

## Комп'ютерний практикум № 2 УМОВНІ ІНСТРУКЦІЇ В PYTHON (IF)

**Мета роботи:** ознайомитися з алгоритмами розгалуженої структури та їх реалізацією. *Об'єкт дослідження* – умовний оператор (процедурна інструкція *if*), алгоритми розгалуженої структури, виключення у вигляді рядків.

### Завдання

- Відповідно до свого варіанту:
  - визначити умови;
  - за допомогою формул описати варіанти виконання необхідний дій;
  - написати програму, яка розв'яже завдання.
  - організувати введення даних з клавіатури, виведення у консоль.
- Скласти звіт і захистити його по роботі.

Захист роботи включає в себе демонстрацію працездатності програми на різних вхідних даних.

Вимоги: не можна використовувати масиви, цикли, власні функції.

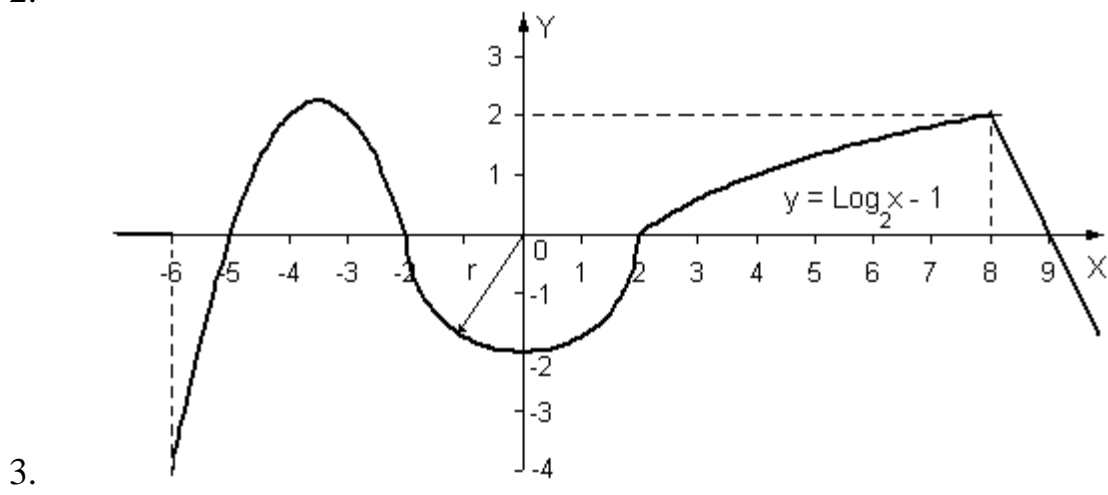
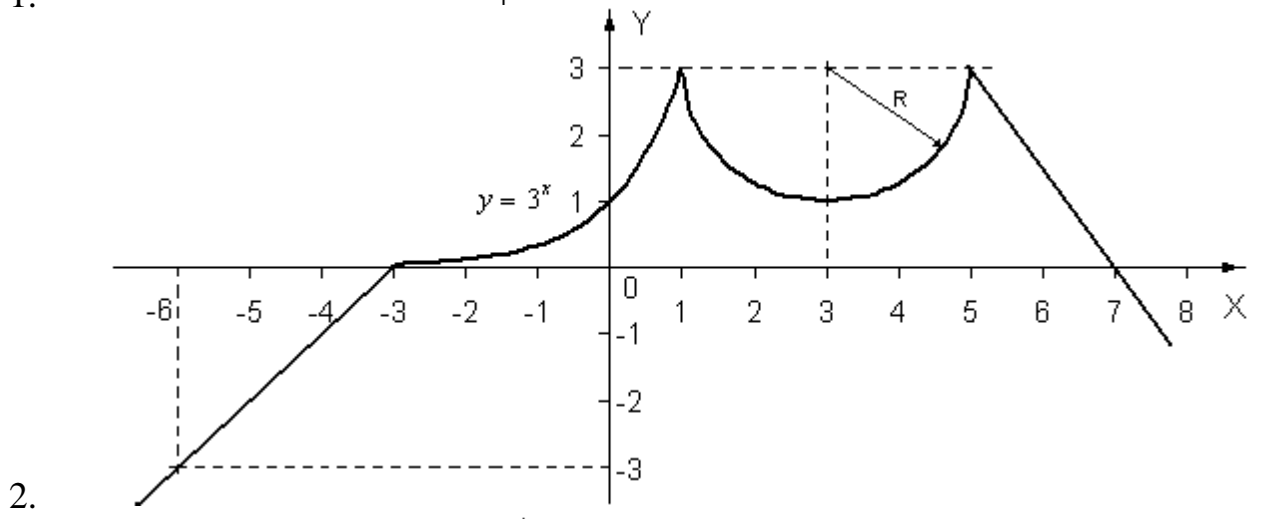
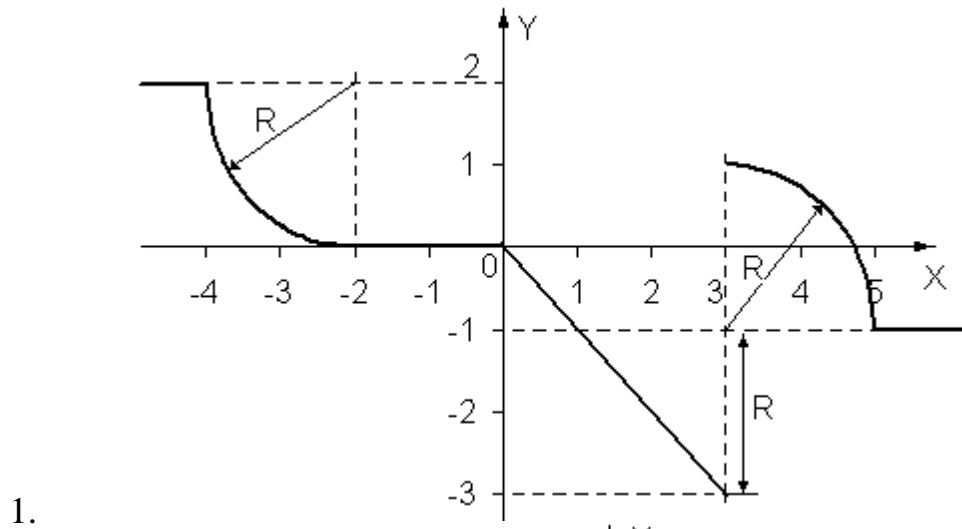
### Варіанти

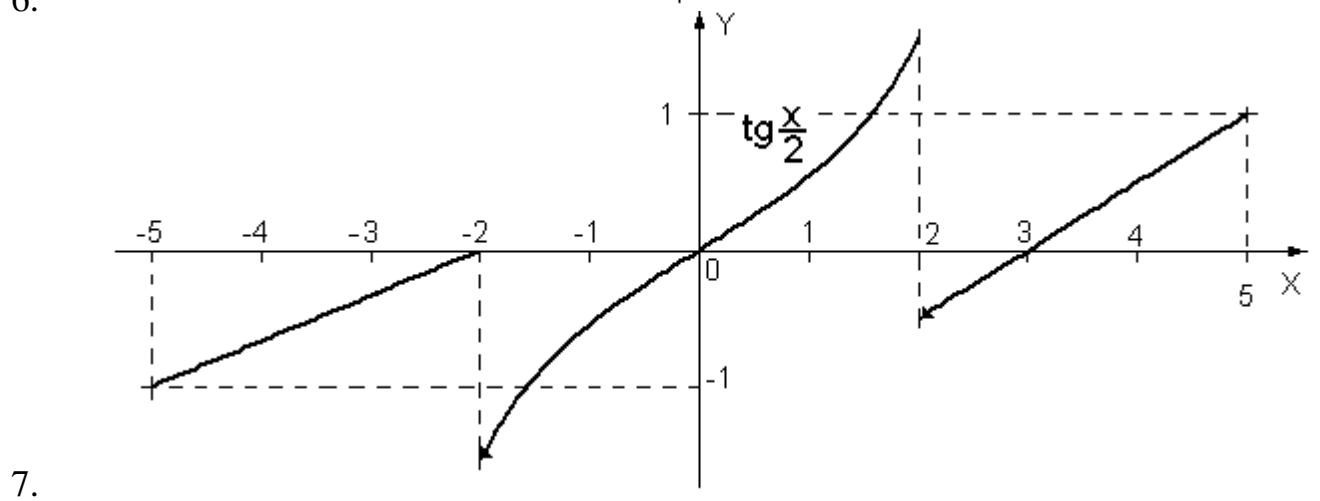
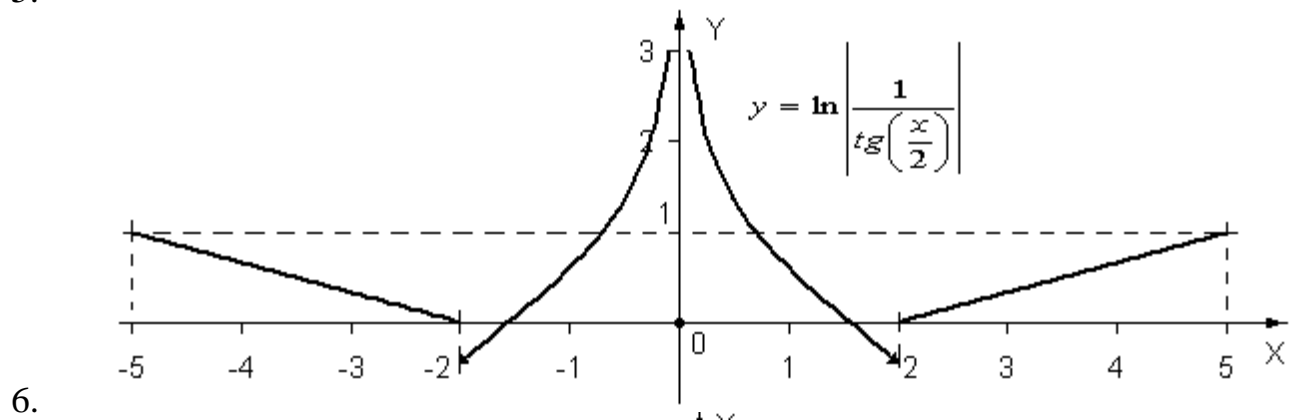
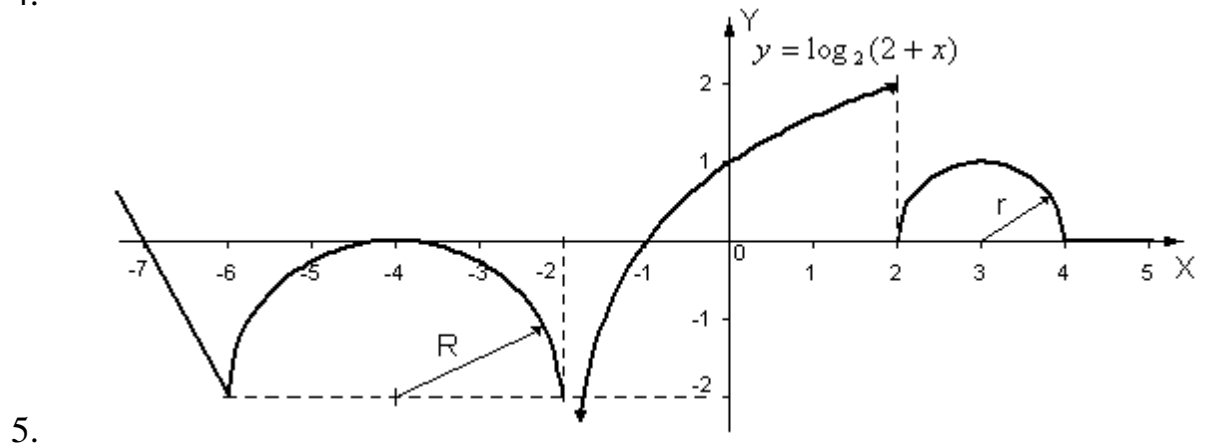
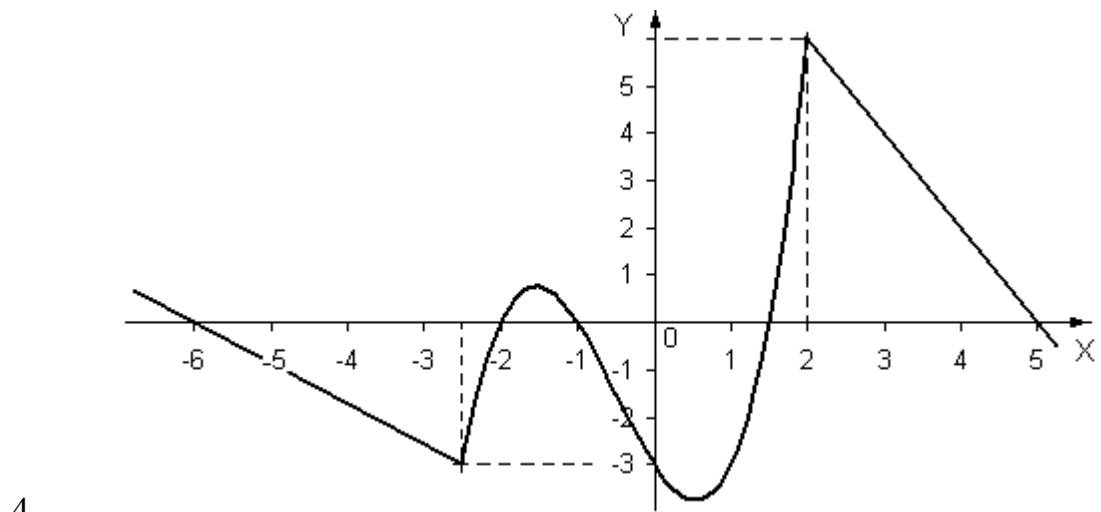
**Завдання 2.1.** Напишіть програму, яка обчислює значення визначеної функції, якщо на вхід подають дійсні числа. Задано функцію  $f(x)$  з параметром  $x$  вигляду:

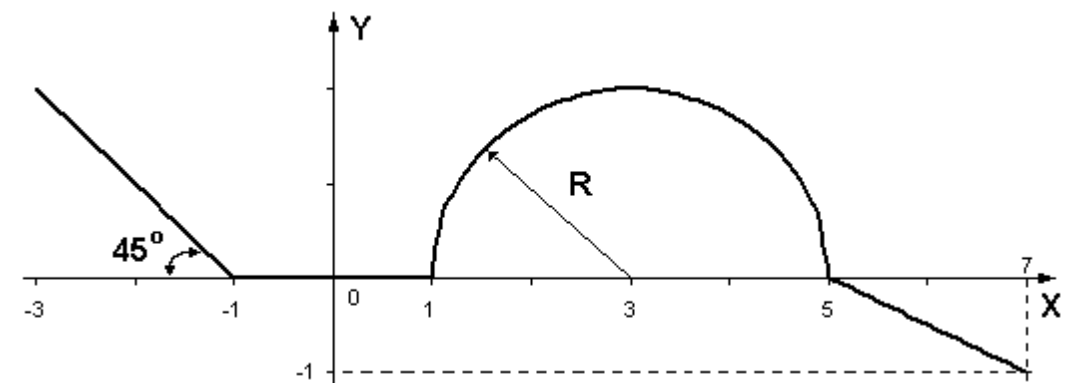
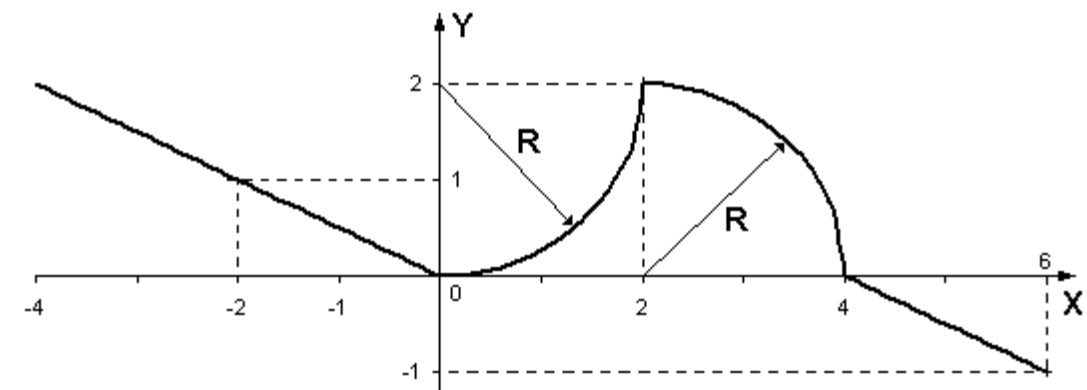
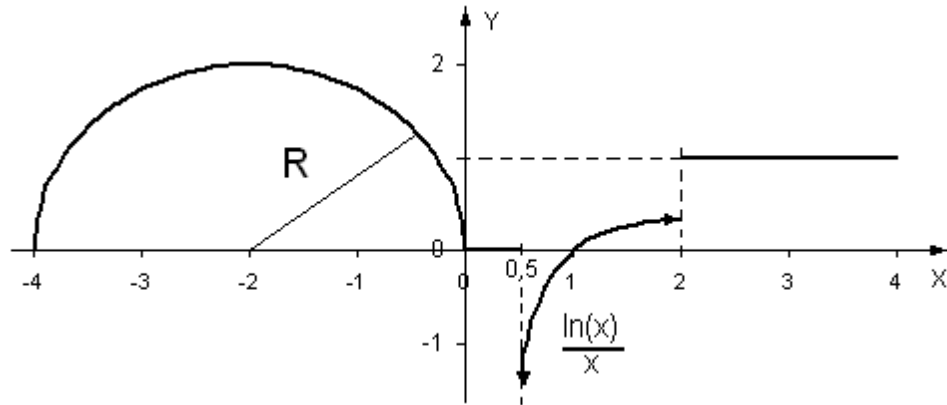
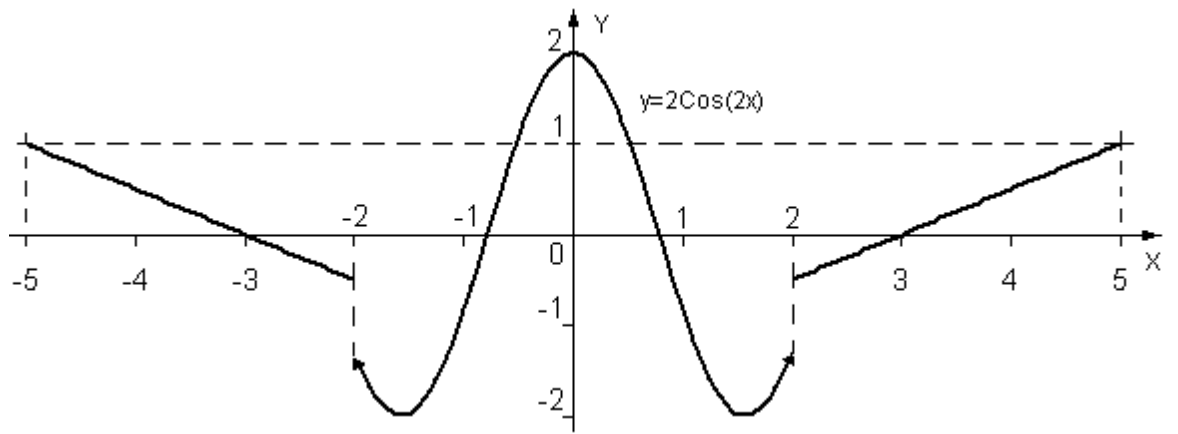
№	$f(x)$	№	$f(x)$	№	$f(x)$
1	$\begin{cases} \frac{1+x^2}{\sqrt{1+x^4}}, x \leq 0 \\ 2x + \frac{\sin^2 x}{2+x}, x > 0 \end{cases}$	11	$\begin{cases} \sqrt{1+x^2} - \cos^2 x, x \leq 0 \\ \frac{x}{\sqrt[3]{e^{x+1}}}, x > 0 \end{cases}$	21	$\begin{cases} \sqrt{1+x^2}, x \leq 0 \\ \frac{1+x}{1+\sqrt[3]{1+e^{0.2x}}}, x > 0 \end{cases}$
2	$\begin{cases} 3\sin x - \cos^2 x, x \leq 0 \\ 3\sqrt{1+x^2}, x > 0 \end{cases}$	12	$\begin{cases} 3\sin^2 x - \cos x, x \leq 0 \\ \sqrt{2+x^2}, x > 0 \end{cases}$	22	$\begin{cases} \sin x - 2\cos x, x \leq 0 \\ \sqrt{1+x^2}, x > 0 \end{cases}$
3	$\begin{cases} \frac{1+ x }{\sqrt[3]{1+x+x^2}}, x \leq -1 \\ 2\ln(1+x^2), x \in (-1;0) \\ (1+x)^{\frac{3}{5}}, x \geq 0 \end{cases}$	13	$\begin{cases} \frac{1+x}{1+x^2}, x < 0 \\ \sqrt{1+\frac{x}{1+x}}, x \in (0;1) \\ 2 \sin(3x) , x \geq 1 \end{cases}$	23	$\begin{cases} \frac{1+x}{\sqrt{1+x^2}}, x \leq 0 \\ -x + 2e^{-2x}, x \in (0;1) \\  2-x ^{1/3}, x \geq 1 \end{cases}$

4	$\begin{cases} \frac{3x^2}{1+x^2}, x \leq 0 \\ \sqrt{1+\frac{2x}{1+x^2}}, x > 0 \end{cases}$	14	$\begin{cases} 3x + \sqrt{1+x^2}, x < 0 \\ 2 \cos x \cdot e^{-2x}, x \in [0;1] \\ 2 \sin 3x, x > 1 \end{cases}$	24	$\begin{cases} \sqrt{1+ x }, x \leq 0 \\ \frac{1+3x}{2+\sqrt[3]{1+x}}, x > 0 \end{cases}$
5	$\begin{cases} \frac{3+\sin x}{1+x^2}, x \leq 0 \\ 2x^2 \cos^2 x, x > 0 \end{cases}$	15	$\begin{cases} \sqrt[3]{1+x^2}, x \leq 0 \\ \frac{1+x}{1+\cos^2 x}, x > 0 \end{cases}$	25	$\begin{cases} \frac{1+\cos x}{1+x^2}, x \leq 0 \\ x \cos x, x > 0 \end{cases}$
6	$\begin{cases} \frac{3+\sin^2 2x}{1+\cos^2 x}, x \leq 0 \\ 2\sqrt{1+2x}, x > 0 \end{cases}$	16	$\begin{cases} \frac{\sqrt{1+ x }}{2+ x }, x \leq 0 \\ \frac{1+x}{2+\cos^3 x}, x > 0 \end{cases}$	26	$\begin{cases} x + \sqrt{1+x^2}, x < 0 \\ \sin x \cdot e^{-x}, x \in [0;1] \\ 2 \cos^2 x, x > 1 \end{cases}$
7	$\begin{cases} \frac{ x }{1+x^2} e^{-2x}, x < 0 \\ \sqrt{1+x^2}, x \geq 0 \end{cases}$	17	$\begin{cases} \frac{ x }{1+x^2}, x < 0 \\ \sqrt{1+x}, x \geq 0 \end{cases}$	27	$\begin{cases}  x e^{-2x}, x < 0 \\ \frac{1}{\sqrt{1+x^2}}, x \geq 0 \end{cases}$
8	$\begin{cases} \sqrt{1+2x^2}, x \leq 0 \\ \frac{1+x}{1+\sqrt[3]{1+e^x}}, x > 0 \end{cases}$	18	$\begin{cases} 2\sqrt{1+x^2}, x \leq 0 \\ \frac{1}{1+\sqrt[3]{e^x}}, x > 0 \end{cases}$	28	$\begin{cases} 3\sqrt{1+x^2}, x < 0 \\ \sin 2x \cdot 3e^{-x}, x \in [0;1] \\ \cos x \sin x, x > 1 \end{cases}$
9	$\begin{cases} \frac{1}{ x ^3}, x < 0 \\ -2x + \sqrt{x}, x \in [0;1) \\  x-3 ^{0.1}, x \geq 1 \end{cases}$	19	$\begin{cases} \sqrt{1+x^2}, x < 0 \\ 2 \cos^2 x, x \in [0;1] \\ \sqrt{1+\sqrt[3]{2 \sin 3x}}, x > 1 \end{cases}$	29	$\begin{cases} 1+x+x^2, x < 0 \\ \sqrt{1+2x}, x \in [0;1) \\ 2 0.5+\sin x , x \geq 1 \end{cases}$
10	$\begin{cases} \sqrt{1+2x^2-\sin^2 x}, x \leq 0 \\ \frac{2+x}{\sqrt[3]{2+e^{-0.1x}}}, x > 0 \end{cases}$	20	$\begin{cases} \sqrt{1+x-\sin x}, x \leq 0 \\ \frac{x}{\sqrt[4]{e^{-0.1x}}}, x > 0 \end{cases}$	30	$\begin{cases} \frac{1+\sin x}{1+2\cos x}, x \leq 0 \\ \sqrt{1+x}, x > 0 \end{cases}$

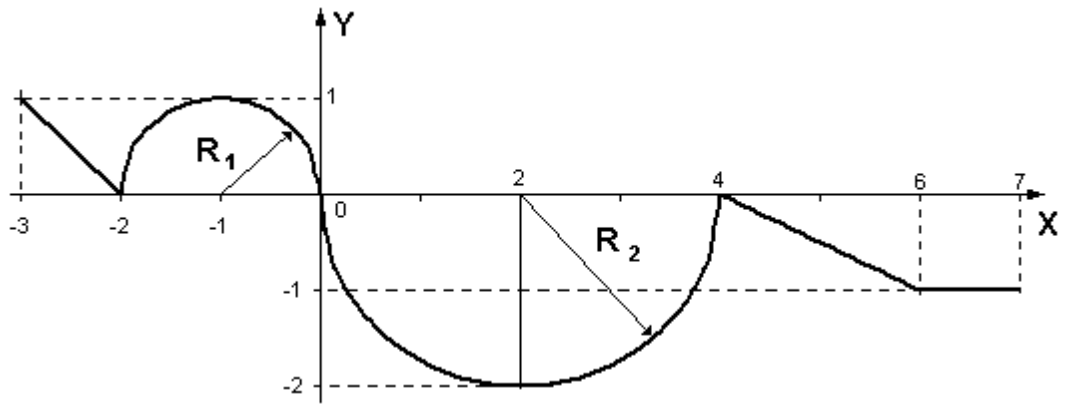
**Завдання 2.2.** Напишіть програму, яка за введеним значенням аргументу обраховує значення функції, яку задано у вигляді графіку. Параметри необхідні для рішення завдання слід отримати із графіка та визначити у програмі.



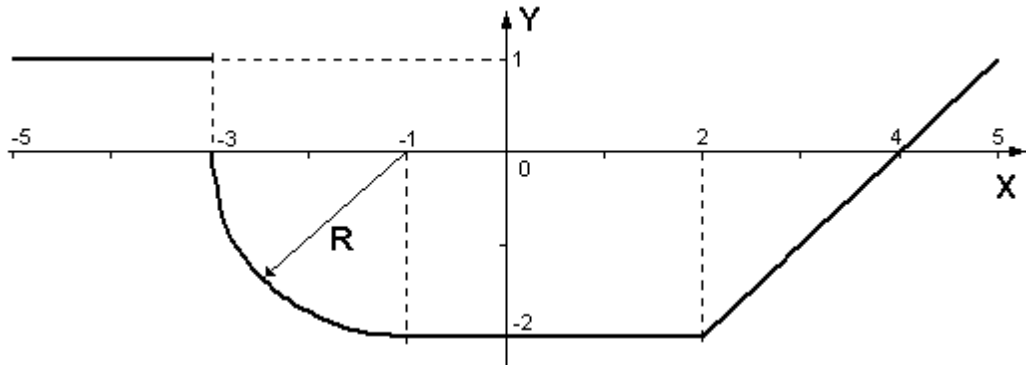




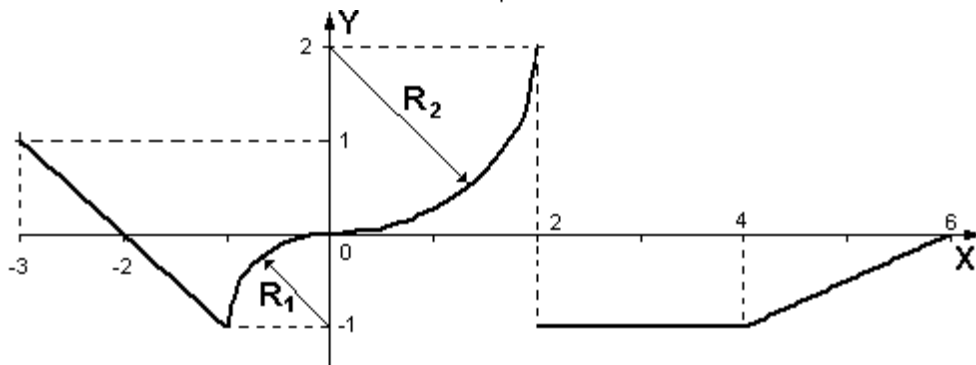
12.



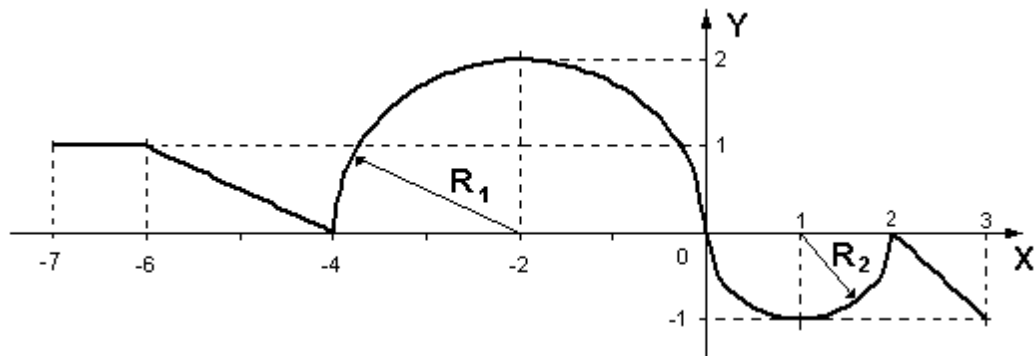
13.

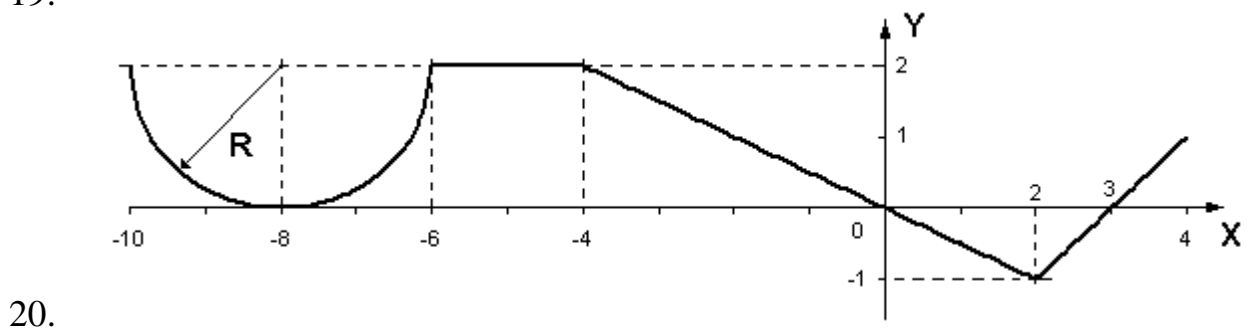
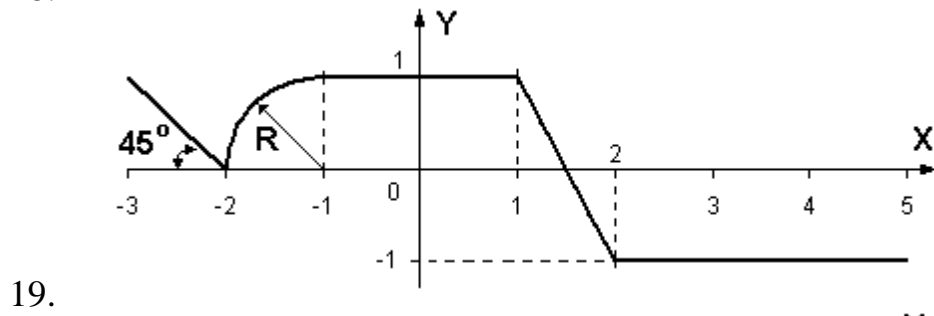
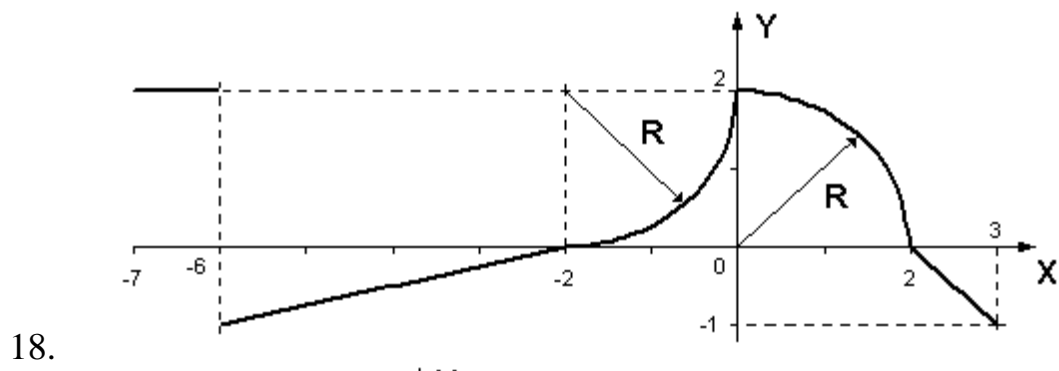
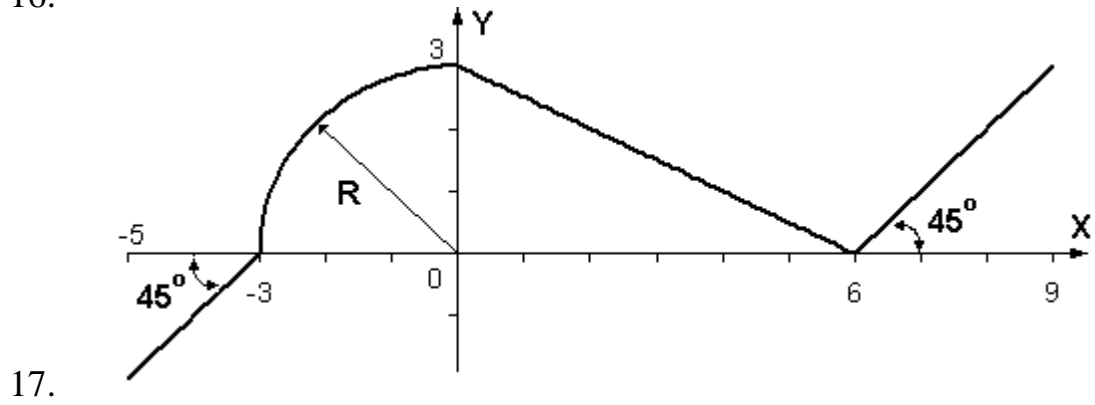
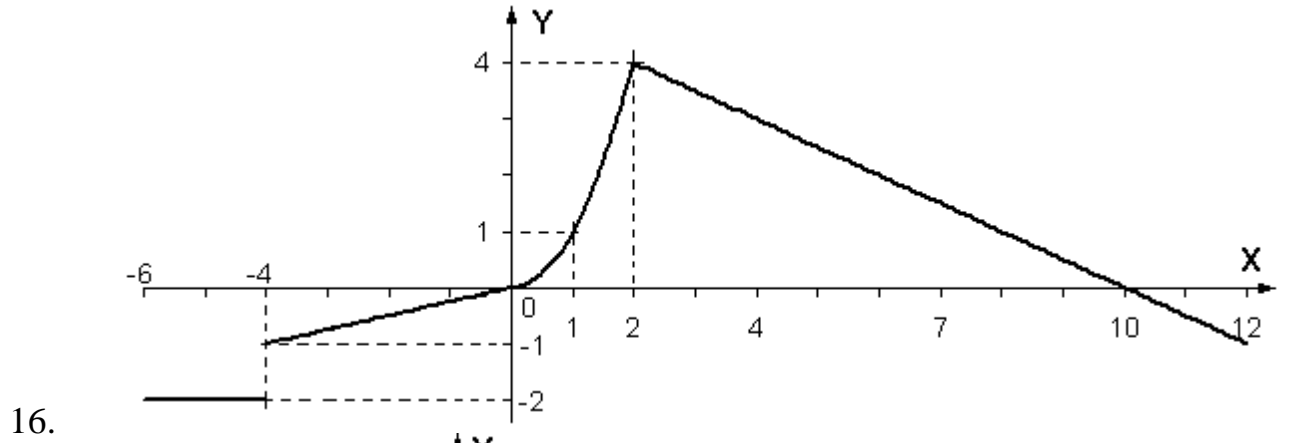


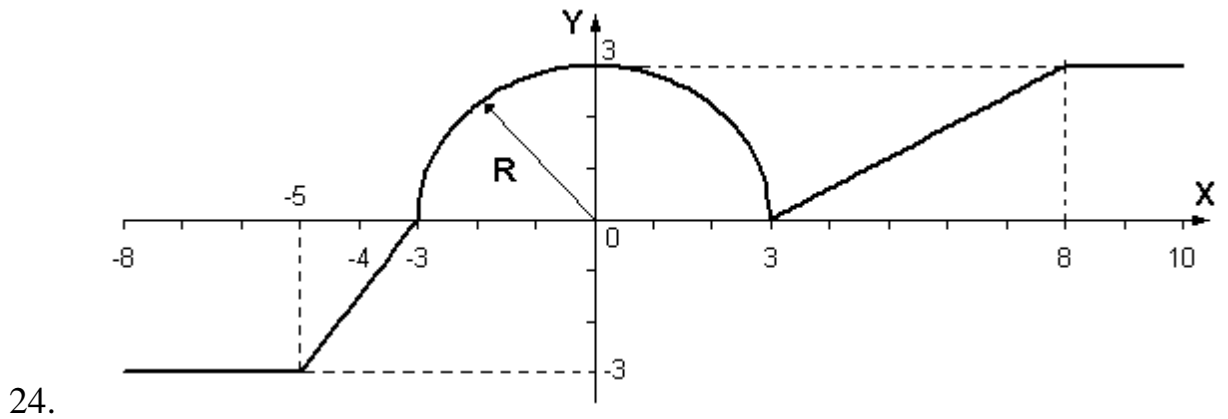
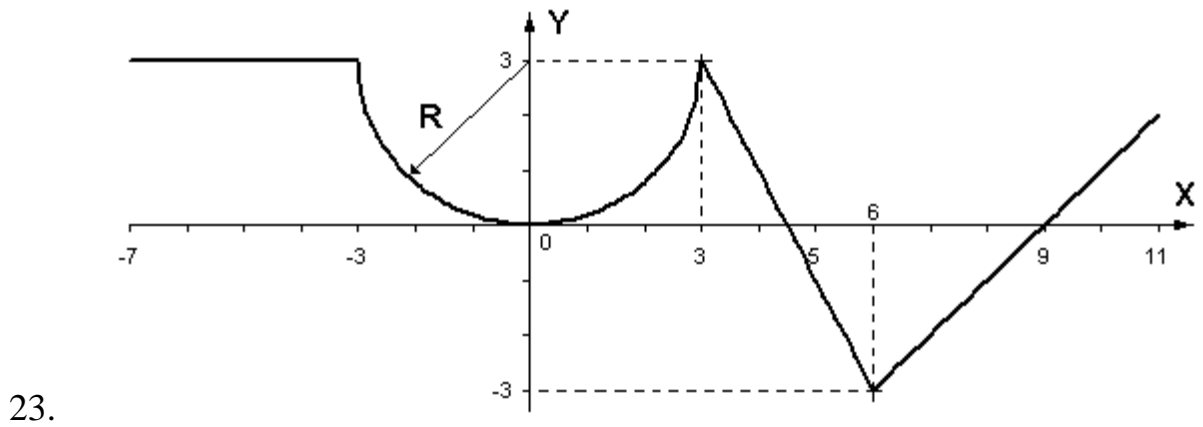
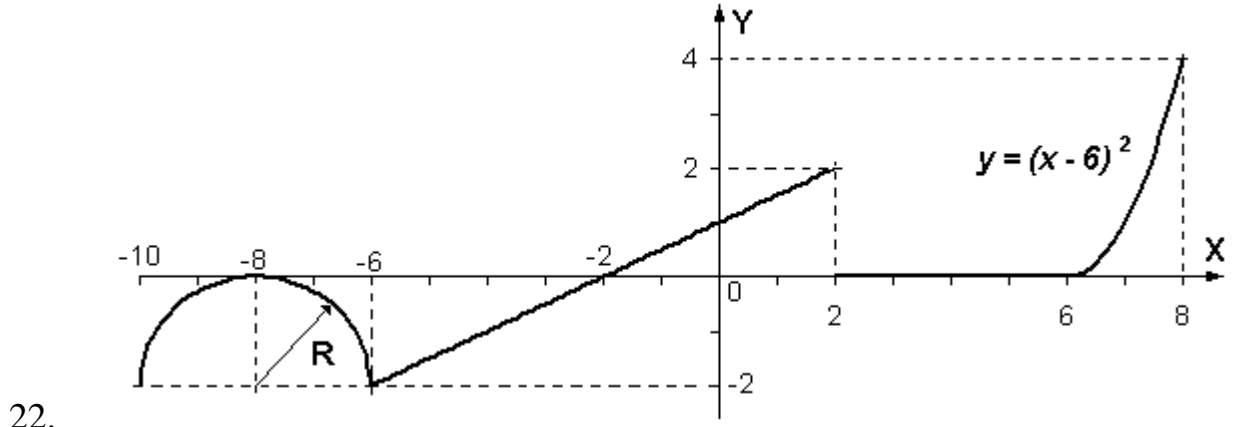
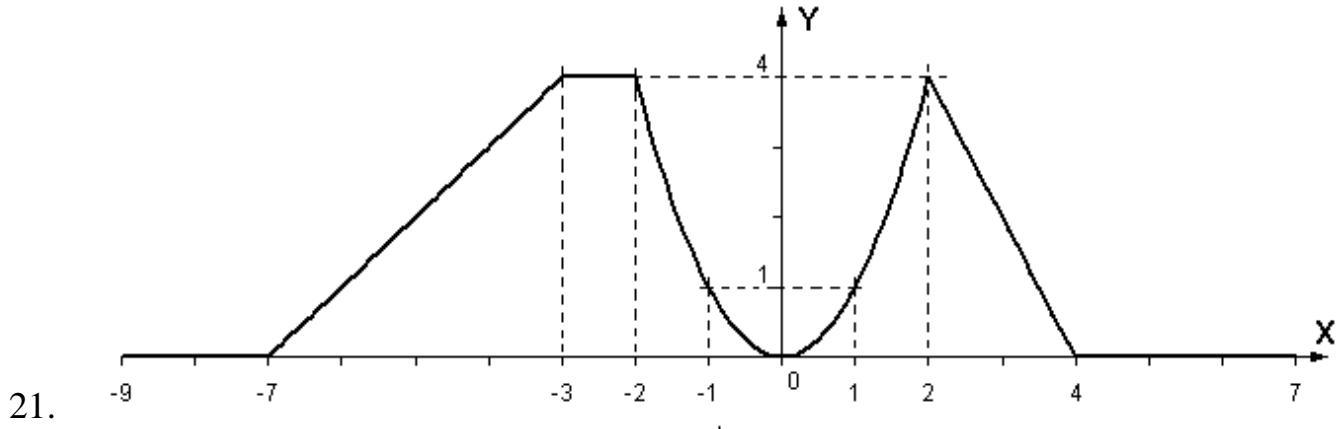
14.



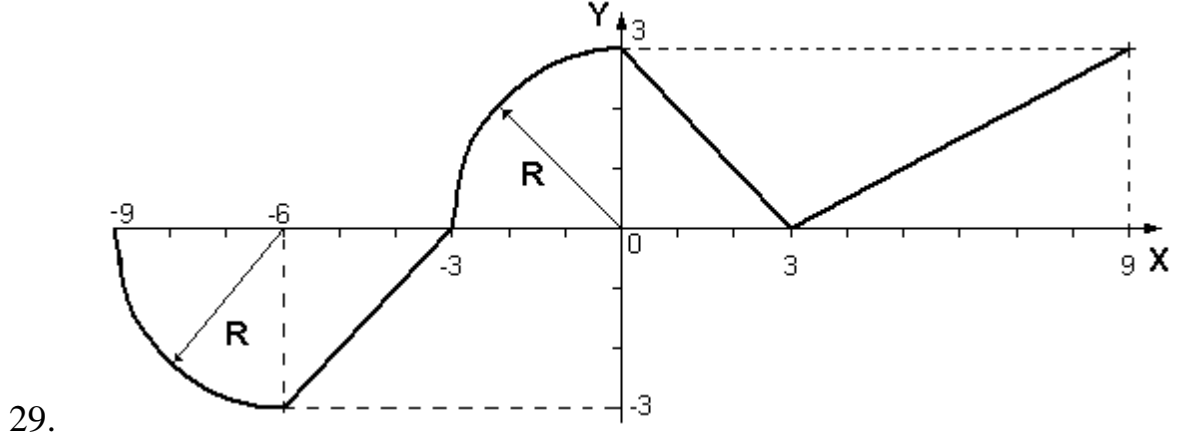
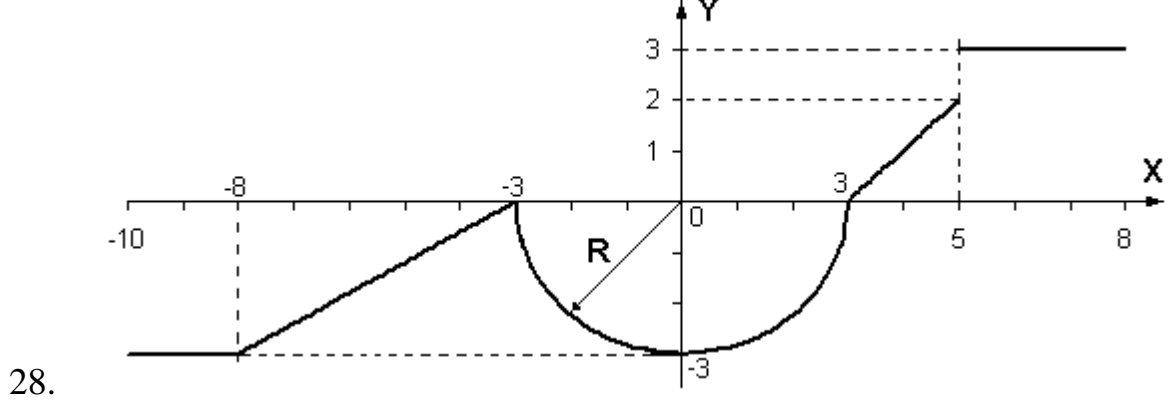
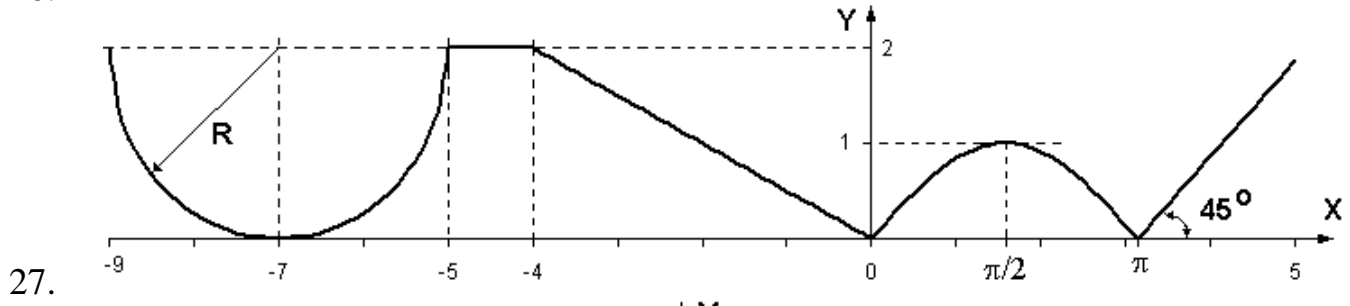
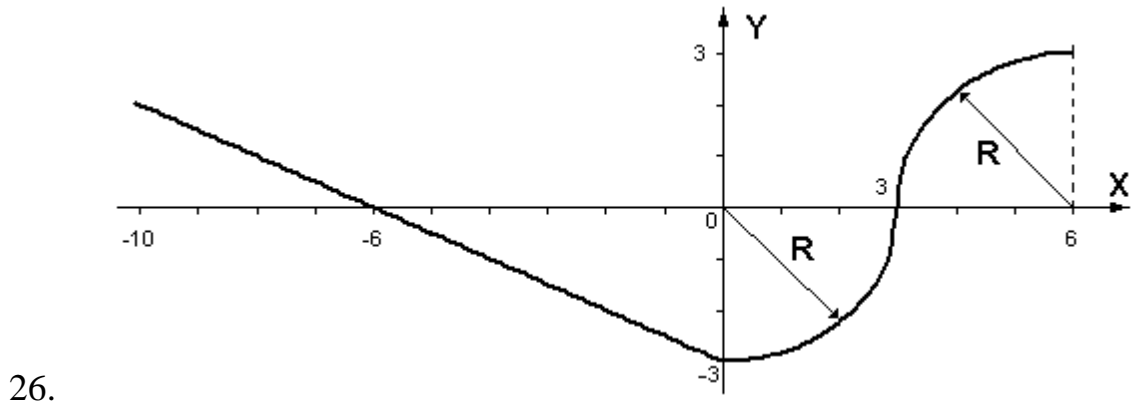
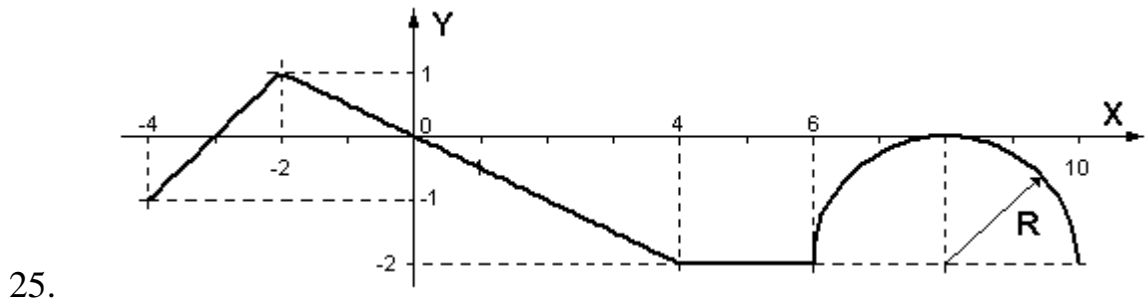
15.





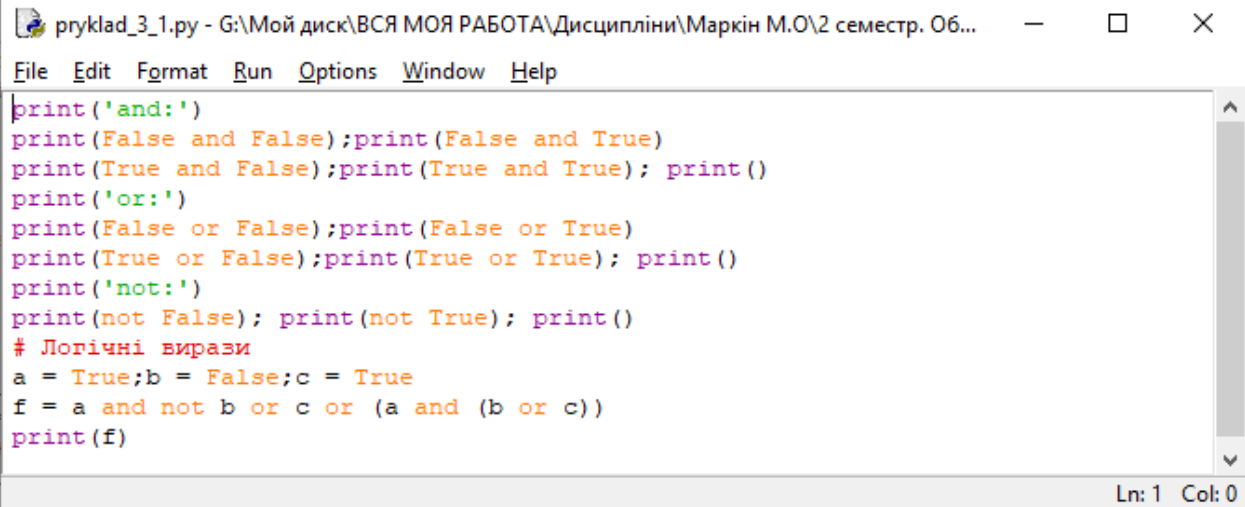






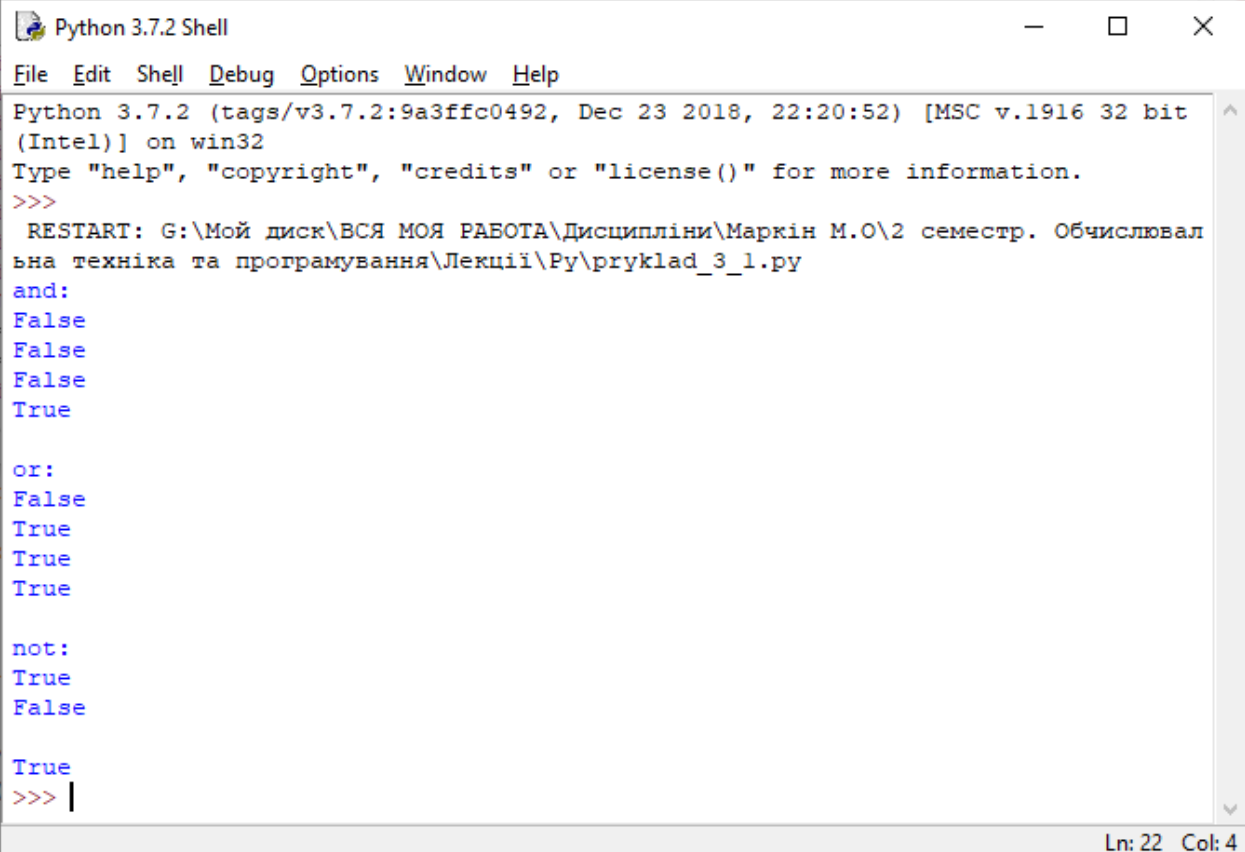
## Приклади

### Приклад 2.1. Приклади логічних операцій та виразів



```
pryklad_3_1.py - G:\Мой диск\ВСЯ МОЯ РАБОТА\Дисципліни\Маркін М.О\2 семестр. Об...
File Edit Format Run Options Window Help
print('and:')
print(False and False);print(False and True)
print(True and False);print(True and True); print()
print('or:')
print(False or False);print(False or True)
print(True or False);print(True or True); print()
print('not:')
print(not False); print(not True); print()
# Логічні вирази
a = True;b = False;c = True
f = a and not b or c or (a and (b or c))
print(f)
Ln: 1 Col: 0
```

### Виконання коду:



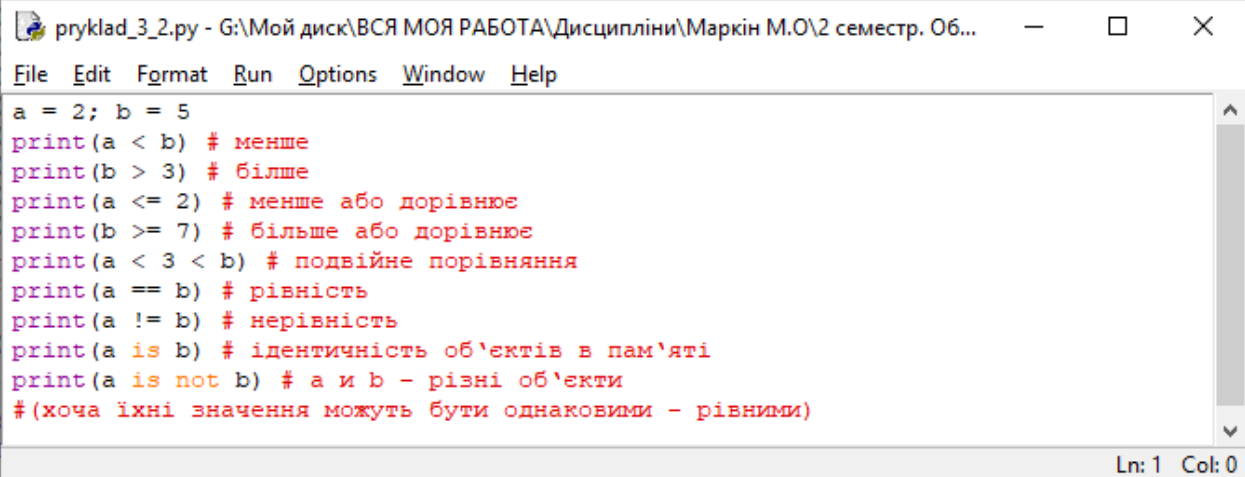
```
Python 3.7.2 Shell
File Edit Shell Debug Options Window Help
Python 3.7.2 (tags/v3.7.2:9a3ffc0492, Dec 23 2018, 22:20:52) [MSC v.1916 32 bit
(Intel)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
RESTART: G:\Мой диск\ВСЯ МОЯ РАБОТА\Дисципліни\Маркін М.О\2 семестр. Обчислювал
ьна техніка та програмування\Лекції\Py\pryklad_3_1.py
and:
False
False
False
True

or:
False
True
True
True

not:
True
False

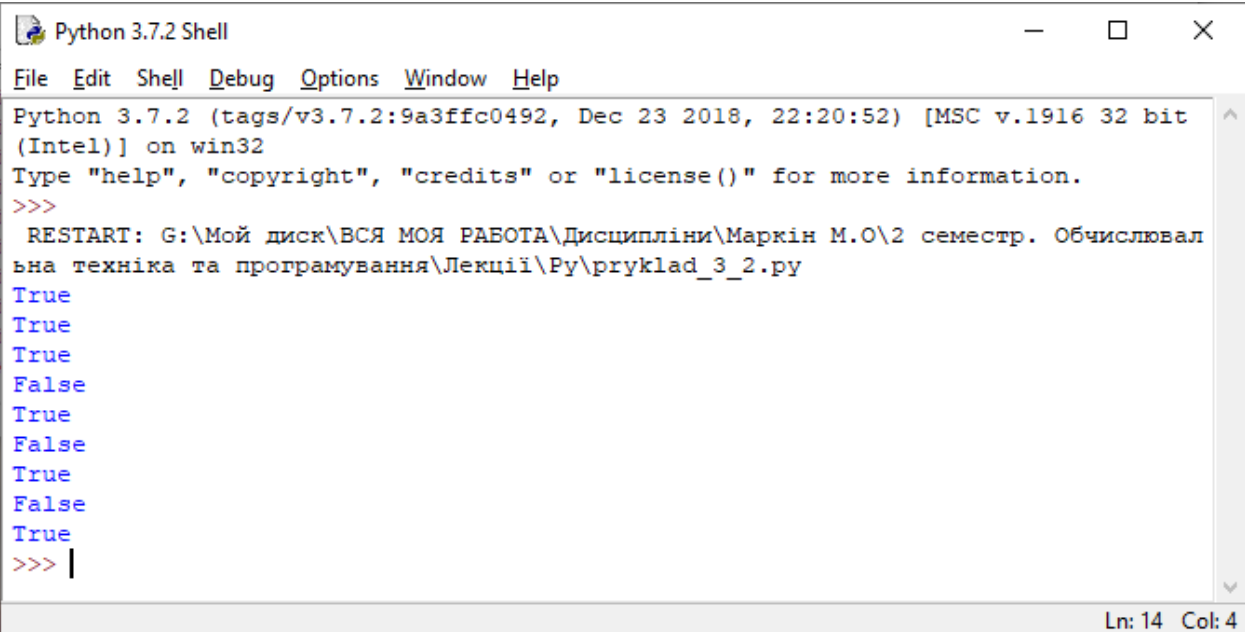
True
>>> |
Ln: 22 Col: 4
```

## Приклад 2.2. Приклади порівнянь



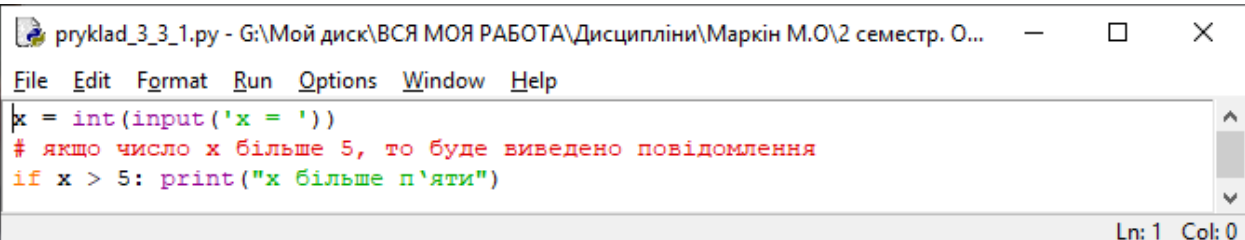
```
pryklad_3_2.py - G:\Мой диск\ВСЯ МОЯ РАБОТА\Дисципліни\Маркін М.О\2 семестр. Об...
File Edit Format Run Options Window Help
a = 2; b = 5
print(a < b) # менше
print(b > 3) # більше
print(a <= 2) # менше або дорівнює
print(b >= 7) # більше або дорівнює
print(a < 3 < b) # подвійне порівняння
print(a == b) # рівність
print(a != b) # нерівність
print(a is b) # ідентичність об'єктів в пам'яті
print(a is not b) # a и b - різні об'єкти
#(хоча їхні значення можуть бути однаковими - рівними)
Ln: 1 Col: 0
```

Виконання коду:



```
Python 3.7.2 Shell
File Edit Shell Debug Options Window Help
Python 3.7.2 (tags/v3.7.2:9a3ffc0492, Dec 23 2018, 22:20:52) [MSC v.1916 32 bit
(Intel)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
RESTART: G:\Мой диск\ВСЯ МОЯ РАБОТА\Дисципліни\Маркін М.О\2 семестр. Обчислювал
ьна техніка та програмування\Лекції\Py\pryklad_3_2.py
True
True
True
False
True
False
True
False
True
True
>>> |
Ln: 14 Col: 4
```

## Приклад 2.3.



```
pryklad_3_3_1.py - G:\Мой диск\ВСЯ МОЯ РАБОТА\Дисципліни\Маркін М.О\2 семестр. О...
File Edit Format Run Options Window Help
x = int(input('x = '))
# якщо число x більше 5, то буде виведено повідомлення
if x > 5: print("x більше п'яти")
Ln: 1 Col: 0
```

## Виконання коду:

```

Python 3.7.2 Shell
File Edit Shell Debug Options Window Help
Python 3.7.2 (tags/v3.7.2:9a3ffc0492, Dec 23 2018, 22:20:52) [MSC v.1916 32 bit
(Intel)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
RESTART: G:\Мой диск\ВСЯ МОЯ РАБОТА\Дисципліни\Маркін М.О\2 семестр. Обчислювал
ьна техніка та програмування\Лекції\Py\pryklad_3_3_1.py
x = 4
>>>
RESTART: G:\Мой диск\ВСЯ МОЯ РАБОТА\Дисципліни\Маркін М.О\2 семестр. Обчислювал
ьна техніка та програмування\Лекції\Py\pryklad_3_3_1.py
x = 7
x більше п'яти
>>> |
Ln: 10 Col: 4

```

```

pryklad_3_3_2.py - G:\Мой диск\ВСЯ МОЯ РАБОТА\Дисципліни\Маркін М.О\2 семестр. О...
File Edit Format Run Options Window Help
value = None
# Синтаксис мови вимагає наявність в блоці хоча б одного
# оператору, тому існує спеціальний оператор pass,
# який нічого не виконує, його можна використовувати як заглушка
if value is not None: pass # TODO: add logic
Ln: 1 Col: 0

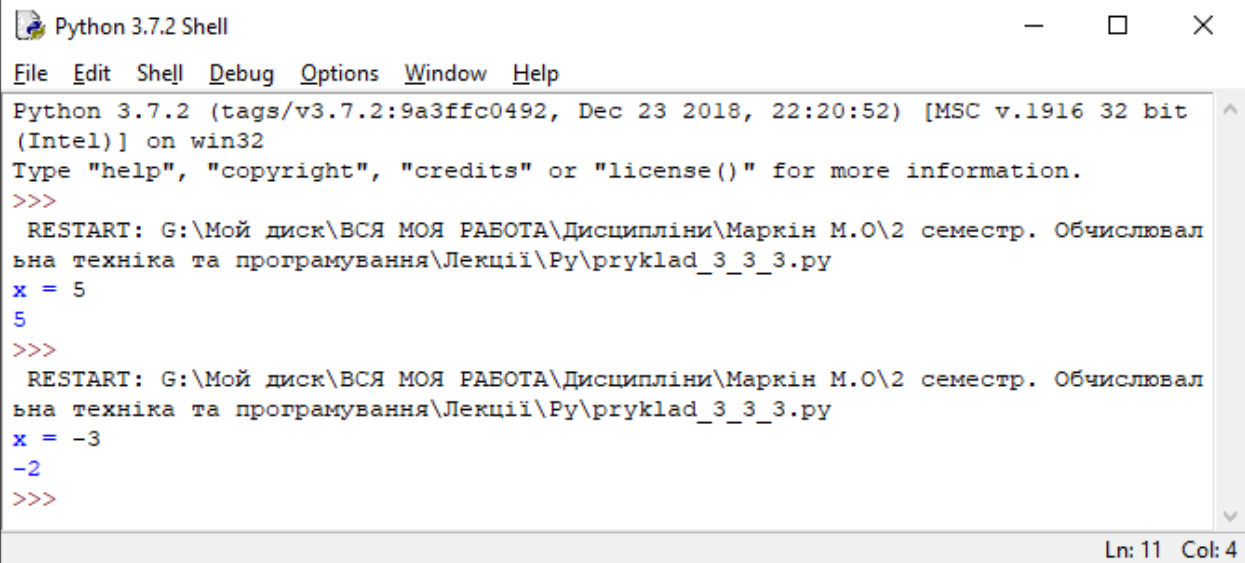
```

```

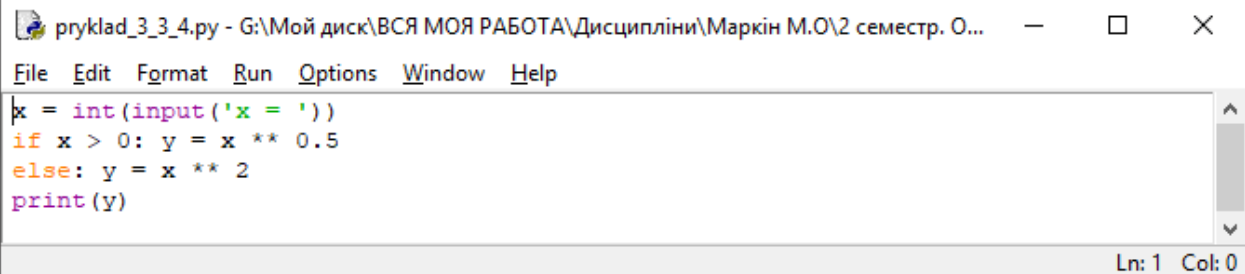
pryklad_3_3_3.py - G:\Мой диск\ВСЯ МОЯ РАБОТА\Дисципліни\Маркін М.О\2 семестр. О...
File Edit Format Run Options Window Help
x = int(input('x = '))
if x < 0: x += 1
print(x)
Ln: 1 Col: 0

```

## Виконання коду:

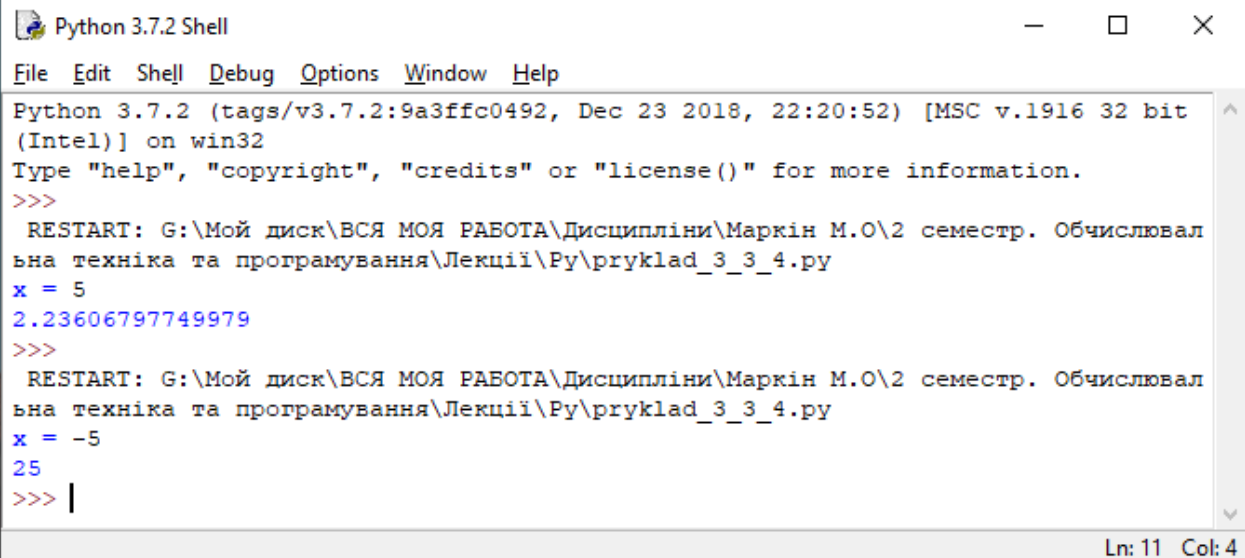


```
Python 3.7.2 Shell
File Edit Shell Debug Options Window Help
Python 3.7.2 (tags/v3.7.2:9a3ffc0492, Dec 23 2018, 22:20:52) [MSC v.1916 32 bit
(Intel)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
RESTART: G:\Мой диск\ВСЯ МОЯ РАБОТА\Дисципліни\Маркін М.О\2 семестр. Обчислювал
ьна техніка та програмування\Лекції\Py\pruklad_3_3_3.py
x = 5
5
>>>
RESTART: G:\Мой диск\ВСЯ МОЯ РАБОТА\Дисципліни\Маркін М.О\2 семестр. Обчислювал
ьна техніка та програмування\Лекції\Py\pruklad_3_3_3.py
x = -3
-2
>>>
Ln: 11 Col: 4
```



```
pruklad_3_3_4.py - G:\Мой диск\ВСЯ МОЯ РАБОТА\Дисципліни\Маркін М.О\2 семестр. О...
File Edit Format Run Options Window Help
x = int(input('x = '))
if x > 0: y = x ** 0.5
else: y = x ** 2
print(y)
Ln: 1 Col: 0
```

## Виконання коду:



```
Python 3.7.2 Shell
File Edit Shell Debug Options Window Help
Python 3.7.2 (tags/v3.7.2:9a3ffc0492, Dec 23 2018, 22:20:52) [MSC v.1916 32 bit
(Intel)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
RESTART: G:\Мой диск\ВСЯ МОЯ РАБОТА\Дисципліни\Маркін М.О\2 семестр. Обчислювал
ьна техніка та програмування\Лекції\Py\pruklad_3_3_4.py
x = 5
2.23606797749979
>>>
RESTART: G:\Мой диск\ВСЯ МОЯ РАБОТА\Дисципліни\Маркін М.О\2 семестр. Обчислювал
ьна техніка та програмування\Лекції\Py\pruklad_3_3_4.py
x = -5
25
>>> |
Ln: 11 Col: 4
```

**Приклад 2.4.** Визначити середнє арифметичне заданої непустиї послідовності додатних цілих чисел, за якою слідує «0» (це ознака кінця послідовності).

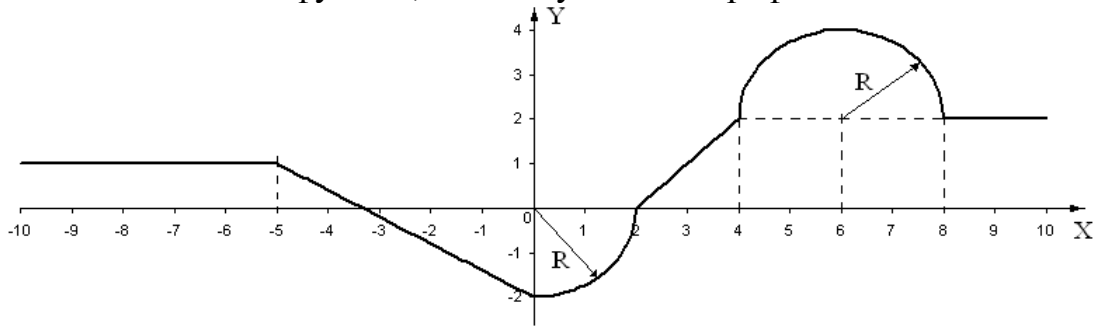
```
pryklad_3_4.py - G:\Мой диск\ВСЯ МОЯ РАБОТА\Дисципліни\Маркін М.О\2 семестр. Об...
File Edit Format Run Options Window Help
from math import *
a=input('Input first number: ')
if not a.isdigit():
    print("String value can not be entered")
    print("or number is negative, restart the program!")
    input("reload the program!")
else:
    a=int(a)
if a==0:
    print("The number can not be zero")
    input()
count=0
ar=0
ar+=a
count+=1
if a!=0:
    try:
        a=int(input('Input next number or Enter 0 to finish: '))
    except:
        print("String value can not be entered")
        print("or number is negative, restart the program!")
        input("reload the program!")
    else:
        ar+=a
        count+=1
ar=ar/count
print("Average: ",ar)
```

Результат виконання:

```
Python 3.7.2 Shell
File Edit Shell Debug Options Window Help
Python 3.7.2 (tags/v3.7.2:9a3ffc0492, Dec 23 2018, 22:20:52) [MSC v.1916 32 bit
(Intel)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
RESTART: G:\Мой диск\ВСЯ МОЯ РАБОТА\Дисципліни\Маркін М.О\2 семестр. Обчислювал
ьна техніка та програмування\Лекції\Py\pryklad_3_4.py
Input first number: 6
Input next number or Enter 0 to finish: 7
Average: 6.5
>>> |
Ln: 8 Col: 4
```

### Приклад 2.5

Написати програму, яка по введеному значенню аргументу обчислює значення функції, заданої у вигляді графіка.



Для розв'язання задачі використано оператор розгалуження, який в мові Python має наступний вигляд:

```

if <Логічний вираз>:
    <Блок - виконується, якщо умова істинна>
elif <Логічний вираз>:
    <Блок - виконується, якщо умова істинна>
]
else:
    <Блок - виконується, якщо всі умови помилкові>
]

```

<Блок> - це набір вкладених інструкцій, які виділяються однаковою кількістю пробілів (зазвичай чотирма).

Для введення даних використовується інструкція `input()`, яка повертає рядок. Введені значення, перед використанням в арифметичних виразах, повинні бути перетворені до числовий формат.

Вивід даних виконується інструкцією `print()`, в якій використано форматований вивід даних.

Графік функції представлений фрагментами прямих ліній, що описуються рівнянням  $y = kx + b$  і дугами кіл. У загальному випадку рівняння кола може бути представлено так:  $(x - a)^2 + (y - b)^2 = R^2$ .

Невідомі параметри, кут нахилу і зміщення прямої, а так само координати центру дуг, визначимо, використовуючи дані з графіка.

Для прямої на інтервалі  $(-5, 0)$  можемо записати наступну систему рівнянь:

$$\begin{cases} 1 = l \cdot (-5) + b \\ -2 = k \cdot 0 + b \end{cases}$$

З другого рівняння випливає, що  $b = -2$ , а з першого -  $k = -3/5$ .

Для півкола з центром  $(6, 2)$  рівняння кола набуде вигляду:

$$(x - 6)^2 + (y - 2)^2 = 2^2$$

Перепишемо рівняння так:

$$(y - 2)^2 = 2^2 - (x - 6)^2$$

$$(y-2)^2 = 4 - (x-6)^2, \text{ звідки } y = 2 + \sqrt{4 - (x-6)^2}.$$

Знак перед коренем обраний для випадку, коли розглядається верхня частина півкола.

Виконавши необхідні обчислення для всіх фрагментів функції, ми отримаємо систему рівнянь, яку запишемо в наступному вигляді:

$$y = \begin{cases} 1 & x < -5 \\ -\frac{3}{5}x - 2 & -5 \leq x < 0 \\ -\sqrt{4+x^2} & 0 \leq x < 2 \\ x - 2 & 2 \leq x < 4 \\ 2 + \sqrt{4 - (x-6)^2} & 4 \leq x < 8 \\ 2 & x \geq 8 \end{cases}$$

Функція визначена на всьому діапазоні  $x \in (-\infty; +\infty)$ . При цьому, особливих точок у неї немає.

### Опис алгоритму

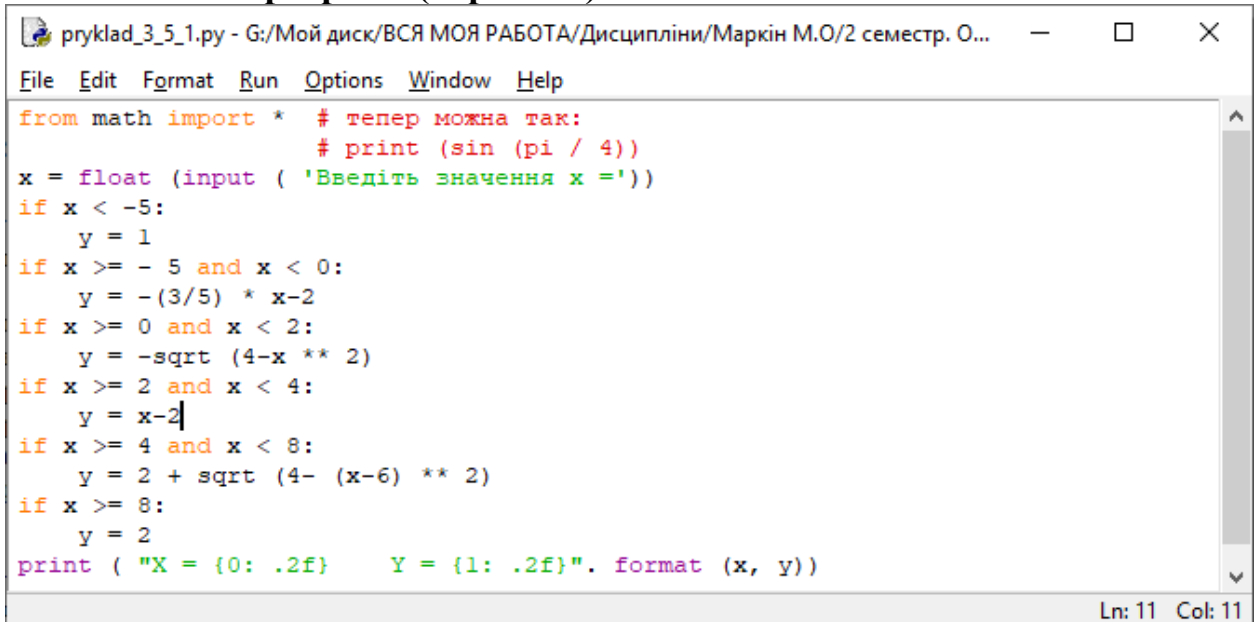
1. Ввести значення аргументу  $x$  і перетворити його до типу *float*.
2. Визначити, до якого інтервалу з області визначення функції воно належить, і обчислити значення функції  $y$  за відповідною формулою.
3. Вивести значення  $x$  і  $y$ .

### Опис вхідних та вихідних даних

Вхідні дані надходять з клавіатури, а вихідні - виводяться на монітор для перегляду. Вхідні і вихідні дані мають тип *float*.



### Лістинг програми (варіант 1)



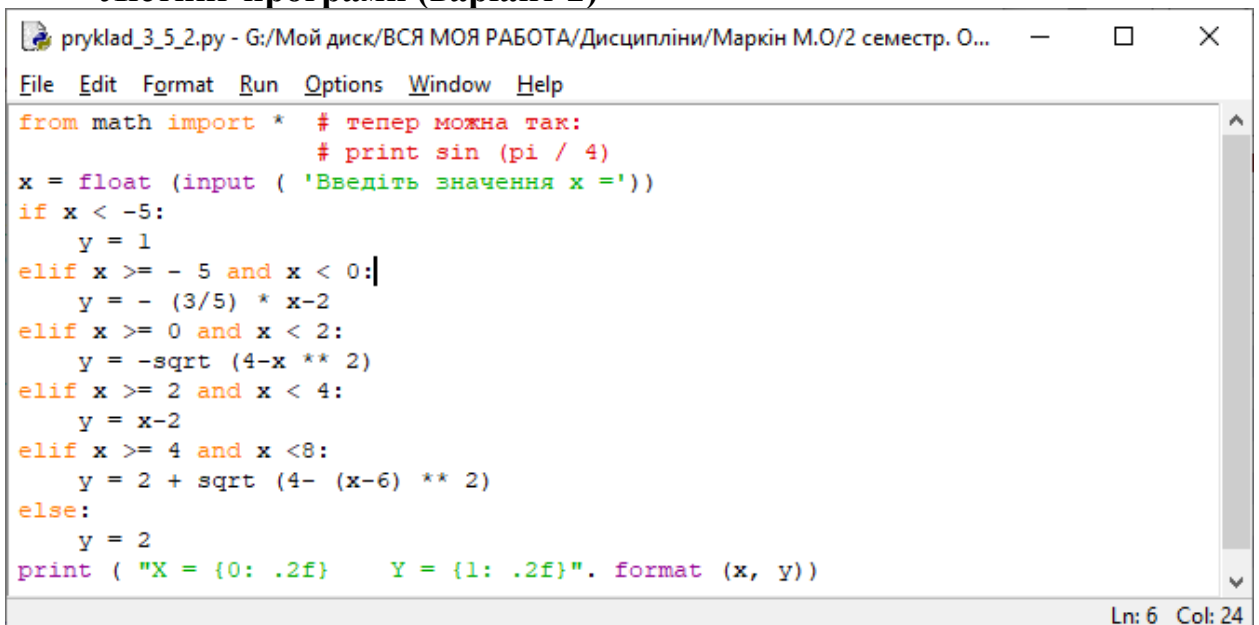
```

pryklad_3_5_1.py - G:/Мой диск/ВСЯ МОЯ РАБОТА/Дисципліни/Маркін М.О/2 семестр. О...
File Edit Format Run Options Window Help
from math import * # тепер можна так:
                    # print (sin (pi / 4))
x = float (input ( 'Введіть значення x =' ))
if x < -5:
    y = 1
if x >= - 5 and x < 0:
    y = - (3/5) * x-2
if x >= 0 and x < 2:
    y = -sqrt (4-x ** 2)
if x >= 2 and x < 4:
    y = x-2
if x >= 4 and x < 8:
    y = 2 + sqrt (4- (x-6) ** 2)
if x >= 8:
    y = 2
print ( "X = {0: .2f}    Y = {1: .2f}". format (x, y))
Ln: 11 Col: 11

```

Слід зазначити, що в такому записі алгоритму перевірка виконується для всіх умовних операторів, в тому числі і тих, які слідують за обчисленим. Так, наприклад, якщо  $x$  ріне  $-3$ , то виконається другий оператор, але і у всіх наступних операторах операція порівняння буде проведена. Число перевірок можна скоротити, якщо написати програму з використанням вкладених умовних операторів.

### Лістинг програми (варіант 2)



```

pryklad_3_5_2.py - G:/Мой диск/ВСЯ МОЯ РАБОТА/Дисципліни/Маркін М.О/2 семестр. О...
File Edit Format Run Options Window Help
from math import * # тепер можна так:
                    # print sin (pi / 4)
x = float (input ( 'Введіть значення x =' ))
if x < -5:
    y = 1
elif x >= - 5 and x < 0:
    y = - (3/5) * x-2
elif x >= 0 and x < 2:
    y = -sqrt (4-x ** 2)
elif x >= 2 and x < 4:
    y = x-2
elif x >= 4 and x < 8:
    y = 2 + sqrt (4- (x-6) ** 2)
else:
    y = 2
print ( "X = {0: .2f}    Y = {1: .2f}". format (x, y))
Ln: 6 Col: 24

```

## Результат роботи програми

```
Python 3.7.2 Shell
File Edit Shell Debug Options Window Help
Python 3.7.2 (tags/v3.7.2:9a3ffc0492, Dec 23 2018, 22:20:52) [MSC v.1916 32 bit
(Intel)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
RESTART: G:/Мой диск/ВСЯ МОЯ РАБОТА/Дисципліни/Маркін М.О/2 семестр. Обчислювал
ьна техніка та програмування/Лекції/Py/pruklad_3_5_1.py
Введіть значення x =-6
X = -6.00    Y = 1.00
>>>
RESTART: G:/Мой диск/ВСЯ МОЯ РАБОТА/Дисципліни/Маркін М.О/2 семестр. Обчислювал
ьна техніка та програмування/Лекції/Py/pruklad_3_5_1.py
Введіть значення x =-3.33
X = -3.33    Y = -0.00
>>>
RESTART: G:/Мой диск/ВСЯ МОЯ РАБОТА/Дисципліни/Маркін М.О/2 семестр. Обчислювал
ьна техніка та програмування/Лекції/Py/pruklad_3_5_1.py
Введіть значення x =6
X = 6.00    Y = 4.00
>>>
RESTART: G:/Мой диск/ВСЯ МОЯ РАБОТА/Дисципліни/Маркін М.О/2 семестр. Обчислювал
ьна техніка та програмування/Лекції/Py/pruklad_3_5_2.py
Введіть значення x =-6
X = -6.00    Y = 1.00
>>>
RESTART: G:/Мой диск/ВСЯ МОЯ РАБОТА/Дисципліни/Маркін М.О/2 семестр. Обчислювал
ьна техніка та програмування/Лекції/Py/pruklad_3_5_2.py
Введіть значення x =-3.33
X = -3.33    Y = -0.00
>>>
RESTART: G:/Мой диск/ВСЯ МОЯ РАБОТА/Дисципліни/Маркін М.О/2 семестр. Обчислювал
ьна техніка та програмування/Лекції/Py/pruklad_3_5_2.py
Введіть значення x =6
X = 6.00    Y = 4.00
>>>
```

### ПЕРЕЛІК ПОСИЛАНЬ

1. **ОСНОВИ ПРОГРАМУВАННЯ:** методичні вказівки до виконання комп'ютерних практикумів на PYTHON з навчальної дисципліни «Основи програмування» для студентів спеціальності 122 «Комп'ютерні науки» зі спеціалізації «Інформаційні технології в біології та медицині» / Уклад. Л.М. Добровська. – К.: НТУУ «КПІ ім. Ігоря Сікорського», 2017. – 254 с.
2. *Федоров, Д. Ю. Программирование на языке высокого уровня Python* : учебное пособие для прикладного бакалавриата / Д. Ю. Федоров. — М. : Издательство Юрайт, 2018. — 126 с. — (Серия : Бакалавр. Прикладной курс).