Министерство образования Республики Беларусь

Белорусский Национальный Технический Университет

Факультет информационных технологий и робототехники

Кафедра «Программное обеспечение вычислительной техники

и автоматизированных систем»

**Отчёт**

по лабораторной работе № 6

по дисциплине ***«Языки программирования»***

тема: «Функции. Рекурсивные алгоритмы»

|  |  |  |
| --- | --- | --- |
| Исполнитель: |  | студент группы 10701118  Воробей Иван Александрович |
| Преподаватель: |  | Белова С.В. |

2018-2019 учебный год

Цель работы: познакомиться с наиболее востребованными стандартными функциями Python, изучить синтаксис объявления и использования пользовательских функций; познакомиться с рекурсивными алгоритмами; научиться проектировать и разбивать большую программу на мелкие фрагменты (функции); закрепить знания на примере разработки интерактивных приложений.

Основное задание

1. Рекурсивно описать функцию f(x, n), вычисляющую xn/n! при любом действительном x и любом неотрицательном целом n.

2. Рекурсивно описать функцию pow(x, n), вычисляющую xn для любого действительного x ≠ 0 и любого целого n.

3. Реализовать функцию, которая вычисляет N-й элемент ряда Фибоначчи с использованием рекурсивного алгоритма. На базе данной функции разработать программу, которая должна предлагать пользователю следующие возможности:

1) вывод конкретного элемента последовательности;

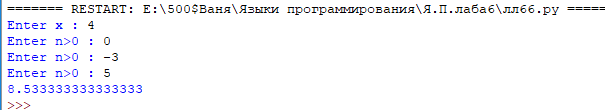
2) вывод всех элементов до указанного пользователем элемента;

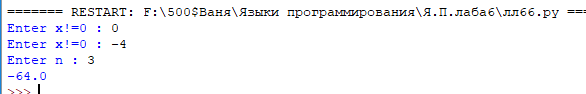
3) вывод части последовательности, значение последнего элемента которой не превосходит введённого пользователем значения. Индивидуальное задание:

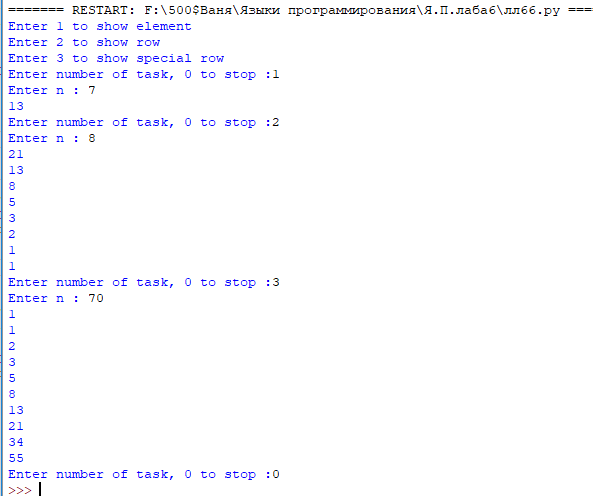
Требования к выполнению:

Программа должна обязательно быть снабжена комментариями, в которых необходимо указать краткое предназначение программы, её версию, ФИО разработчика, номер группы и дату разработки.

Результаты выполнения основного задания:







Ответы на контрольные вопросы:

1.Что такое структурное программирование?(представление программы в виде иерархической структуры блоков)

2. Что такое функция? Как описывается функция в Python?”ЭТО объект, принимающий аргументы и возвращающий значение.

def add(x,y)

return x+y

3. Зачем нужны функции?(для упрощения кода, для удобства)

4. Для чего используется оператор return в функциях?(Что бы вернуть значение) Как возвратить из функции несколько значений?(перечислить значения через запятую)

5. Чем формальные параметры отличаются от фактических?(формальные-указываемые при описании программыб фактические-указываемые при вызове программы)

6. Что такое позиционные параметры?(использыется для обозначения аргумента, который может быть передан как позиционный.этот тип параметра используется по умолчанию)

7. Что такое параметры по умолчанию?(это параметр функции, который имеет определенное значение по умолчанию)

8. Чем отличается глобальная переменная от локальной?локальная переменная объявляется внутри функции и работает только в этоу функции, глобальная работает везде)

9. Зачем применяются рекурсивные алгоритмы?(для решения некоторых задач боие рационально использывать рекурсию.Программа обращается сама к себе)

10. В чём заключается смысл принципа единственной ответственности (Single Responsibility Principle) при грамотной разработки единиц программного кода (в частности, функций)?( Принцип единственной ответственности (The Single Responsibility Principle) декларирует разграничение ответственности между функциями: за решение одной конкретной задачи должна отвечать одна функция. Использование принципа единственной ответственности при работе с функциями является хорошим стилем программирования. )

Что я освоил в процессе выполнения лабораторной работы (выводы):

1. Как вызывать функции в языке pyton
2. Научился использывать рекурсию для возведения числа в степень.
3. Научился работать с рядом фибоначи с помощью рекурсии.
4. Узнал зачем нужны функции.

Листинг исходных кодов программ

#lab 6 Vorobei Ivan

#10701118

#Main task number 1

x=float(input("Enter x : "))

while True:

n=float(input("Enter n>0 : "))

if n>0: break

def stepen(x,n):

if n==1:

return x

return (x/n)\*stepen(x,n-1)

print(stepen(x,n))

#lab6 task 2

#Ivan Varabei

#functiom pow

while True:

x=float(input("Enter x!=0 : "))

if x!=0: break

n=int(input("Enter n : "))

def pow(x,n):

if n==1:

return x

return (x)\*pow(x,n-1)

print(pow(x,n))

#lab 6 task 3

#Ivan Varabei

#fibonachi

def fib(n):

if n==1 or n==2:

return 1

return fib(n-1)+fib(n-2)

print("Enter 1 to show element\nEnter 2 to show row\nEnter 3 to show special row")

while True:

k=int(input("Enter number of task, 0 to stop :"))

if k==0:

break

if k==1:

n=int(input("Enter n : "))

print(fib(n))

if k==2:

n=int(input("Enter n : "))

while n>0:

print(fib(n))

n-=1

if k==3:

n=int(input("Enter n : "))

x=int(n)

n=1

while True:

if fib(n)>=x:

break

print(fib(n))

n+=1