st equation: |4|>|1|+|1| = 2 – True;

2nd equation: |2|<|1|+|5| = 6 – False;

3rd equation: |-1|<|1|+|6|=7 – False.

$$\begin{cases} 4x_1 + x_2 + x_3 = 9 \\ x_1 + 6x_2 - x_3 = 10 \\ x_1 + 2x_2 + 5x_3 = 20 \end{cases}$$

```
import math
import numpy as np

A=np.array([[4, 1, 1], [1, 6, -1],[1, 2, 5]])
b=np.array([9, 10, 20 ])
N=3
print(' A \n',A)
print(' \n b \n',b)

i_idx, j_idx = np.mgrid[0:N,0:N]
print( np.mgrid[0:N,0:N])
Adiag =  A[i_idx == j_idx]
print(Adiag)
```

Simple iteration method

$$\begin{cases} x_1^{k+1} = (9 - x_2^k - x_3^k)/4 \\ x_2^{k+1} = (10 - x_1^k + x_3^k)/6 \\ x_3^{k+1} = (20 - x_1^k - 2x_2^k)/5 \end{cases}.$$

$$x_i^{k+1} = \frac{1}{a_{ii}}(b_i - \sum_{\substack{j=1 \\ j\neq i}}^{N} a_{ij}x_j^k), \quad i = 1, 2, \dots, N.$$

$$x_i^{k+1} = \frac{b_i}{a_{ii}} - \frac{1}{a_{ii}} \sum_{\substack{j=1 \\ j\neq i}}^{N} a_{ij}x_j^k$$

```
Alpha =  A.transpose()/Adiag
Alpha[i_idx == j_idx] = 0
Alpha =   Alpha.transpose()
print('==================')
print(Alpha)
print('==================')
```

```
Beta = b/Adiag
print(Beta)
print('==================')

x1 = np.zeros(N, dtype = float)
x = Beta
KMAX=100


k=0;
while (np.max(abs(x-x1))>0.001) & (k<KMAX):
   x=x1
   x1= Beta  -  np.dot(Alpha,x)
   k+=1

print("k = ",k, "Error = ", np.max(abs(x-x1)))


print('')
print('X')
for i in range (0,N):
   print('%6.3f'%x1[i],end=' ')
print('')
```

Manual account. Simple iteration method

$$\textbf{1-й шаг } (k=0): \begin{cases} x_1^1 = (9-0-0)/4 = 2.25 \\ x_2^1 = (10-0+0)/6 = 1.67. \\ x_3^1 = (20-0-2\cdot 0)/5 = 4 \end{cases}$$

Вычислим невязку $z_0 = \max\{|2.25-0|,\ |1.67-0|,\ |4-0|\} = 4;$

$$\textbf{2-й шаг } (k=1): \begin{cases} x_1^2 = (9-1.67-4)/4 = 0.83 \\ x_2^2 = (10-2.25+4)/6 = 1.96 \\ x_3^2 = (20-2.25-2\cdot 1.67)/5 = 2.88 \end{cases}.$$

$z_1 = \max\{|0.83-2.25|,\ |1.96-1.67|,\ |2.88-4|\} = 1.42;$

3-й шаг ($k = 2$): $\begin{cases} x_1^3 = (9 - 1.96 - 2.88)/4 = 1.04 \\ x_2^3 = (10 - 0.83 + 2.88)/6 = 2.01 \\ x_3^3 = (20 - 0.83 - 2 \cdot 1.96)/5 = 0.21 \end{cases}$.

$z_2 = \max\{|1.04 - 0.83|, \ |2.01 - 1.96|, \ |3.05 - 2.88|\} = 0.21$.

Ответ: $\tilde{x}_1 \approx 1.04$; $\tilde{x}_2 \approx 2.01$; $\tilde{x}_3 \approx 3.05$.

iterative process Seidel method

$$\begin{cases} x_1^{k+1} = (9 - x_2^k - x_3^k)/4 \\ x_2^{k+1} = (10 - x_1^{k+1} + x_3^k)/6 \\ x_3^{k+1} = (20 - x_1^{k+1} - 2x_2^{k+1})/5 \end{cases}.$$

Начальное приближение: $x_1^0 = x_2^0 = x_3^0$.

$$\textbf{1-й шаг } (k=0):\begin{cases} x_1^1 = (9-0-0)/4 = 2.25 \\ x_2^1 = (10-2.25+0)/6 = 1.29 \\ x_3^1 = (20-2.25-2\cdot 1.29)/5 = 3.03 \end{cases}.$$

$$z_0 = \max\{|2.25-0|,\ |1.29-0|,\ |3.03-0|\} = 3.03;$$

$$\textbf{2-й шаг } (k=1):\begin{cases} x_1^2 = (9-1.29-3.03)/4 = 1.17 \\ x_2^2 = (10-1.17+3.03)/6 = 1.98. \\ x_3^2 = (20-1.17-2\cdot 1.98)/5 = 2.97 \end{cases}$$

$$z_1 = \max\{|1.17-2.25|,\ |1.98-1.29|,\ |2.97-3.03|\} = 1.08;$$

$$\textbf{3-й шаг } (k=2):\begin{cases} x_1^3 = (9-1.98-2.97)/4 = 1.01 \\ x_2^3 = (10-1.01+2.97)/6 = 1.99 \\ x_3^3 = (20-1.01-2\cdot 1.99)/5 = 3.00 \end{cases}.$$

$$z_2 = \max\{|1.01-1.17|,\ |1.99-1.98|,\ |3.00-2.97|\} = 0.16.$$

Ответ: $\tilde{x}_1 \approx 1.01;\quad \tilde{x}_2 \approx 1.99;\quad \tilde{x}_3 \approx 3.00.$

```
import math
import numpy as np

A=np.array([[4, 1, 1], [1, 6, -1],[1, 2, 5]])
b=np.array([9, 10, 20 ])
N=3
print(' A \n',A)
print(' \n b \n',b)

i_idx, j_idx = np.mgrid[0:N,0:N]
print( np.mgrid[0:N,0:N])
Adiag =  A[i_idx == j_idx]
print(Adiag)

Alpha =  A.transpose()/Adiag
Alpha[i_idx == j_idx] = 0
```

```
Alpha =   Alpha.transpose()
print('===================')
print(Alpha)
print('===================')


Beta = b/Adiag
print(Beta)
print('===================')

x = np.zeros(N, dtype = float)

KMAX=100
k=0;
while (z>0.001) & (k<KMAX):
   z=0
   for i in range(0,N):
      y=x[i]
      x[i]= Beta[i]  -  np.dot(Alpha[i],x)
      z=z+abs(x[i]-y)
   k+=1
print("k = ",k, "Error = ", z)
print('')
print('X')
for i in range (0,N):
   print('%6.3f'%x[i],end=' ')
print('')
```

Task Homework 9

Find a solution to your SLAE manually and using the simple iteration method and the Seidel method. Execute the program on the computer using one method specified in the task.
Compare with the result of the Gauss method. Understand the code.

1 $\begin{cases} x_1+5x_2+x_3 = 11 \\ 4x_1-x_2+x_3 = 2 \\ x_1+x_2+6x_3 = 3 \end{cases}$ 1 2 0 **Simple iteration method**	**2** $\begin{cases} x_1+5x_2+x_3 = 12 \\ 4x_1-x_2+x_3 = 3 \\ 2x_1+x_2+8x_3 = 12 \end{cases}$ 1 2 1 **Seidel method**
3 $\begin{cases} x_1+5x_2+x_3 = 13 \\ 4x_1-x_2+x_3 = 4 \\ 3x_1+x_2+10x_3 = 25 \end{cases}$ 1 2 2 **Simple iteration method**	**4.** $\begin{cases} x_1+5x_2+x_3 = 14 \\ 4x_1-x_2+x_3 = 5 \\ 4x_1+x_2+12x_3 = 42 \end{cases}$ 1 2 3 **Seidel method**
5. $\begin{cases} x_1+5x_2+x_3 = 15 \\ 4x_1-x_2+x_3 = 6 \\ 5x_1+x_2+14x_3 = 63 \end{cases}$ 1 2 4 **Simple iteration method**	**6.** $\begin{cases} x_1+5x_2+x_3 = 16 \\ 4x_1-x_2+x_3 = 7 \\ 6x_1+x_2+16x_3 = 88 \end{cases}$ 1 2 5 **Seidel method**
7. $\begin{cases} x_1+5x_2+x_3 = 17 \\ 4x_1-x_2+x_3 = 8 \\ 7x_1+x_2+18x_3 = 117 \end{cases}$ 1 2 6 **Simple iteration method**	**8.** $\begin{cases} x_1+5x_2+x_3 = 18 \\ 4x_1-x_2+x_3 = 9 \\ 8x_1+x_2+20x_3 = 150 \end{cases}$ 1 2 7 **Seidel method**

9. $\begin{cases} x_1+5x_2+x_3 = 19 \\ 4x_1-x_2+x_3 = 10 \\ 9x_1+x_2+22x_3 = 187 \end{cases}$ 1 2 8 **Simple iteration method**	**10.** $\begin{cases} x_1+5x_2+x_3 = 20 \\ 4x_1-x_2+x_3 = 11 \\ 10x_1+x_2+24x_3 = 228 \end{cases}$ 1 2 9 **Seidel method**
11. $\begin{cases} x_1+5x_2+x_3 = 21 \\ 4x_1-x_2+x_3 = 12 \\ 11x_1+x_2+26x_3 = 273 \end{cases}$ 1 2 10 **Simple iteration method**	**12.** $\begin{cases} x_1+5x_2+x_3 = 22 \\ 4x_1-x_2+x_3 = 13 \\ 12x_1+x_2+28x_3 = 322 \end{cases}$ 1 2 11 **Seidel method**
13. $\begin{cases} x_1+5x_2+x_3 = 23 \\ 4x_1-x_2+x_3 = 14 \\ 13x_1+x_2+30x_3 = 375 \end{cases}$ 1 2 12 **Simple iteration method**	**14.** $\begin{cases} x_1+5x_2+x_3 = 24 \\ 4x_1-x_2+x_3 = 15 \\ 14x_1+x_2+32x_3 = 432 \end{cases}$ 1 2 13 **Seidel method**

20.05.2026

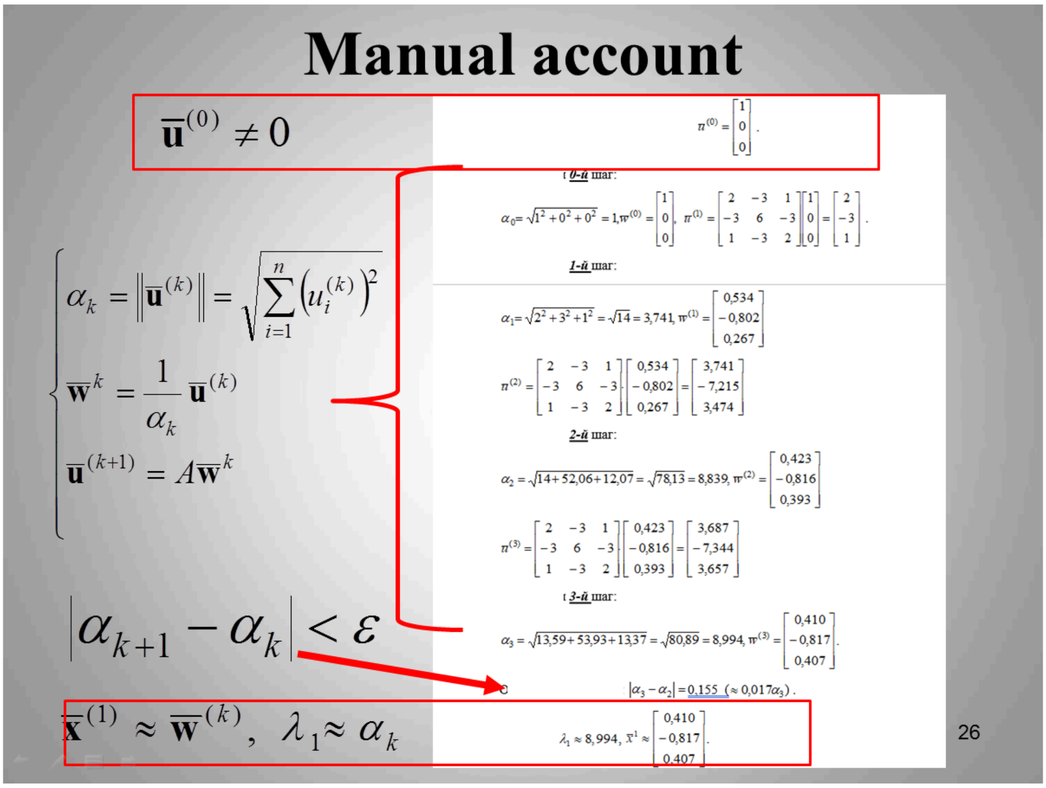
Task Homework 10

Lecture 6

$$A = \begin{bmatrix} 12 & -18 & 6 \\ -18 & 36 & -18 \\ 6 & -18 & 12 \end{bmatrix}$$

Manual account

$$A = \frac{A}{6} = \begin{bmatrix} 2 & -3 & 1 \\ -3 & 6 & -3 \\ 1 & -3 & 2 \end{bmatrix}$$



```
import math
import numpy as np
N=3
A=np.array([[12, -18, 6],[-18, 36, -18], [6, -18, 12]])
A=A/6  # A = np.dot(A,1/6)
print(‵ A')
print(A)
L,X = np.linalg.eig(A)
print('Eigenvalues A')
print(L)
print('Eigenvectors A')
for i in range (0,N):
      for j in range (0,N):
           print('%6.3f'%X[i][j],end='  ‵)
      print('')
```

Python

```
import math
import numpy as np
N=3
A=np.array([[12, -18, 6],[-18, 36, -18], [6, -18, 12]])
A=A/6  # A = np.dot(A,1/6)
print(‘ A')
print(A)
L,X = np.linalg.eig(A)
print('Eigenvalues A')
print(L)
print('Eigenvectors A')
for i in range (0,N):eigenvectors
    for j in range (0,N):
        print('%6.3f'%X[i][j],end=' ')
    print('')
```

```
  2.00 -3.00  1.00
 -3.00  6.00 -3.00
  1.00 -3.00  2.00
Eigenvalues A
 9.00   1.00   0.00
Eigenvectors A
     0.4082      -0.7071    0.5774
     -0.8165       0.0000   0.5774
     0.4082       0.7071    0.5774
```

x= $0.4082^2 + (-0.8165)^2 + 0.4082^2 \approx$ **1.0000**

1. **Computing the length of a vector from the eigenvalue problem.**

Given a matrix X - a matrix of . To extract the first column from it is to refer to the index of this column (in python it is 0)

```
eigX = X[:,0]

lenX= sum(eigX**2)**(½)

print(‘ length X = ‘, lenX)
```

To create a function that can be called to calculate the length of any vector

```
def lengthVect(x):

    lenX= sum(x**2)**(½)

    return lenX
```

You can call a function by specifying its name and parameter.

```
print(‘ length X = ‘, lengthVect(X[:,0]))
```

2 create a matrix of random NumPy integers

np.random.randint (low, high, (rows, columns))

A= np.random.randint (-10, 10, (4, 4))

A

3. Solve a problem

Lesson 4.05

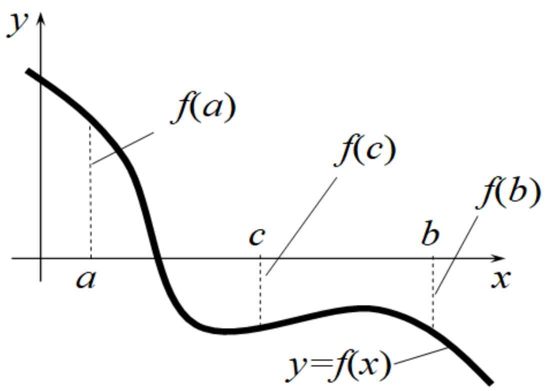
Task Homework 12

Lecture 6

Methods for solving nonlinear equations

1. Bisection method

The bisection method is an approximation method to find the roots of the given equation by repeatedly dividing the interval. This method will divide the interval until the resulting interval is found, which is extremely small.



1. By signs - there is a single root

2. Calculate the midpoint between a and b

3. Calculate the values of the functions at all three points

4. Discard the part where the function does not change its sign

5. We carry out points 2-4 until the end condition is met

f(c)< eps or f(c) = 0 or |b-a|< eps

Example 1

$P_3(x) = x^3 - 3{,}2x^2 + 4{,}84x - 2{,}928$.

Using the bisection method, find the root of the equation on the segment **[0,3]**

k	a	b	$c=(a+b)/2$	$f(a)$	$f(b)$	$f(c)$	$b-a$
0	0	3.0	1.5	-2.92	9.79	0.5	3.0
1	0	1.5	0.75	-2.92	0.51	-0.67	1.5
2	0.75	1.5	1.125	-0.67	0.51	-0.11	0.75
3	1.125	1.5	1.3125	-0.11	0.51	0.173	0.375
4	1.125	1.3125	**1.218**	-0.11	0.173	0.0267, что меньше ε=0.1	0.187≈ 0.06(b-a)

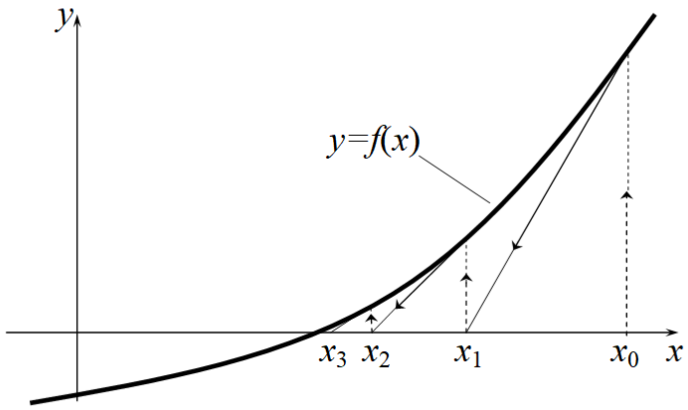
Ответ: $x \approx 1.218$

```
import math
def F(x):
   return x**3-3*x**2+9*x-5
a=float(input(" Enter a: ")) # a=0
b=float(input(" Enter b: ")) # b=3
eps=float(input(" Enter eps: ")) # eps=0.001
k=1
while (abs(b-a)>eps) & (k<100):
   x=(a+b)/2
   k += 1
   if(F(x) == 0): break
   if (F(a)*F(x)>0):
      a = x
   else:
      b = x
print('K= ', k,' X=',x)
print('F(x)= ', F(x))
```

Iteration methods. Newton's method

Newton's method is a technique for solving equations of the form f(x)=0 by successive approximation. The idea is to pick an initial guess x0 such that f(x0) is reasonably close to 0. We then find the equation of the line tangent to y=f(x) at x=x0 and follow it back to the x axis at a new (and improved!) guess x1.

$$x_{k+1} = x_k - f(x_k)/f'(x_k)$$



Example 2

$$x_{k+1} = x_k - \frac{f(x_k)}{f'(x_k)}, \text{ где } f(x)=x^3\text{-}3.2x^2+4.84x\text{-}2.928, \quad f'(x)=3x^2\text{-}6.4x+4.84,$$

k	x_k	$f(x_k)$	$f'(x_k)$	x_{k+1}	$\lvert x_{k+1}-x_k \rvert$
0	3.0	9.79	12.64	2.22	0.78
1	2.22	3.01	5.45	1.67	0.55
2	1.67	0.89	2.52	1.32	0.35
3	1.32	0.18	1.62	**1.20**	0.12
4	1.20	0	критерий окончания расчета $\lvert f(x_k) \rvert < \varepsilon = 0.1$		

Ответ: $x \approx 1.20$

Homework 12

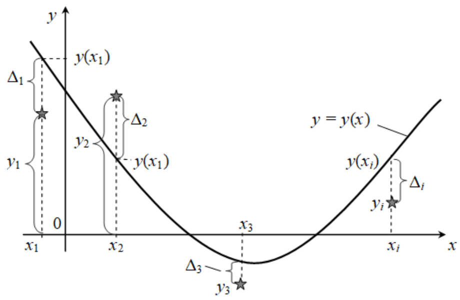
Find the solution to the nonlinear equation. Calculate manually for bisection and Newton's method on the segment [0,3]. We solve the problems manually using two methods. Create a program for one option. and the program using one: the bisection method - even variants, Newton's method - odd variants

1. $P_3(x) = -0{,}101 + 1{,}21x - 2{,}1x^2 + x^3$ **Bisection method**	2. $P_3(x) = -0{,}208 + 1{,}44x - 2{,}2x^2 + x^3$ **Newton's method**
3. $P_3(x) = -0{,}327 + 1{,}69x - 2{,}3x^2 + x^3$ **Newton's method**	4. $P_3(x) = -0{,}464 + 1{,}96x - 2{,}4x^2 + x^3$ **Bisection method**
5. $P_3(x) = -0{,}625 + 2{,}25x - 2{,}5x^2 + x^3$ **Bisection method**	6. $P_3(x) = -0{,}816 + 2{,}56x - 2{,}6x^2 + x^3$ **Newton's method**
7. $P_3(x) = -1{,}043 + 2{,}89x - 2{,}7x^2 + x^3$ **Newton's method**	8. $P_3(x) = -1{,}312 + 3{,}24x - 2{,}8x^2 + x^3$ **Bisection method**
9. $P_3(x) = -1{,}629 + 3{,}61x - 2{,}9x^2 + x^3$ **Bisection method**	10. $P_3(x) = -2 + 4x - 3x^2 + x^3$ **Newton's method**
11. $P_3(x) = -2{,}431 + 4{,}41x - 3{,}1x^2 + x^3$ **Newton's method**	12. $P_3(x) = -2{,}928 + 4{,}84x - 3{,}2x^2 + x^3$ **Bisection method**

Task Homework 13

Least square method LSM

The concept of the method of least squares



$$\Delta_i = y(x_i) - y_i$$

deviations of the values of the curve values and points of the experiment along the y coordinate

The least squares method is the process of finding the best-fitting curve or line of best fit for a set of data points by reducing the sum of the squares of the offsets (residual part) of the points from the curve.

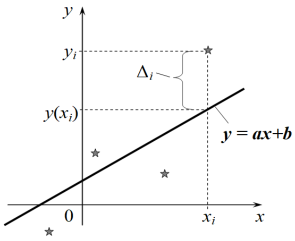
The simplest case is the representation of the required dependence in the form of a polynomial y(x)

$$y(x) = \sum_{k=0}^{m} c_k x^k = c_0 + c_1 x + \ldots + c_m x^m$$

given degree m, which best reflects the dependence y(x).

The minimum criterion is the minimum of the sum of squared deviations for all points of the experiment $S=\sum_{i=1}^{n}(\Delta Y_i)^2 \to \min$

for ***y=ax+b***



$$\begin{cases}\left(\sum_{i=1}^{n}x_i^2\right)a+\left(\sum_{i=1}^{n}x_i\right)b=\sum_{i=1}^{n}x_iy_i\\ \left(\sum_{i=1}^{n}x_i\right)a+nb=\sum_{i=1}^{n}y_i\end{cases}$$

An example of a practical work

It is required to determine the optimal line for the given 12 points on the plane with coordinates. The coordinates of the points are presented in the table:

i	1	2	3	4	5	6	7	8	9	10	11	12
x_i	1	2	3	4	4	5	3	6	10	8	9	7
y_i	2	2,5	2	4	4,5	5	4	5	9	7	8	7

Manual account n=4

i	x_i	y_i	x_i^2	x_iy_i
1	1	2	1	2
2	2	2,5	4	5
3	3	2	9	6
4	4	4	16	16
Σ	**10**	**10,5**	**30**	**29**

$$\begin{cases}\left(\sum_{i=1}^{n}x_i^2\right)a+\left(\sum_{i=1}^{n}x_i\right)b=\sum_{i=1}^{n}x_iy_i\\ \left(\sum_{i=1}^{n}x_i\right)a+nb=\sum_{i=1}^{n}y_i\end{cases}$$

$$\begin{cases}30a + 10b = 29\\ 10a + 4b = 10{,}5\end{cases}$$

Cramer's method for solving this SLAE

$a=\Delta_a/\Delta$, $b=\Delta_b/\Delta$, где $\Delta=\begin{vmatrix}30 & 10\\ 10 & 4\end{vmatrix}=30\cdot 4-10^2=20$,

$\Delta_a=\begin{vmatrix}29 & 10\\ 10{,}5 & 4\end{vmatrix}=29\cdot 4-10{,}5\cdot 10=11$, $\Delta_b=\begin{vmatrix}30 & 29\\ 10 & 10{,}5\end{vmatrix}=30\cdot 10{,}5-10\cdot 29=25$

$a=11/20=0{,}55$, $b=25/20=1{,}25$. $y=0{,}55\cdot x+1{,}25$.

0

```
import math
import numpy as np
import matplotlib.pyplot as plt
```

```
import math
import numpy as np
import matplotlib.pyplot as plt

YY=[]
for x in X:
   YY.append(float(a*x+b))
Z=(np.array(Y)-np.array(YY))**2
SS=sum(Z)
print('SS=',SS)
plt.title('LSM')
plt.xlabel('X')
plt.ylabel('Y')
plt.scatter(X,Y,c='r',marker='*')
# draw a graph
plt.plot(X, YY,'b')
#  show plot
plt.show()
```

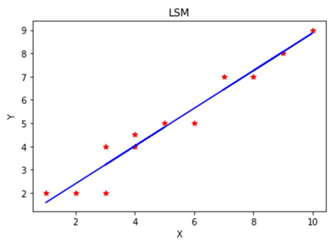
X//

[1. 2. 3. 4. 4. 5. 3. 6. 10. 8. 9. 7.]

Y

[2. 2. 2. 4. 4.5 5. 4. 5. 9. 7. 8. 7.]

y= 0.8094795539033457 *x+ 0.7760223048327136
SS= 3.474442379182155



Task Homework 13

It is required to determine the optimal line for the given 12 points on the plane with coordinates. The coordinates of the points are presented in the table.

For the first four points, you need to solve them manually.

Homework

Solve a problem

Example

Given a matrix A(N,M). In each line, find the sum of the modules of the elements. Determine in which of the rows the largest of these sums is obtained. Output all elements of this string. Each stage is solved using a subroutine that works with a vector. Vector "cut" from the matrix.

 # 1 input data

```
import numpy as np

N = int(input(" Enter N "))

M = int(input(" Enter M "))

A= np.random.randint (-10, 10, (N, M))

print(' matrix A(N,M)', A)


# 2 In each line, find the sum of the modules of the elements

# function find the sum of the modules of the elements

def sumAbs(line):

    return sum(abs(line))


x= np.zeros(N)

for i in range(N):

    x[i] = sumAbs(A[i,:])

print(' sum of the modules of the elements ', x)


#3 Determine in which of the rows the largest of these sums is obtained


def maxValue(x):

    maxX = x[0]
```

```
    index = 0
    for i in range(len(x)):
        if x[i] > maxX:
            maxX = x[i]
            index = i
    return maxX, index

[Max_value, index_value] = maxValue(x)
print('max =', Max_value, 'index', index_value)

# print line
print('row ',A[index_value,:])
```

jupyter SLAE Last Checkpoint: 04.04.2023 (autosaved)

File Edit View Insert Cell Kernel Widgets Help

In [48]:
```python
N = int(input(" Enter N "))
M = int(input(" Enter M "))
A= np.random.randint (-10, 10, (N, M))
print(' matrix A(N,M)', A)
```

```
Enter N 4
Enter M 4
matrix A(N,M) [[  7  -5 -10   7]
[  1   4   4   3]
[  9  -2   4  -2]
[  1  -6   4  -3]]
```

In [49]:
```python
#In each line, find the sum of the modules of the elements

def sumAbs(line):
    return sum(abs(line))
```

In [52]:
```python
x= np.zeros(N)
for i in range(N):
    x[i] = sumAbs(A[i,:])
print(' sum of the modules of the elements ', x)
```

```
sum of the modules of the elements  [29. 12. 17. 14.]
```

In [54]:
```python
# Determine in which of the rows the largest of these sums is obtained
def maxValue(x):
    maxX = x[0]
    index = 0
    for i in range(len(x)):
        if x[i] > maxX:
            maxX = x[i]
            index = i
    return maxX, index
```

In [55]:
```python
[Max_value, index_value] = maxValue(x)
print('max =', Max_value, 'index', index_value)
```

```
max = 29.0 index 0
```

In [56]:
```python
# print line
print('row ',A[index_value,:])
```

```
row  [  7  -5 -10   7]
```

Homework

1. Given a matrix A(N,M) and a number k entered from the keyboard. Find the minimum element in the kth column. In the line where this element is located, find the sum of positive elements. Each stage is solved using a subroutine that works with a vector. Vector "cut" from matrix.
2. Given a matrix A(N,M) and a number k entered from the keyboard. Find the maximum element in the k-th line. In the column where this element is located, find the sum of negative elements. Each stage is solved using a subroutine that works with a vector. Vector "cut" from the matrix.
3. Given a matrix A(N,M). Find the sum of positive elements in each column. Determine in which of the columns the smallest of these sums is obtained. Display all elements

of this column. Each stage is solved using a subroutine that works with a vector. Vector "cut" from the matrix.

4. Given a matrix A(N,M), which is entered from the file. In each line, find the sum of negative elements. Determine which line produces the smallest of these amounts. Print all elements of this string. Each stage is solved using a subroutine that works with a vector. The vector is “cut” from the matrix.
5. Given a matrix A(N,M), which is entered from the file. In each column, find the product of non-zero elements. Determine which column produces the smallest of these products. List all elements of this column. Each stage is solved using a subroutine that works with a vector. The vector is “cut” from the matrix.
6. Given a matrix A(N,M), which is entered from the file. In each column, find the sum of the moduli of the elements. Determine which column produces the smallest of these sums. List all elements of this column. Each stage is solved using a subroutine that works with a vector. The vector is “cut” from the matrix.
7. Given a matrix A(N,M), which is entered from the file. In each column, find the sum of positive elements. Determine which column produces the smallest of these sums. List all elements of this column. Each stage is solved using a subroutine that works with a vector. The vector is “cut” from the matrix.
8. Given a matrix A(N,M), which is entered from the file, and a number k, entered from the keyboard. Find the maximum element in the k-th row. In the column where this element is located, find the product of non-zero elements. Each stage is solved using a subroutine that works with a vector. The vector is “cut” from the matrix.
9. Given a matrix A(N,M), which is entered from the file, and a number k, entered from the keyboard. Find the maximum element in the k-th row. In the column where this element is located, find the sum of the modules of the elements. Each stage is solved using a subroutine that works with a vector. The vector is “cut” from the matrix.
10. Given a matrix A(N,M), which is entered from the file, and a number k, entered from the keyboard. Find the minimum element in the k-th row. In the column where this element is located, find the sum of the negative elements. Each stage is solved using a subroutine that works with a vector. The vector is “cut” from the matrix.
11. Given a matrix A(N,M), which is entered from the file, and a number k, entered from the keyboard. Find the maximum element in the k-th row. In the column where this element is located, find the sum of the negative elements. Each stage is solved using a subroutine that works with a vector. The vector is “cut” from the matrix.
12. Given a matrix A(N,M), which is entered from the file, and a number k, entered from the keyboard. Find the maximum element in the kth column. In the line where this element is located, find the product of nonzero elements. Each stage is solved using a subroutine that works with a vector. The vector is “cut” from the matrix.

Credit!

For the credit, you need to bring a notebook in which there will be completed tasks: the condition of the problem, the solution - the program code and the result. For problems of numerical methods, manual calculation is also needed.