**ФГБОУ ВО “МОСКОВСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ”**

**Лабораторная работа № 3**

Разветвляющиеся вычислительные процессы

**Задание 1 Вариант № 10**

по дисциплине:

Основы программирования

Выполнил:

студент 1 курса

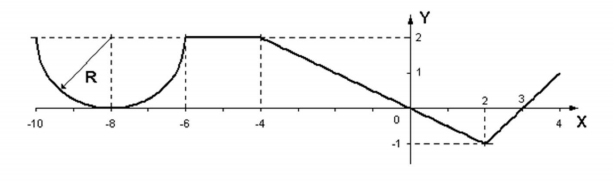
группы 201-322

Кириллов Д.П

**МОСКВА 2020**

**Постановка задачи**

Написать программу, которая по введённым значениям вычисляет все значения функции заданной графически на интервале от xStart до xEnd с шагом dx

**Теоретическая часть**

Для решения задачи использован оператор ветвления, который в языке Python имеет следующий вид:

**while** <Логическое выражение>:

<Инструкции>

Для ввода данных используется инструкция input(), которая возвращает введенную строку. Она должна быть преобразована в числовое значение для дальнейшей работы в вычислениях

Вывод данных выполняется инструкцией print(), в которой используется форматированный вывод

Функция представлена фрагментами прямых линийи дугами кругов

**Описание программы**

Программа написана на алгоритмическом языке Python 3.6, реализована в среде OC Windows 10 и состоит из частей, отвечающих за ввод данных, вычисление и представление данных на экране монитора.

**Описание алгоритма**

1. Ввести значение xStart, xEnd, dx
2. Вывести на монитор входные данные
3. Вывести на монитор шапку таблицы
4. Присвоить стартовому значению x = xStart
5. Вычислить значение функции и вывести на экран в виде строки таблицы
6. Присвоить x = xStart + dx
7. Если x < xEnd, то вернуться к 5 пункту
8. Завершить рисование таблицы и работу программы

**Описание входных и выходных данных**

Входные данные поступают с клавиатуры, а выходные – выводятся на монитор для просмотра. Входные и выходные данные имеют тип float.

**Листинг программы**

from math import (sqrt, fabs)

try:

xStart = float(input('Введите первую границу: '))

xEnd = float(input('Введите вторую границу: '))

dx = float(input('Введите интвервал (dx): '))

if (dx <= 0):

raise ValueError

def f(x):

if x <= -6:

D = 16 - 4 \* (x + 8)\*\*2

y1 = (4 - sqrt(D))/2

y2 = (4 + sqrt(D))/2

if (y1 < 0 or y1 > 2):

return y2

else:

return y1

if x >= -6 and x <= -4:

return 2

if x >= -4 and x <= 2:

if x == 0:

y = fabs(-3 \* x/6)

else:

y = -3 \* x/6

return y

if x >= 2:

return x - 3

# Вывод входных данных

print('')

print('=====================================')

print('xStart = {0:.1f}, xEnd = {1:.1f}, dx = {2:.1f}'.format(xStart, xEnd, dx))

print('=====================================')

print('')

print('╔======================╗')

print('║ X ║ Y ║')

print('║======================║')

# Вывод таблицы с аргументом и результатом функции

x = xStart

while (x < xEnd):

print('║ {0:.1f} ║ {1:.1f} ║'.format(x, f(x)))

x += dx

print('╚=====================╝')

except ValueError:

print('Введите корректное число.')

**Результат тестирования программы**

=====================================

xStart = -10.0, xEnd = 10.0, dx = 1.0

=====================================

╔======================╗

║ X ║ Y ║

║======================║

║ -10.0 ║ 2.0 ║

║ -9.0 ║ 0.3 ║

║ -8.0 ║ 0.0 ║

║ -7.0 ║ 0.3 ║

║ -6.0 ║ 2.0 ║

║ -5.0 ║ 2.0 ║

║ -4.0 ║ 2.0 ║

║ -3.0 ║ 1.5 ║

║ -2.0 ║ 1.0 ║

║ -1.0 ║ 0.5 ║

║ 0.0 ║ 0.0 ║

║ 1.0 ║ -0.5 ║

║ 2.0 ║ -1.0 ║

║ 3.0 ║ 0.0 ║

║ 4.0 ║ 1.0 ║

║ 5.0 ║ 2.0 ║

║ 6.0 ║ 3.0 ║

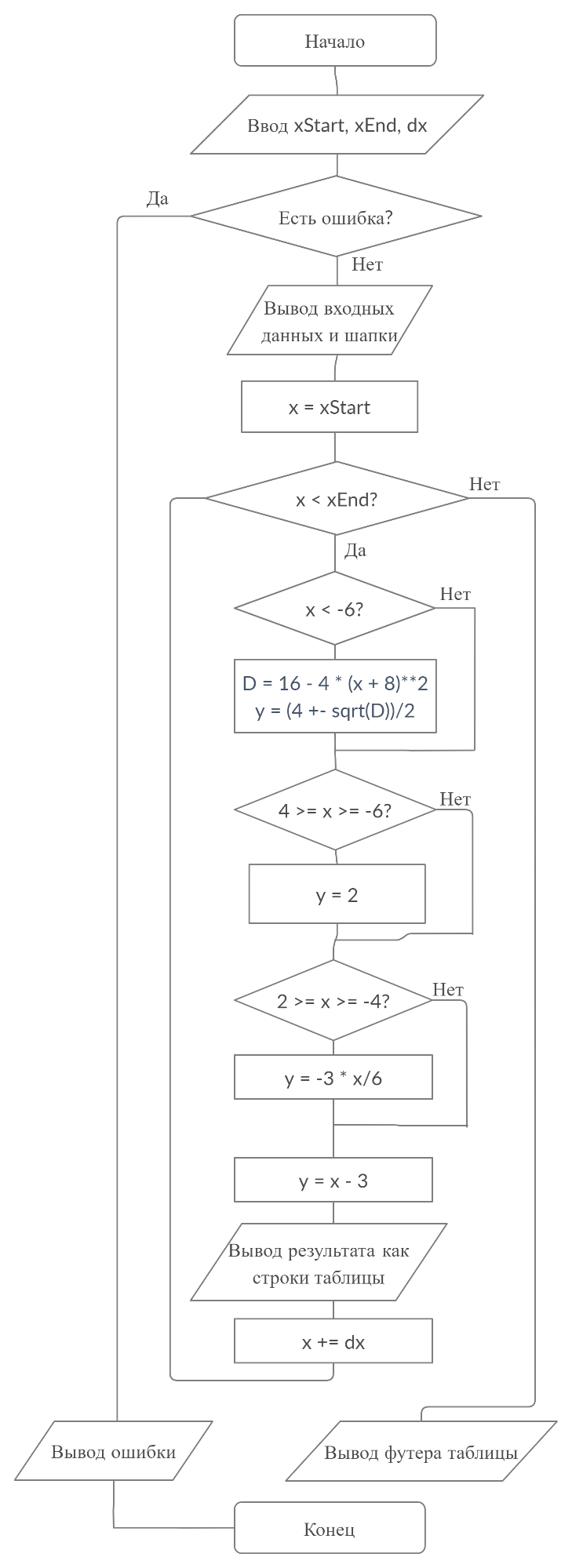
║ 7.0 ║ 4.0 ║

║ 8.0 ║ 5.0 ║

║ 9.0 ║ 6.0 ║

╚======================╝

**Блок-схема**



**Использованная литература**

pythonworld.ru Справочник для начинающих. Python 3