

Учреждение образования
«БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
ИНФОРМАТИКИ И РАДИОЭЛЕКТРОНИКИ»
Кафедра информатики

Отчет по лабораторной работе №3
По дисциплине "Избранные главы информатики"
Python

Выполнил:
студент группы 253503
Купрейчик М.Д.

Руководитель:
доцент
Жвакина А.В.

Минск 2024

Вариант 14

Задание 1. В соответствии с заданием своего варианта составить программу для вычисления значения функции с помощью разложения функции в степенной ряд. Задать точность вычислений ϵ .

Код функции:

```
import math

def pw_function(x, eps=0.1):
    """Function that calculating the value of a function
    using a power series expansion of the function( $\ln(1-x)$ ),
    where
     $|x| < 1$ , takes two arguments  $x$  and  $\epsilon$  - accuracy for
    result."""
    if math.fabs(x) >= 1:
        print("|x| >= 1")
        return
    sum = 0.0
    n = 1
    curr = 0.0
    while n < 501:
        curr = -(math.pow(x, n)/n)
        if math.fabs(curr) <= eps:
            sum += curr
            break
        else:
            sum += curr
            n += 1

    rou = int(math.log10(eps)*-1)

    sum = round(sum, rou)
    Fx = round(math.log(1-x, math.e), rou)
```

```

print(f"|{'x':<5}|{'n':<{len(str(n))+1}}|{'F(x)':<{(rou + 6)}}|{'Math.F(x)':<{(rou + 8)}}|{'epsilon':<{(rou + 7)}}|")
    print(f"|x:<5|n:<{len(str(n))+1}|sum:<{(rou + 6)}}|")
        f"{Fx:<{(rou + 8)}}|eps:<{(rou + 7)}}|")

```

Вывод функции при ее вызове:

```

Choose option between 1-5:
1) PowerSeriesFunction(First task)
2) CycleFunction(Second task)
3) InputStringAnalyseFunction(Third task)
4) PermanentStringAnalyseFunction(Four task)
5) FloatListFuncion(Five task)
Or just 'q' for exit

option:
1

Enter x:
0.01

Enter epsilon:
ihvib
Not any number

0.0000001
|x|n|F(x)|Math.F(x)|epsilon|
|0.01|4|-0.0100503|-0.0100503|1e-07|

```

Задание 2. В соответствии с заданием своего варианта составить программу для нахождения суммы последовательности чисел. Организовать цикл, который принимает целые числа и вычитает их из 10000. Окончание – получение отрицательного итога.

Код функции:

```

def cycle_function():
    """Function that create a loop that takes integers and
    subtracts them from 10000. Ends with a negative result."""
    num = 10000

```

```
print("Input any whole numbers:")

while num >= 0:
    try:
        inp = int(input())
    except ValueError:
        print("Not whole number\n")
        continue

    num -= inp

    if num == 0:
        print("\nRemain number already == 0")
        return
    elif num < 0:
        print("\nRemain number already < 0")
        return
    else:
        print(f"Remain number = {num}\n")
```

Вывод функции при ее вызове:

```
option:
2

Process started:
Input any whole numbers:
1
Remain number = 9999

2
Remain number = 9997

3
Remain number = 9994

3.3
Not whole number

9994

Remain number already == 0

Process ended: Time = 15.790273666381836
```

Задание 3. В соответствии с заданием своего варианта составить программу для анализа текста, вводимого с клавиатуры. В строке, вводимой с клавиатуры, подсчитать количество пробельных символов и запятых.

Код функции:

```
def find_coin():
    """Function that finds number of ' ' and ',' symbols
    in input string."""
    print("Input any string:")
    string = input()

    print(f"\nNumber of ' ' in string: {string.count(' ')}")
```

```
'})\n")  
    print(f"Number of ',' in string: {string.count(',')}")
```

Вывод функции при ее вызове:

```
Choose option between 1-5:  
1) PowerSeriesFunction(First task)  
2) CycleFunction(Second task)  
3) InputStringAnalyseFunction(Third task)  
4) PermanentStringAnalyseFunction(Four task)  
5) FloatListFunc lion(Five task)  
Or just 'q' for exit  
  
option:  
3  
  
Process started:  
Input any string:  
i, love, python!!!!  
  
Number of ' ' in string: 2  
  
Number of ',' in string: 2  
  
Process ended: Time = 22.11329960823059
```

Задание 4. Дана строка текста, в которой слова разделены пробелами и запятыми. В соответствии с заданием своего варианта составьте программу для анализа строки, инициализированной в коде программы.

Код функции:

```
def find_substrings():  
    """Function that operate with string: determine the  
    number of words whose length is less than 7 characters,
```

```

find
    the shortest word ending with the letter 'a' and
    display all words in descending order of their lengths."""
    str = ("So she was considering in her own mind, as
    well as she could, for the hot day made her feel very
    sleepy and "
           "stupid, whether the pleasure of making a
    daisy-chain would be worth the trouble of getting up and
    picking "
           "the daisies, when suddenly a White Rabbit with
    pink eyes ran close by her.")
    str.replace(",", "")

    all_words = str.split(" ")

    dict = {}

    for word in all_words:
        if len(word) in dict:
            dict[len(word)].append(word)
        else:
            dict[len(word)] = [word]

    print("\nAll words with their lenght:")
    for key in sorted(dict.keys(), reverse=True):
        print(f"{key}: {dict[key]}")

    print(f"\nNumber of words with lenght < 7:
    {len([len(word) for word in all_words if len(word) <
    7])}")

    print(f"\nShortest word that ends with 'a' symbol:
    {min([word for word in all_words if word.endswith('a')],
    key=len)}")

```

Вывод функции при ее вызове:

```

Choose option between 1-5:
1) PowerSeriesFunction(First task)
2) CycleFunction(Second task)
3) InputStringAnalyseFunction(Third task)
4) PermanentStringAnalyseFunction(Four task)
5) FloatListFunction(Five task)
Or just 'q' for exit

option:
4

Process started:

All words with their lenght:
11: ['considering', 'daisy-chain']
8: ['pleasure', 'daisies,', 'suddenly']
7: ['stupid,', 'whether', 'trouble', 'getting', 'picking']
6: ['could,', 'sleepy', 'making', 'Rabbit']
5: ['mind,', 'would', 'worth', 'White', 'close']
4: ['well', 'made', 'feel', 'very', 'when', 'with', 'pink', 'eyes', 'her.']
3: ['she', 'was', 'her', 'own', 'she', 'for', 'the', 'hot', 'day', 'her', 'and', 'the', 'the', 'and', 'the', 'ran']
2: ['So', 'in', 'as', 'as', 'of', 'be', 'of', 'up', 'by']
1: ['a', 'a']

Number of words with lenght < 7: 45

Shortest word that ends with 'a' symbol: a

Process ended: Time = 0.0009832382202148438

```

Задание 5. В соответствии с заданием своего варианта составить программу для обработки вещественных списков. Программа должна содержать следующие базовые функции.

Код функции:

```

import GeneratorAndDecoratorFile

def float_list_function():
    """This function finds the number of the minimum negative
    list element and the sum of the list elements located

```



```

        between the first and second negative elements."""
        print("\nInput float numbers: (enter 'q' for exit
inputting)")
        flist =
list(GeneratorAndDecoratorFile.input_generator())

        print(f"\nList: {flist}")

        min_neg_index = None
        min_neg_value = None

        for i, value in enumerate(flist):
            if value < 0 and (min_neg_value is None or value <
min_neg_value):
                min_neg_value = value
                min_neg_index = i

        if min_neg_value != None and min_neg_index != None:
            print(f"Index of minimal negative number: index
{min_neg_index}, value {flist[min_neg_index]}\n")
        else:
            print("Input not have any negative number")
            return

        all_negative = [i for i, value in enumerate(flist) if
value < 0]

        sum_ = 0

        if len(all_negative) >= 2:
            sum_ = sum(flist[all_negative[0] +
1:all_negative[1]])
        else:
            print("Input flist not have 2 negative numbers")
            return

```

```
print(f"Sum between two first negative numbers: sum  
{sum_}")
```

Вывод функции при ее вызове:

```
option:  
5  
  
Process started:  
  
Input float numbers: (enter 'q' for exit inputting)  
1.1  
-1.2  
1.3  
1.4  
1.5  
-1.6  
q  
  
List: [1.1, -1.2, 1.3, 1.4, 1.5, -1.6]  
Index of minimal negative number: index 5, value -1.6  
  
Sum between two first negative numbers: sum 4.2  
  
Process ended: Time = 20.417444229125977
```