Министерство науки и высшего образования РФ

ФГАОУ ВО Пермский национальный исследовательский  
политехнический университет

Кафедра «Вычислительная математика, механика и биомеханика»

Отчёт по лабораторной работе № 2  
тема «Линейные алгоритмы»  
по дисциплине «Информатика»

Выполнил: студент группы ПМ-23-1б Кузнецов Д.В.

Проверил: ст. пр. каф. ВММБ Ильиных Г.В.

Пермь, 2023

**Задание 1**

* 1. ***Постановка задачи***

Написать функцию, которая определяет, оканчивается ли число на чётную цифру. Написать функцию которая определяет, является ли введённое число больше заданного. Создать программу, которая в бесконечном цикле получает от пользователя числа и проверяет их с помощью двух функций.

* 1. ***Алгоритм выполнения***

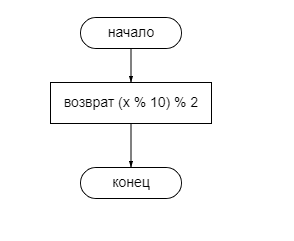
**

Рис.1 – Блок-схема *func1(x)*

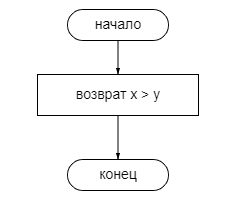


Рис.2 – Блок-схема *func2(x,y)*

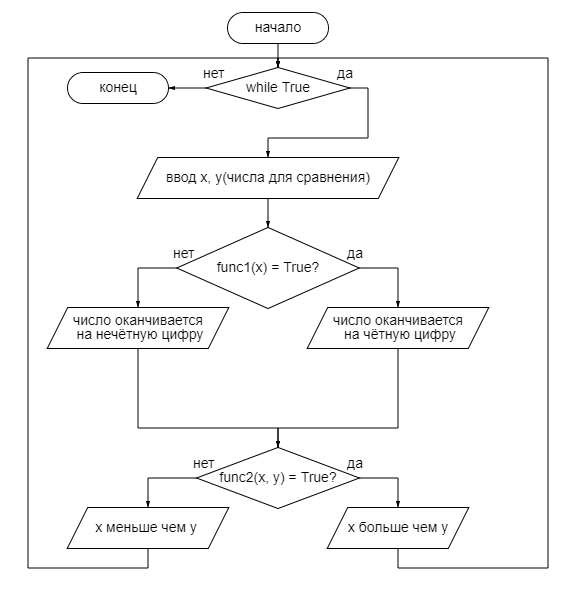


Рис.3 – Блок-схема программы №1

* 1. ***Код программы***

def task1():  
 def func1(x):  
 return (x % 10) % 2 == 0:  
  
 def func2(x, y):  
 return x > y  
  
 x = int(input("Введите число "))  
 y = int(input("Введите число для сравнения "))  
 if func1(x):  
 print("Число оканчивается на чётную цифру")  
 else:  
 print("Число оканчивается на нечётную цифру")  
 if func2(x, y):  
 print(f'{x} больше чем {y}')  
 else:  
 print(f'{x} меньше чем {y}'))

* 1. ***Тестирование работы программы с проверкой***

Для проверки решений в MS Excel была создана таблица:  
В ячейке A1 было записано заданное значение(x), в ячейке B1 было записано  
число для сравнения(y). В ячейку С1 была записана func1, в ячейку D1 func2  
Ячейки C1 и D1 были заданы формулами:  
C1 - =ЕСЛИ(ОСТАТ(ОСТАТ(A1;10);2) = 0;"правда"; "ложь")  
D1 - =ЕСЛИ(A1 > B1; "правда"; "ложь")

Таблица 1 – Тестирование работы и проверка решений программы №1

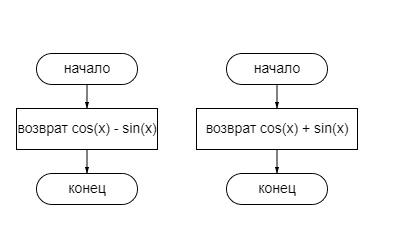
|  |  |  |
| --- | --- | --- |
| №п.п | Решение Python | Решение MS Excel |
| 1 | Введите число **1234**  Введите число для сравнения **4321**  Число оканчивается на чётную цифру 1234 меньше чем 4321 | |  |  |  |  | | --- | --- | --- | --- | | x | y | func1 | func2 | | 1234 | 4321 | правда | ложь | |
| 2 | Введите число **999**  Введите число для сравнения **51083**  Число оканчивается на нечётную цифру  999 меньше чем 51083 | |  |  |  |  | | --- | --- | --- | --- | | x | y | func1 | func2 | | 999 | 51083 | ложь | ложь | |
| 3 | Введите число **32132**  Введите число для сравнения **32123**  Число оканчивается на чётную цифру  32132 больше чем 32123 | |  |  |  |  | | --- | --- | --- | --- | | x | y | func1 | func2 | | 32132 | 32123 | правда | правда | |

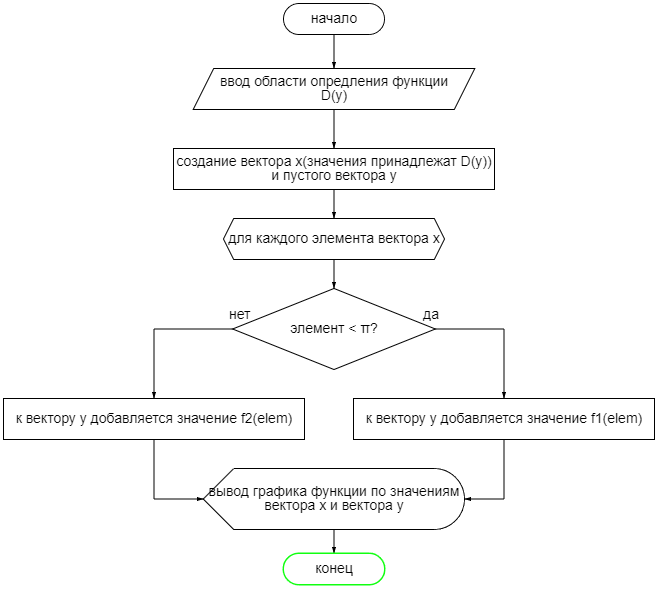
**Задание 2**

***2.1 Постановка задачи***

По формуле вычислить значения функции f(x) в диапазоне значений   
x ∈ [a;b] и представить в виде графика:

***2.2 Алгоритм решения***

Рис.4- Блок-схема *f1(x)* и *f2(x)*

  
Рис.5 – Блок-схема программы №2

***2.3 Код программы***

import matplotlib.pyplot as plt  
import numpy as np  
  
  
def task2():  
 def f1(x):  
 return np.cos(x) - np.sin(x)  
  
 def f2(x):  
 return np.cos(x) + np.sin(x)  
 d\_y = (int(input("введите область определения функции \n")), int(input()))  
 vector\_x = np.linspace(d\_y[0], d\_y[1], 200)  
 vector\_y = []  
 for elem in vector\_x:  
 if elem < np.pi:  
 vector\_y.append(f1(elem))  
 else:  
 vector\_y.append(f2(elem))  
 plt.plot(vector\_x, vector\_y, color='r')  
 plt.grid()  
 plt.show()

***2.4 Тестирование работы программы с проверкой***

Для проверки построения графика используется сайт Wolfram Alpha.

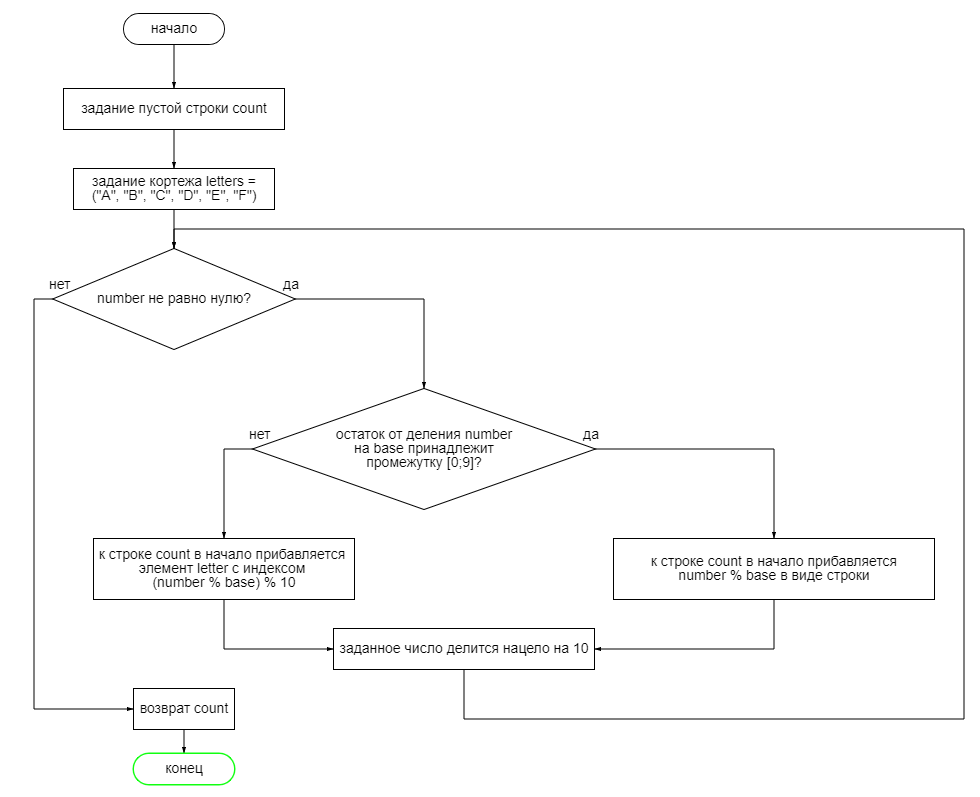
Таблица 2 - Тестирование работы и проверка решений программы №2

|  |  |  |
| --- | --- | --- |
| № | Решение Python | График Wolfram Alpha |
| 1 | введите область определения функции **1 10** |  |
| 2 | введите область определения функции  **-10 0** |  |
| 3 | введите область определения функции **200 210** |  |

**Задание 3  
*3.1 Постановка задачи***

Написать функцию *number\_in\_new\_numeral\_system(number, base)*, которая переводит целое десятичное число в СС с выбранным основанием. Написать программу, которая переводит заданное пользователем значение в заданную пользователем СС.

***3.2 Алгоритм решения***

Рис.6 – Блок-схема функции *number\_in\_new\_numeral\_system(number, base)*

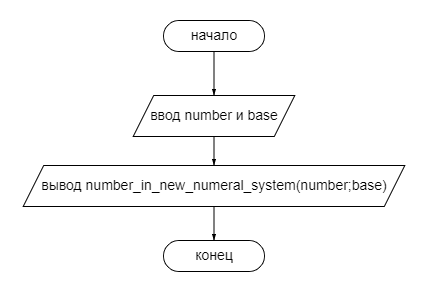


Рис.7 – Блок-схема программы №3

***3.3 Код программы***

def task3():  
  
 def number\_in\_new\_numeral\_system(number, base):  
 count = ''  
 letters = ["A", "B", "C", "D", "E", "F"]  
 while number != 0:  
 if 0 <= number % base <= 9:  
 count = str(number % base) + count  
 else:  
 count = letters[(number % base) % 10] + count  
  
 number //= base  
 return count  
  
 user\_count = int(input("Введите число в 10-чной СС "))  
 cs\_base = int(input("Введите основание новой СС "))  
  
 print(number\_in\_new\_numeral\_system(user\_count, cs\_base))

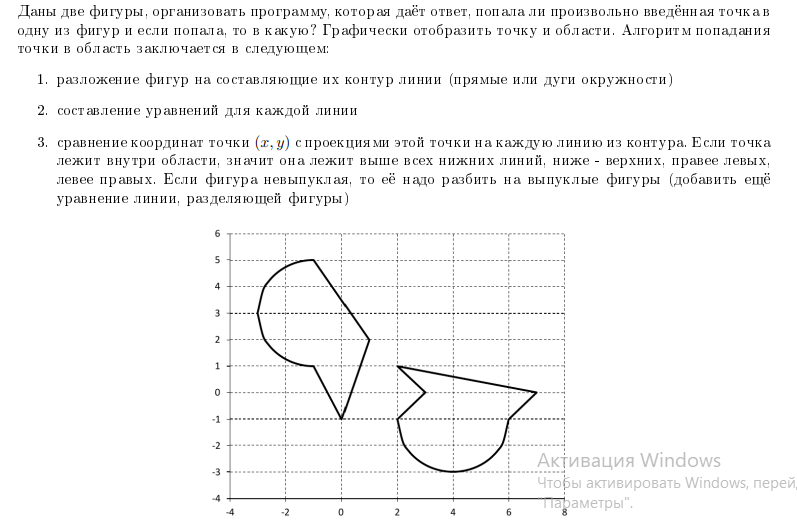
***3.4 Тестирование работы программы с проверкой***

Для проверки решений используется онлайн-калькулятор систем счисления.

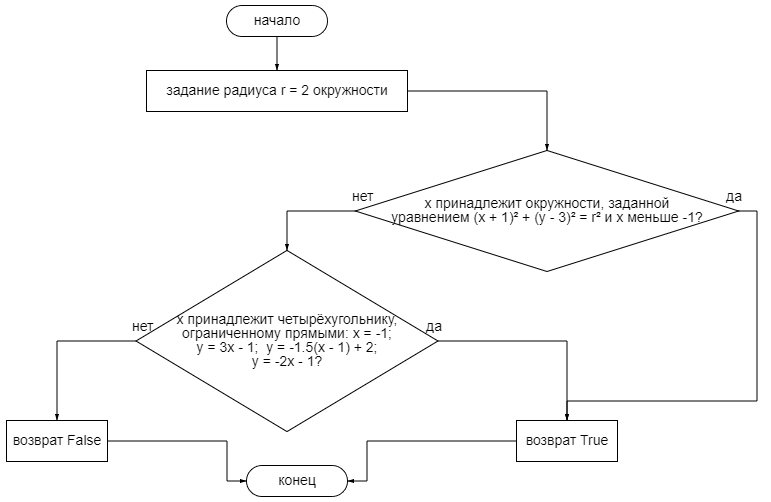
Таблица 3 - Тестирование работы и проверка решений программы №3

|  |  |  |
| --- | --- | --- |
| №п.п | Решение Python | Решение онлайн-калькулятора |
| 1 | Введите число в 10-чной СС **322**  Введите основание новой СС **13**  1BA |  |
| 2 | Введите число в 10-чной СС **5243**  Введите основание новой СС **6**  40135 |  |
| 3 | Введите число в 10-чной СС **729**  Введите основание новой СС **3**  1000000 |  |

**Задание 4**

***4.1 Постановка задачи***

***4.2 Алгоритм решения***

  
Рис.9 – Блок-схема функции *check1(x, y)*