Федеральное государственное автономное

образовательное учреждение

высшего образования

«СИБИРСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

Институт космических и информационных технологий

Кафедра вычислительной техники

**ОТЧЕТ О ПРАКТИЧЕСКОЙ** **РАБОТЕ № 7**

Табулирование функции  
 Вариант № 19

Преподаватель \_\_\_\_\_\_\_\_ Пушкарев К. В.

подпись, дата

Студент КИ18-09б, 031831293 \_\_\_\_\_\_ 28.10.2018 Овсянников В.А.

подпись

Красноярск 2018

**1 Дополнительное упражнение № 2, вариант 9**

Протабулировать функцию на интервале с шагом . Дополнительно заданы координаты произвольной точки на плоскости. Найти наиболее удаленную точку на кривой от точки заданной дополнительно.

**2 Цели работы**

1. Продолжить изучение операторов цикла;
2. Получить навыки разработки циклических программ;
3. Получить навыки по отладке и тестированию программ с использованием функций.

**3 Подход к решению задачи**

Согласно условию задачи, требуется разработать графическую схему алгоритма и программу, которая протабулирует функцию на интервале . Для этого создадим переменные A, B, h, x0, y0, d1, x, y, i типа float и функцию , которая будет вычислять значение . Создадим цикл, который будет табулировать функцию и выводить её значения на экран, посредством подставления нового числа интервала в функцию каждую итерацию, переменной d1 каждую итерацию будет присваиваться расстояние от текущей точки функции до точки введённой пользователем, и если оно будет больше предыдущего значения, переменным x и y будут присвоены значения текущих координат функции. После завершения цикла на экран будут выведены координаты наиболее удалённой точки от точки введённой пользователем.

**4 Графическая схема алгоритма**

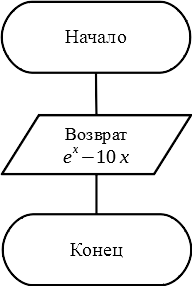


Рисунок 1 – Функция .

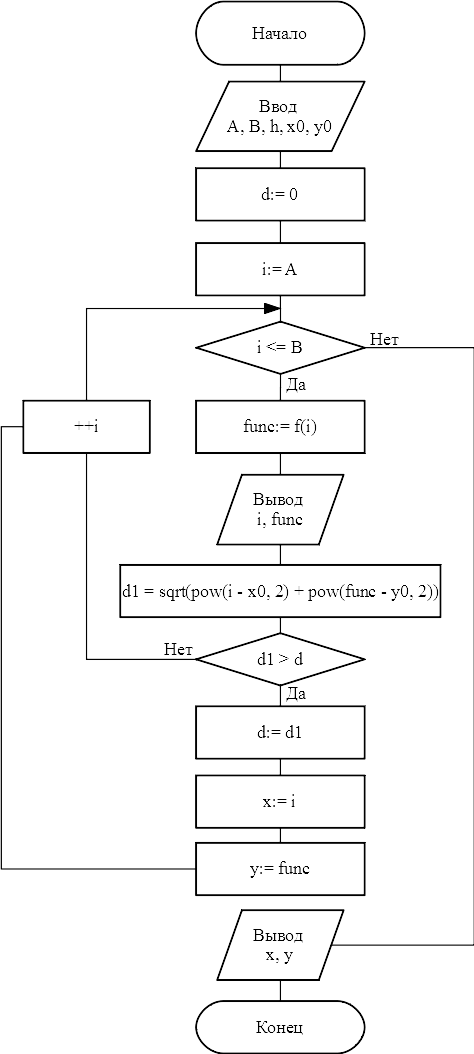


Рисунок 2 – функция

**5 Код программы**

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27 28 29 30 31 32 33 34 35 36 37 38 39  40  41  42  43  44  45  46  47  48  49  50  51 | #include "pch.h"  #include <iostream>  #include <clocale>  #include <cmath>  using namespace std;  float f(float x);  int main(){  setlocale(LC\_ALL, "rus");  float A, B, h, x0, y0, d1, x, y, i;    cout << "Введите начальную точку интервала: ";  cin >> A;  cout << "Введите конечную точку интервала: ";  cin >> B;  cout << "Введите шаг на интервале: ";  cin >> h;    cout << "Введите координату x0: ";  cin >> x0;  cout << "Введите координату y0: ";  cin >> y0;  cout << endl;  float d = 0;  for (i = A; i <= B; i += h) {  float func = f(i);  cout <<"(" << i << "; " << func << ")" << endl;  d1 = sqrt(pow(i - x0, 2) + pow(func - y0, 2));  if (d1 > d) {  d = d1;  x = i;  y = func;  }  }  cout << "\nНаиболее удаленная точка: (" << x << "; " << y << ")" << endl;  return 0;  }  float f(float x) {  return exp(x) - 10 \* x;  } |

**6 Результаты выполнения экспериментальной части работы**

Результаты запуска программы с различными входными значениями приведены в таблице 1.

Таблица 1 – Работа программы на тестовых наборах данных

| Входные данные | Результат |
| --- | --- |
| A = 0,  B = 2,  h = 0.5,  x0 = -1,  y0 = 100 | (0; 1)  (0.5; -3.35128)  (1; -7.28172)  (1.5; -10.5183)  (2; -12.6109)  Наиболее удаленная точка: (2; -12.6109) |
| A = -10,  B = 0,  h = 2,  x0 = 0,  y0 = 0 | (-10; 100)  (-8; 80.0003)  (-6; 60.0025)  (-4; 40.0183)  (-2; 20.1353)  (0; 1)  Наиболее удаленная точка: (-10; 100) |
| A = 7.6,  B = 9.7,  h = 0.3,  x0 = 8.5,  y0 = 6958.3 | (7.6; 1922.2)  (7.9; 2618.28)  (8.2; 3558.95)  (8.5; 4829.77)  (8.8; 6546.25)  (9.1; 8864.3)  (9.4; 11994.4)  Наиболее удаленная точка: (9.4; 11994.4) |