МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Львівський національний університет імені Івана Франка
Факультет прикладної математики та інформатики

ЗАГАЛЬНИЙ ЗВІТ

про виконання індивідуальних завдань
з дисципліни ”Навчальна практика”

студента І-го курсу
групи ПМП-12
[redacted]

Керівник роботи
асист. Борисюк Я.Є.

Львів – 2022

Звіт до завдання 1:

Завдання 1. Задано три прямі a, b, c. Вияснити чи можна побудувати з них прямокутник. Вивести повідомлення виду : «З прямих a, b, c прямокутник побудувати можна/ не можна». Якщо можна, то знайти периметр, якщо не можна - то вивести сторони у порядку спадання..

Алгоритм:

- Вводимо довжини трьох відрізків за допомогою функції input().
- За допомогою логічного оператора if(): перевіряємо чи є хоча б однин відрізок, який є більша за суму двох інших відрізків.
- Якщо є хоча б один відрізок, який є більша за суму двох інших відрізків, тоді з даних відрізків неможливо скласти трикутник.
- Якщо трикутник неможиво утворити, то за допомогою логічного оператора if() порівнюємо всі відрізки та виводимо їх у порядку спадання за допомогою функції print()
- Якшо всі відрізки є меншими за суму двох інших відрізків, тоді з даних відрізків можливо скласти трикутник, тоді ми виводимо суму сторін трикутника.

Код:

```
a = float(input("Введіть довжину сторони a "))
b = float(input("Введіть довжину сторони b "))
c = float(input("Введіть довжину сторони c "))
if a < b + c and b < a + c and c < a + b:
    print('З прямих a, b, c трикутник побудувати  можна ')
    print('Периметр трикутника = ' + str(a + b + c))
else:
    print('З прямих a, b, c трикутник побудувати  не можна')
    if a >= b and a >= c:
        print('a = ' + str(a))
        if b >= c:
            print('b = ' + str(b))
            print('c = ' + str(c))
        else:
            print('c = ' + str(c))
            print('b = ' + str(b))
    elif b >= c:
        print('b = ' + str(b))
        if a >= c:
            print('a = ' + str(a))
            print('c = ' + str(c))
        else:
            print('c = ' + str(c))
            print('a = ' + str(a))
    else:
        if a >= b:
            print('a = ' + str(a))
            print('b = ' + str(b))
        else:
            print('b = ' + str(b))
            print('a = ' + str(a))
a = print()
```

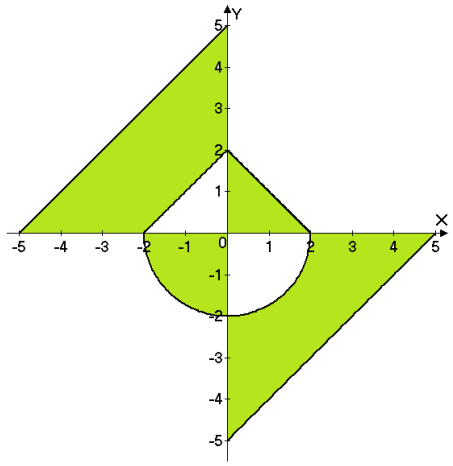
Приклад виконання:

Введіть довжину сторони a 3
Введіть довжину сторони b 4
Введіть довжину сторони c 5
З прямих a, b, c трикутник побудувати можна
Периметр трикутника = 12.0

Введіть довжину сторони a 13
Введіть довжину сторони b 4

Введіть довжину сторони c 5
З прямих a, b, c трикутник побудувати не можна
a = 13 c = 5 b = 4

Завдання 2 Задано точку P(x,y). Визначити чи належить точка заданій області (зеленого кольору)



Алгоритм:

- Вводимо значення змінних x і y за допомогою функції input()
- За допомогою логічного оператора if(): перевіряємо чи x та y дорівнюють 0 -точка належить області.
- Перевіряюмо у якій чверті знаходиться задана точка.
  Використовучи рівняння прямої y=kx+b та/ чи рівняння кола $(x-a)^2+ (y-b)^2$ перевіряємо чи задана точка входить в область визначення.
- Виводимо результат за допомогою функції print().

Код:

```
x = float(input('x = '))
y = float(input('y = '))
if x == 0 and y == 0:
    print('this point belongs to area ')
    if x <= 0:
        if y <= x + 5 and y >= x + 2:
            print('this point belongs to area ')
        else:
            print('this point doesn\'t belong to area')
a = input()
```

Приклад виконання:
x = 0
y = -2
this point belongs to area

Звіт до завдання 2:

Завдання 1. Знайти суму ряду S при заданому значенні кількість доданків n та значенні x, тобто знайти суму перших n доданків. Використовувати цикл for та функцію range.

$$S = 1 + \frac{2*3*4*5*x}{1!} + \frac{3*4*5*6*x^2}{2!} + \frac{4*5*6*7*x^2}{3!} + \frac{5*6*7*8*x^4}{4!} + \dots$$

Алгоритм #1:

- Вводимо ціле значення n та дійсне x за допомогою функції input()
- В циклі for: шукаємо кожен член ряду за формулою:

$$a_n = (-1^{n+1})\frac{(n+1)(n+2)(n+3)(n+4)x^n}{n!}$$

- І додаємо до змінної суми.
- Виводимо суму ряду за допомогою функції print().

Алгоритм #2(з використанням рекурентного множника):

- Вводимо ціле значення n та дійсне x за допомогою функції input()
- Шукаємо перший член прогресії за фомулою:
  - $a_1 = 120x$
- За допомогою логічного оператора if() перевіряємо чи n = 1 . Якщо n = 1, тоді виводимо суму $(1 + a_1)$ за допомогою функції print().
- Якщо n != 1, тоді в циклі for: домножуємо на рекурентний множник попередній член ряду і додаємо в змінну суми.
- Рекурентний множник:

$$(-1^{n+1})\frac{(n+1)(n+2)(n+3)(n+4)x^n}{n!} * (-1^n)\frac{(n-1)!}{n(n+1)(n+2)(n+3)x^{n-1}} =$$

$$= -\frac{(n+4)x}{n^2}$$

- Виводимо суму ряду за допомогою функції print().

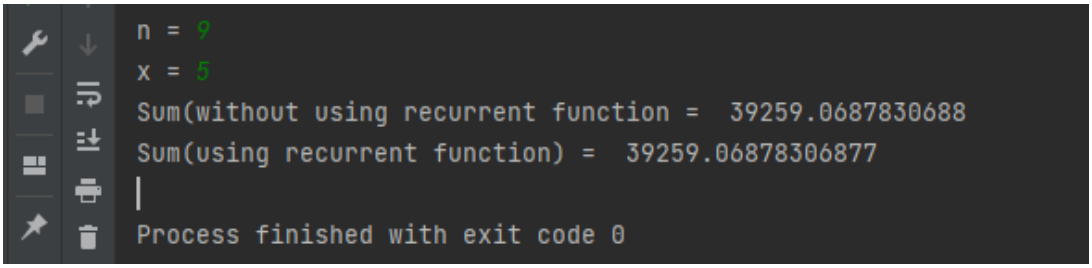
Код:

```
#S = 1 + (2*3*4*5*x)/1!-(3*4*5*6*x^2)/2!+(4*5*6*7*x^3)/3!-(5*6*7*8*x^4)/4!+...
import math
n = int(input("n = "))
x = float(input("x = "))
a1 = 1
Sum1 = a1
for i in range(1,n+1):
    a1 = 1
    a1 *= pow(-1,i+1)
    for j in range(1,5):
        a1 *= i + j
    a1 *= x**(i)
    a1 /= math.factorial(i)
    Sum1 += a1
print("Sum(without using recurrent function = ",Sum1)

a2 = 120*x
Sum2 = a2 + 1
if n == 1:
    print("Sum(using recurrent function) = ",Sum2)
else:
    for i in range(2,n+1):
        a2 *= (-(i+4)*x)/i**2
```

```
        Sum2 += a2
    print("Sum(using recurrent function) = ",Sum2)
```

Приклад виконання:



Завдання 2. Знайти суму ряду S при заданому значенні кількість доданків n та значенні x, тобто знайти суму перших n доданків. Використовувати цикл for та функцію range.

$$S = 1 + \frac{3\sin(x)}{1!} - \frac{5\sin^2(x)}{3!} + \cdots + (-1)^{n+1}\frac{(2n+1)sin^n(x)}{(2n-1)!}$$

Алгоритм #1:

- Вводимо ціле значення n та дійсне x за допомогою функції input()
- В циклі for: шукаємо кожен член ряду за формулою:

  $$a_n = (-1)^{n+1}\frac{(2n+1)sin^n(x)}{(2n-1)!}$$

  І додаємо до змінної суми.
- Виводимо суму ряду за допомогою функції print().

Алгоритм #2(з використанням рекурентного множника):

- Вводимо ціле значення n та дійсне x за допомогою функції input()
- Шукаємо перший член прогресії за фомулою:

  $$a_1 = 3\sin(x)$$
- За допомогою логічного оператора if(): перевіряємо чи n = 1 – якщо n = 1, тоді виводимо суму (1 + a_1) за допомогою функції print().
- Якщо n != 1, тоді в циклі for: домножуємо на рекурентний множник попередній член ряду і додаємо в змінну суми.
  Рекурентний множник:

  $$(-1)^{n+1}\frac{(2n+1)sin^n(x)}{(2n-1)!} * (-1)^n\frac{(2n-3)!}{(2n-1)sin^{n-1}(x)} = \\ = \frac{-(2n+1)\sin(x)}{(2n-1)^2(2n-2)}$$
- Виводимо суму ряду за допомогою функції print().

Код:

```
# S = 1 + (3*sin(x))/1! - (5*(sin(x))^2)/3! + ... + (-
1^n+1)*((2n+1)*(sin(x))^n)/(2n-1)!
import math
n = int(input("n = "))
x = float(input("x = "))
Sum1 = 1
if n == 1:
    print("Sum(without using recurrent function) = ",3*math.sin(x)+1)
```

```
else:
[redacted] function) = ",Sum1)

a2 = 3*math.sin(x)
Sum2 = 1 + a2
if n == 1:
   print("Sum(using recurrent function) = ",Sum2)
else:
   for i in range(2,n+2):
      a2 *= (-1*(2*i+1)*math.sin(x))/((2*i-2)*(2*i-1)**2)
      Sum2 += a2
   print("Sum(using recurrent function) = ",Sum2)
```

Приклад виконання:

```
n = 9
x = 5
Sum(without using recurrent function) =  -2.6960236660219086
Sum(using recurrent function) =  -2.6960236660219086

Process finished with exit code 0
```

……

……

Звіт до завдання 3:

Завдання а. Знайти k перших чисел Фібоначчі, що більші заданого числа A, в записі яких є хоча б одна задана цифра Z. Вивести дані числа.

Алгоритм:

Вводимо ціле значення k, A та Z за допомогою функції input()

- Перевіряємо за допомогою логічного оператора If(): чи k = 1 і чи A < 1 і чи Z = 1.
- Якщо твердження (k = 1 і A < 1 і Z = 1) істине, тоді виводимо одиницю.
- Якщо твердження (k = 1 і A < 1 і Z = 1) неправдиве тоді в циклі while():
  за рекурентною формулою:
  Fn=Fn-1+Fn-2
  шукаємо кожен член послідовності Фібоначчі
- І всередині циклу, використовуючи вкладений цикл, перевіряємо чи член послідовності є більшим за змінну A і чи містить член послідовності хоча б одну цифру Z.
- Виводимо результат.

Код:

```
#Знайти k перших чисел Фібоначчі, що більші заданого числа A,
#в записі яких є хоча б одна задана цифра Z.
#Вивести дані числа.
k = int(input("k = "))
A = int(input("A = "))
Z = int(input("Z = "))
f1 = f2 = 1
checker = True
if k == 1 and A < 1 and Z == 1:
    print("f1 = ",f1)
else:
    if A < 1:
        print("f1 = ", f1)
    n = 3
    while n <= k:
        checker = False
        temp_1 = f2
        l = 0
……
…..
…..
```

Приклад виконання:

.....

.....