МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

федеральное государственное автономное образовательное учреждение высшего образования

«САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ АЭРОКОСМИЧЕСКОГО ПРИБОРОСТРОЕНИЯ» (ГУАП)



КАФЕДРА ВЫЧИСЛИТЕЛЬНЫХ СИСТЕМ И СЕТЕЙ

Преподаватель

канд. техн. наук, доцент Л.Н. Бариков

Отчет

по лабораторной работе №2  
по дисциплине ОСНОВЫ ПРОГРАММИРОВАНИЯ

на тему: «Операторы ветвления»

Работу выполнил

студент гр. 4143 А.И. Круглов

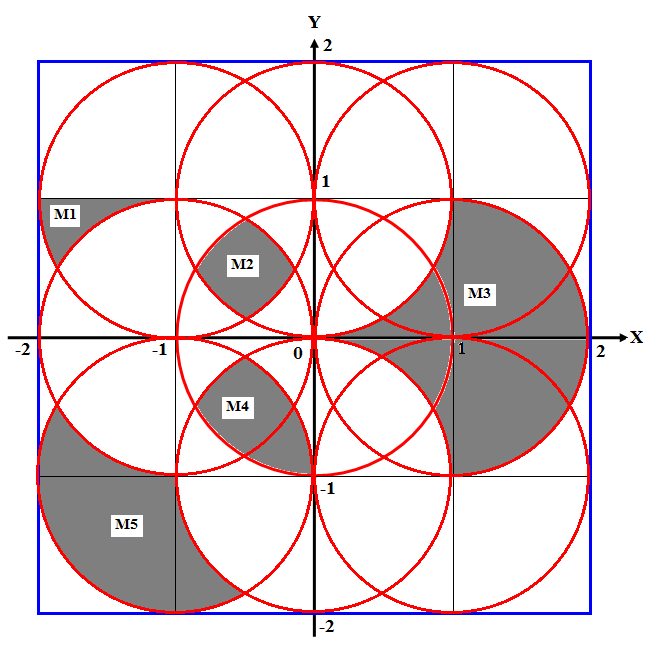
Санкт-Петербург

2022

**Вариант №18**

***Цель лабораторной работы:*** *приобретение навыков структурного программирования на языке C/C++ при решении задач с использованием операторов ветвления* ***if*** *и* ***switch****.*

***Задание на программирование:*** *используя технологию структурного программирования разработать программу для решения индивидуальной задачи по определению места нахождения точки с произвольно заданными координатами на координатной плоскости. В случае если точка попадает в одну из выделенных областей, программа должна определять площадь этой области по аналитической формуле.*

***Математическая модель***

1. Принадлежность области M1:
   1. (x + 1)2 + (y - 1)2 < 1 *{ В левой верхней окружности }*
   2. (x + 1)2 + y2 > 1 *{ Вне левой средней окружности }*
   3. y < 1 *{ Ниже линии y = 1 }*
   4. x < -1 *{ Левее линии x = -1 }*
2. Принадлежность области M2:
   1. x2 + (y - 1)2 < 1 *{ В средней верхней окружности }*
   2. (x + 1)2 + (y - 1)2 < 1 *{ В левой верхней окружности }*
   3. x2 + y2 < 1 *{ В центральной окружности }*
   4. (x + 1)2 + y2 < 1 *{ В левой средней окружности }*
3. Принадлежность области M3:
   1. (x - 1)2 + y2 < 1 *{ В правой средней окружности }*
   2. x2 + (y - 1)2 > 1 *{ Вне средней верхней окружности }*
   3. x2 + (y + 1)2 > 1 *{ Вне средней нижней окружности }*
4. Принадлежность области M4:
   1. x2 + y2 < 1 *{ В центральной окружности }*
   2. x2 + (y + 1)2 < 1 *{ В средней нижней окружности }*
   3. (x + 1)2 + (y + 1)2 < 1 *{ В левой нижней окружности }*
5. Принадлежность области M5:
   1. (x + 1)2 + (y + 1)2 < 1 *{ В левой нижней окружности }*
   2. x2 + (y + 1)2 > 1 *{ Вне средней нижней окружности }*
   3. (x + 1)2 + y2 > 1 *{ Вне левой средней окружности }*
6. Площадь области М1: Площадь сектора (π / 12) – площадь сегмента ((π / 3 – √3 / 2) / 2) ≈ 0,1712
7. Площадь области М2: Площадь квадрата ((√3 – 1) 2 / 2) + 4 × площадь сегмента ((π / 6 – ½) / 2) ≈ 0,3151
8. Площадь области М3: Площадь сектора (π / 2) + 2 × (площадь квадрата (12) – площадь сектора (π / 4)) = 2
9. Площадь области М4: Площадь сектора (π / 12) + 2 × площадь сегмента ((π / 3 – √3 / 2) / 2) ≈ 0,4429
10. Площадь области М5: Площадь сектора (π / 4) + 2 × (площадь сектора (π / 12) – площадь сегмента ((π / 3 – √3 / 2) / 2)) ≈ 1,1278

***Текст программы***

#include <iostream>

#include <math.h>

using namespace std;

int main()

{

float x, y;

cout << "Введите координаты точки: x, y ";

cin >> x >> y;

cout.precision(4);

if

(

pow((x + 1), 2) + pow((y - 1), 2) < 1 &&

pow((x + 1), 2) + pow(y, 2) > 1 &&

y < 1 &&

x < -1

)

cout << "Точка находится в области М1. S = " << (M\_PI / 12) - ((M\_PI / 3 - sqrt(3) / 2) / 2);

else if

(

pow(x, 2) + pow((y - 1), 2) < 1 &&

pow((x + 1), 2) + pow((y - 1), 2) < 1 &&

pow(x, 2) + pow(y, 2) < 1 &&

pow((x + 1), 2) + pow(y, 2) < 1

)

cout << "Точка находится в области М2. S = " << (pow((sqrt(3) - 1), 2) / 2) + 4 \* ((M\_PI / 6 - 0.5) / 2);

else if

(

pow((x - 1), 2) + pow(y, 2) < 1 &&

pow(x, 2) + pow((y - 1), 2) > 1 &&

pow(x, 2) + pow((y + 1), 2)

)

cout << "Точка находится в области М3. S = " << (M\_PI / 2) + 2 \* (1 - M\_PI / 4);

else if

(

pow(x, 2) + pow(y, 2) < 1 &&

pow(x, 2) + pow((y + 1), 2) &&

pow((x + 1), 2) + pow((y + 1), 2) < 1

)

cout << "Точка находится в области М4. S = " << (M\_PI / 12) + 2 \* ((M\_PI / 3 - sqrt(3) / 2) / 2);

else if

(

pow((x + 1), 2) + pow((y + 1), 2) < 1 &&

pow(x, 2) + pow((y + 1), 2) > 1 &&

pow((x + 1), 2) + pow(y, 2) > 1

)

cout << "Точка находится в области М5. S = " << (M\_PI / 4) + 2 \* (M\_PI / 12 - ((M\_PI / 3 - sqrt(3) / 2) / 2));

else

cout << "Точка находится вне выделенных областей";

return 0;

}

***Контрольные примеры***







