

Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського»

Факультет інформатики та обчислювальної техніки

Кафедра обчислювальної техніки

Програмування

Лабораторна робота №2

«Типи даних, змінні та оператори мови
програмування Python»

Виконала:
студентка групи ІО-15
Кушнерик Є.О.
Залікова книжка № 1508

Перевірив:
Пономаренко А.М

Київ – 2021

Тема: «Типи даних, змінні та оператори мови програмування Python».

Мета: вивчити типи даних, які використовуються в мові програмування Python. Змінні та правила їх іменування, операції над змінними. Оператори та їх застосування.

Завдання:

1. Вивчити матеріал лекцій 3, 4, 5 та 6.
2. Виконати індивідуальне завдання лабораторної роботи, вибране відповідно до варіанту.

Теоретичні відомості:

Основні типи даних:

bool – логічний тип даних.

NoneType – об'єкт зі значенням None.

int – цілі числа.

float – дійсні числа.

complex – комплексні числа.

str – Unicode-рядки.

bytes – незмінювана послідовність байтів.

bytearray – змінювана послідовність байтів.

list – списки.

tuple – кортежі.

range – діапазони.

dict – словники.

set – множини.

Математичні оператори:

+ – додавання.

- – віднімання.

***** – множення.

/ – ділення.

// – Ділення з округленням униз.

% – остача від ділення.

****** – піднесення до степеня.

Унарний мінус (**-**) і унарний плюс (**+**) .

Двійкові математичні оператори:

~ – двійкова інверсія.

& – двійкове І.

| – двійкове АБО.

^ – двійкове виключення.

<< – зсув вліво.

>> – зсув вправо.

Оператори для послідовностей:

+ - конкатенація.

***** - повторення.

in - перевірка на входження.

not in - перевірка на невходження

Оператори розгалуження й цикли:

Оператор розгалуження **if ... else;**

If логічний вираз:

інструкції

else:

інструкції, якщо не виконується **if**

while

Початкові дані

While умова1:

Інструкції

Збільшення

Else:

Умова,якщо не виконується умова 1

Числа:

`int` – цілі числа.

`float` – дійсні числа;

`complex` – комплексні числа.

Модуль `math`. Математичні функції. :

`Pi` – повертає число π

`e` – повертає значення константи e :

`sin()`, `cos()`, `tan()` – стандартні тригонометричні функції

`asin()`, `acos()`, `atan()`– обернені тригонометричні функції

`degrees()` – перетворює радіани в градуси.

`radians()`– перетворює градуси в радіани.

`exp()` – експонента.

`log(число[, База])` – логарифм по заданій базі.

`log10()` – десятковий логарифм.

`log2()` – логарифм по базі 2.

`sqrt()` – квадратний корінь.

`ceil()` – значення, округлене до найближчого більшого цілого

`floor()` – значення, округлене до найближчого меншого цілого

`pow(Число,Степінь)` – підносить Число до Степеня.

`fabs ()` – абсолютне значення.

Завдання 1

8	$R = \frac{2.37 \sin(x+1)}{\sqrt{4y^2 - 0.1y + 5}}$
---	---

Код програми

```
1 import math;
2 def isfloat(value):
3     try:
4         float(value)
5         return True
6     except ValueError:
7         return False
8
9 x = input("Введіть число x");
10 while 1:
11     if isfloat(x):
12         x=float(x)
13         break
14     else:
15         print("Помилка! Введіть число!")
16         x=input("Введіть число x\n")
17
18 y = input("Введіть число y");
19 while 1:
20     if isfloat(y):
21         y=float(y)
22         break
23     else:
24         print("Помилка! Введіть число!")
25         y=input("Введіть число y\n")
26 R = float((2.37*(math.sin(x+1)))/(math.sqrt(4*(pow(y,2))-0.1*y+5)));
27 print("Відповідь: R = ", "%.3f"%R);
```

Результат програми

Run: first ×

D:\ProgramData\Anaconda3\envs\pythonProject\python.exe C:/Users/HP8.
Введіть число xjdkk
Помилка! Введіть число!
Введіть число x
-2
Введіть число y3
Відповідь: R = -0.313

Process finished with exit code 0

Алгоритм програми

1. Підключаємо модуль `math`.
2. Вводимо значення змінних `x` та `y`.
3. Для того, щоб не запускати наново програму у випадку введення значень неправильного типу, створюємо функцію, яка перевіряє, чи значення змінної `float`, і результат цієї функції (`True/False`) використовуємо в циклі `while`.
4. Коли усе введено правильно значення присвоюється змінним і цикл переривається за допомогою оператора `break`.
5. Далі програма обчислює приклад та виводить результат обчислень.

Завдання 2

8	<p>Ввести з клавіатури значення для довжин сторін двох трикутників a1, b1, c1 і a2, b2, c2.</p> <ol style="list-style-type: none">1. Довжини сторін трикутника представлені дійсними числами2. Визначити, площа якого трикутника є більшою.3. Відповідь вивести у вигляді повідомлення.4. Передбачити ситуацію, коли площі трикутників дорівнюють одна одній
---	---

Код програми

```
import math;
from decimal import Decimal
def isfloat(value):
    try:
        float(value)
        return True
    except ValueError:
        return False

def ispositive(value):
    if(Decimal(value)>0):return True
    else: return False

def CreateTriangle(a,b,c):
    if ((float(a) + float(b)) > float(c)) & ((float(a) + float(c)) > float(b)) &
    ((float(b) + float(c)) > float(a)): return True
    else: return False

a1 = input("Введіть сторону a1: ");
while 1:
    if isfloat(a1):
        if ispositive(a1):
            a1 = float(a1)
            break
        else:
            print("Помилка! Введіть додатне число!")
            a1 = input("Введіть сторону a1: \n")
    else:
        print("Помилка! Введіть число!")
        a1 = input("Введіть сторону a1: \n")

b1 = input("Введіть сторону b1: ");
while 1:
    if isfloat(b1):
        if ispositive(b1):
            b1 = float(b1)
            break
        else:
            print("Помилка! Введіть додатне число!")
            b1 = input("Введіть сторону b1: \n")
    else:
        print("Помилка! Введіть число!")
        b1 = input("Введіть сторону b1: \n")

c1 = input("Введіть сторону c1: ");
while 1:
    if isfloat(c1):
        if ispositive(c1):
            c1 = float(c1)
            break
        else:
            print("Помилка! Введіть додатне число!")
            c1 = input("Введіть сторону c1: \n")
    else:
        print("Помилка! Введіть число!")
        c1 = input("Введіть сторону c1: \n")

while 1:
    if (CreateTriangle(a1, b1, c1) == True):
        print("Трикутник створений")
        break
    else:
        print("Error! Трикутник не може бути створений! Введіть сторони знову")
        a1 = input("Введіть сторону a1: ");
        while 1:
```

```

        if isfloat(a1):
            if ispositive(a1):
                a1 = float(a1)
                break
            else:
                print("Помилка! Введіть додатне число!")
                a1 = input("Введіть сторону a1: \n")
        else:
            print("Помилка! Введіть число!")
            a1 = input("Введіть сторону a1: \n")

b1 = input("Введіть сторону b1: ");
while 1:
    if isfloat(b1):
        if ispositive(b1):
            b1 = float(b1)
            break
        else:
            print("Помилка! Введіть додатне число!")
            b1 = input("Введіть сторону b1: \n")
    else:
        print("Помилка! Введіть число!")
        b1 = input("Введіть сторону b1: \n")

c1 = input("Введіть сторону c1: ");
while 1:
    if isfloat(c1):
        if ispositive(c1):
            c1 = float(c1)
            break
        else:
            print("Помилка! Введіть додатне число!")
            c1 = input("Введіть сторону c1: \n")
    else:
        print("Помилка! Введіть число!")
        c1 = input("Введіть сторону c1: \n")

a2 = input("Введіть сторону a2: ");
while 1:
    if isfloat(a2):
        if ispositive(a2):
            a2 = float(a2)
            break
        else:
            print("Помилка! Введіть додатне число!")
            a2 = input("Введіть сторону a2: \n")
    else:
        print("Помилка! Введіть число!")
        a2 = input("Введіть сторону a2: \n")

b2 = input("Введіть сторону b2: ");
while 1:
    if isfloat(b2):
        if ispositive(b2):
            b2 = float(b2)
            break
        else:
            print("Помилка! Введіть додатне число!")
            b2 = input("Введіть сторону b2: \n")
    else:
        print("Помилка! Введіть число!")
        b2 = input("Введіть сторону b2: \n")

c2 = input("Введіть сторону c2: ");
while 1:
    if isfloat(c2):
        if ispositive(c2):

```



```

        c2 = float(c2)
        break
    else:
        print("Помилка! Введіть додатне число!")
        c2 = input("Введіть сторону c2: \n")
else:
    print("Помилка! Введіть число!")
    c2 = input("Введіть сторону c2: \n")

while 1:
    if (CreateTriangle(a2,b2,c2)==True):
        print("Трикутник створений")
        break
    else:
        print("Error! Трикутник не може бути створений! Введіть сторони знову")
        a2 = input("Введіть сторону a2: ");
        while 1:
            if isfloat(a2):
                if ispositive(a2):
                    a2 = float(a2)
                    break
            else:
                print("Помилка! Введіть додатне число!")
                a2 = input("Введіть сторону a2: \n")
        else:
            print("Помилка! Введіть число!")
            a2 = input("Введіть сторону a2: \n")

        b2 = input("Введіть сторону b2: ");
        while 1:
            if isfloat(b2):
                if ispositive(b2):
                    b2 = float(b2)
                    break
            else:
                print("Помилка! Введіть додатне число!")
                b2 = input("Введіть сторону b2: \n")
        else:
            print("Помилка! Введіть число!")
            b2 = input("Введіть сторону b2: \n")

        c2 = input("Введіть сторону c2: ");
        while 1:
            if isfloat(c2):
                if ispositive(c2):
                    c2 = float(c2)
                    break
            else:
                print("Помилка! Введіть додатне число!")
                c2 = input("Введіть сторону c2: \n")
        else:
            print("Помилка! Введіть число!")
            c2 = input("Введіть сторону c2: \n")

P1 = (a1 + b1 + c1) / 2;
S1 = math.sqrt(P1 * (P1 - a1) * (P1 - b1) * (P1 - c1));
print("Площа першого трикутника: ", S1)
P2 = (a2 + b2 + c2) / 2;
S2 = math.sqrt(P2 * (P2 - a2) * (P2 - b2) * (P2 - c2));
print("Площа другого трикутника: ", S2)
if(S1!=S2):
    Max = max(S1,S2);
    print("Більша площа: ", "%.3f"%Max)
else: ("Площі однакові: ", "%.3f"%S1)

```

Результат програми

```
Run: second x
D:\ProgramData\Anaconda3\envs\pythonProject\python.exe C:/Users/HP/
Введіть сторону a1: euieu
Помилка! Введіть число!
Введіть сторону a1:
0
Помилка! Введіть додатне число!
Введіть сторону a1:
15
Введіть сторону b1: 22
Введіть сторону c1: 5
Error! Трикутник не може бути створений! Введіть сторони знову
Введіть сторону a1: |

Run: second x
Error! Трикутник не може бути створений! Введіть сторони знову
Введіть сторону a1: 12
Введіть сторону b1: 11
Введіть сторону c1: 12
Трикутник створений
Введіть сторону a2: 3
Введіть сторону b2: 4
Введіть сторону c2: 5
Трикутник створений
Площа першого трикутника: 58.65950477117924
Площа другого трикутник: 6.0
Більша площа: 58.660

Process finished with exit code 0

Run: second x
Введіть сторону a1: 5
Введіть сторону b1: 12
Введіть сторону c1: 13
Трикутник створений
Введіть сторону a2: -1
Помилка! Введіть додатне число!
Введіть сторону a2:
5
Введіть сторону b2: 12
Введіть сторону c2: 13
Трикутник створений
Площа першого трикутника: 30.0
Площа другого трикутник: 30.0
```

Алгоритм програми

1. Створюємо три функції: а) для перевірки, чи число дійсне, б) для перевірки, чи число додатне (бо сторони трикутника можуть набувати лише таких значень) і в) для перевірки, чи можна утворити трикутник з цих сторін.
2. Далі відбувається ввід значень змінних (сторін). Після кожного введення вмикається цикл `while`, в якому запускаються функції для перевірки умов правильності значень сторін. Коли значення введені правильно цикл переривається за допомогою `break`. Після введення усіх сторін вмикається цикл `while` з функцією всередині для перевірки існування трикутника і аналогічно переривається за допомогою оператора `break`, коли все введено правильно. Дані дії виконані для обох трикутників.
3. Далі обчислюємо площі трикутників за формулою Герона. Для цього спочатку знаходимо півпериметр кожного з трикутників ($P1$, $P2$) і підставляємо в формулу площі.
4. За допомогою оператора `if, else` перевіряємо, чи площі не однакові. Якщо однакові, то виводимо про це повідомлення і значення однієї з площ. В іншому випадку знаходимо більшу площу за допомогою функції `max` і виводимо на консоль значення з повідомленням.

Завдання 3

8	<p>1. Ввести з клавіатури дійсне число A (> 1).</p> <p>2. Вивести найменше із цілих чисел N, для яких сума</p> $S = 1 + \frac{1}{2} + \dots + \frac{1}{N}$ <p>буде більше A, (тобто , S>A)</p> <p>3. Вивести значення суми S.</p>
---	--

Код програми

```
1 A = input("Ведіть число більше за 1");
2 N = int(0);
3 S = float();
4 def isfloat(value):
5     try:
6         float(value)
7         return True
8     except ValueError:
9         return False
10 def ispositive(value):
11     if(float(value)>1):return True
12     else: return False
13 while 1:
14     if isfloat(A):
15         if ispositive(A):
16             A = float(A)
17             break
18         else:
19             print("Помилка! ")
20             A = input("Введіть число більше за 1! \n")
21     else:
22         print("Помилка!")
23         A = input("Введіть число: \n")
24 while(S<=A):
25     N=N+1;
26     S+= float(1/N);
27 print("Сума ", "%.3f"%S, " Номер", N);
```

Результат програми

```
Run: first x
D:\ProgramData\Anaconda3\envs\pythonProject\python.exe C:/Us
Ведіть число більше за 1 iudu
Помилка!
Введіть число:
0
Помилка!
Введіть число більше за 1!
4.6
Сума  4.611  Номер 56
Process finished with exit code 0
```

Алгоритм програми

1. Присвоюємо значення змінній A введеного користувачем, а змінній N інтове значення 0.
2. Створюємо дві функції: а) для перевірки, чи число дійсне, б) для перевірки, чи число більше за 1.
3. Запускаємо цикл while, де включаємо вище перераховані функції. Якщо умови введення значення A дотримані, то цикл переривається оператором break.
4. Так, як сума значень послідовності може бути безкінечно більшою за введене значення з клавіатури, далі вмикається цикл з умовою, доки сума не буде наближена до введеного значення з клавіатури (A), програма повторює свої дії. Всередині циклу ми збільшуємо на одиницю значення N (ніби лічильника) і виконуємо умову завдання.
5. Коли цикл закінчив свої дії, виводимо значення суми та найменше ціле число N.

Висновок:

Я навчилась:

№1: Підключати модуль math та користуватись його функціями, щоб рахувати синуси та корені. Також навчилась користуватись арифметичними операторами.

№2 : Створювати функції. Використовувати перевірку типу даних значення змінної. Навчилась використовувати оператори розгалуження, циклу та функцію max.

№3: Використовувати оператори розгалуження, циклу та перевірку даних.