

using System;

class Program

{

static double CalculateFunction(double x, double R)

{

if (x >= -6 && x <= 0)

{

// Полуокружность

return Math.Sqrt(R \* R - (x + 3) \* (x + 3));

}

else if (x > 0 && x <= 9)

{

// Прямая линия

return x / 9;

}

else

{

// Если x вне диапазона, возвращаем double.NaN

return double.NaN;

}

}

static void Main()

{

// Ввод параметра R

Console.Write("Введите значение параметра R: ");

double R = double.Parse(Console.ReadLine());

// Ввод начального значения x

Console.Write("Введите начальное значение x: ");

double xStart = double.Parse(Console.ReadLine());

// Ввод конечного значения x

Console.Write("Введите конечное значение x: ");

double xEnd = double.Parse(Console.ReadLine());

// Ввод шага dr

Console.Write("Введите шаг dr: ");

double dr = double.Parse(Console.ReadLine());

// Вывод заголовка таблицы

Console.WriteLine("Таблица значений функции");

Console.WriteLine("------------------------");

Console.WriteLine("x\t\tf(x)");

Console.WriteLine("------------------------");

// Вычисление и вывод значений функции

for (double x = xStart; x <= xEnd; x += dr)

{

double result = CalculateFunction(x, R);

if (!double.IsNaN(result))

{

Console.WriteLine(x.ToString("F2") + "\t\t" + result.ToString("F2"));

}

else

{

Console.WriteLine(x.ToString("F2") + "\t\tВне диапазона");

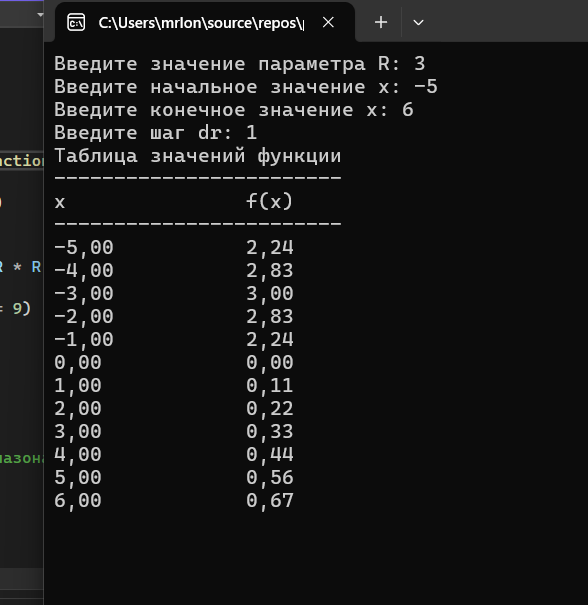
}

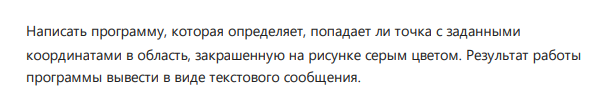
}

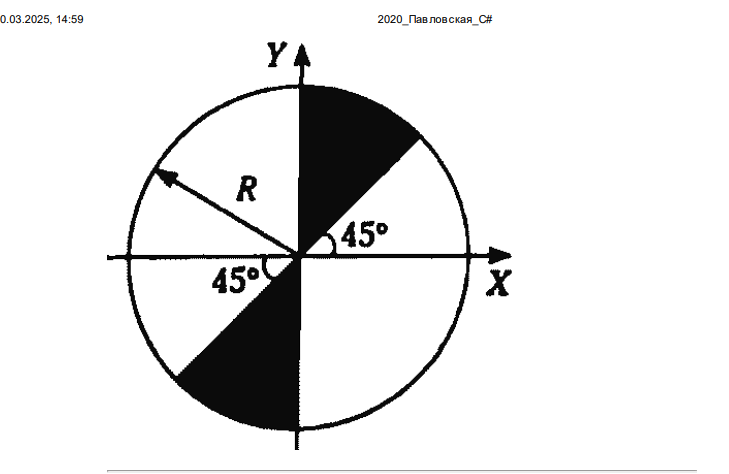
Console.ReadKey(true);

}

}







using System;

class Program

{

static void Main()

{

Console.WriteLine("Программа проверки попадания в мишень (сектор круга 45°)");

Console.Write("Введите радиус мишени R: ");

double R = double.Parse(Console.ReadLine());

// Проверка 10 выстрелов

for (int i = 1; i <= 10; i++)

{

Console.WriteLine($"\nВыстрел #{i}");

Console.Write("Введите координату X: ");

double x = double.Parse(Console.ReadLine());

Console.Write("Введите координату Y: ");

double y = double.Parse(Console.ReadLine());

// Проверка попадания в сектор

bool hit = IsPointInSector(x, y, R, 45);

if (hit)

{

Console.WriteLine("Попадание в мишень!");

}

else

{

Console.WriteLine("Промах!");

}

}

}

// Функция проверки попадания точки в сектор круга

static bool IsPointInSector(double x, double y, double radius, double angleDegrees)

{

// 1. Проверка попадания в круг

double distance = Math.Sqrt(x \* x + y \* y);

if (distance > radius)

{

return false;

}

// 2. Проверка угла (сектор симметричен относительно оси X)

double angleRad = Math.Atan2(y, x);

double angle = angleRad \* (180 / Math.PI); // преобразуем в градусы

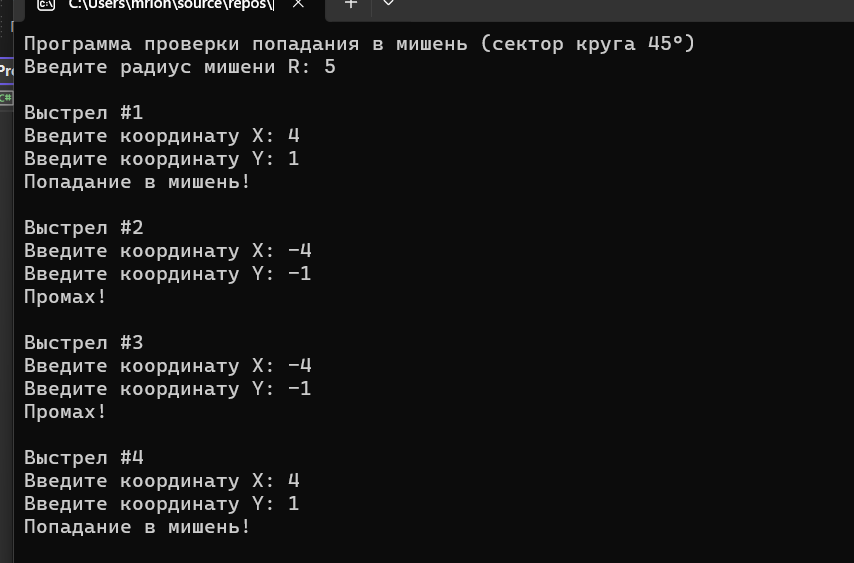
// Нормализуем угол к диапазону [0, 360)

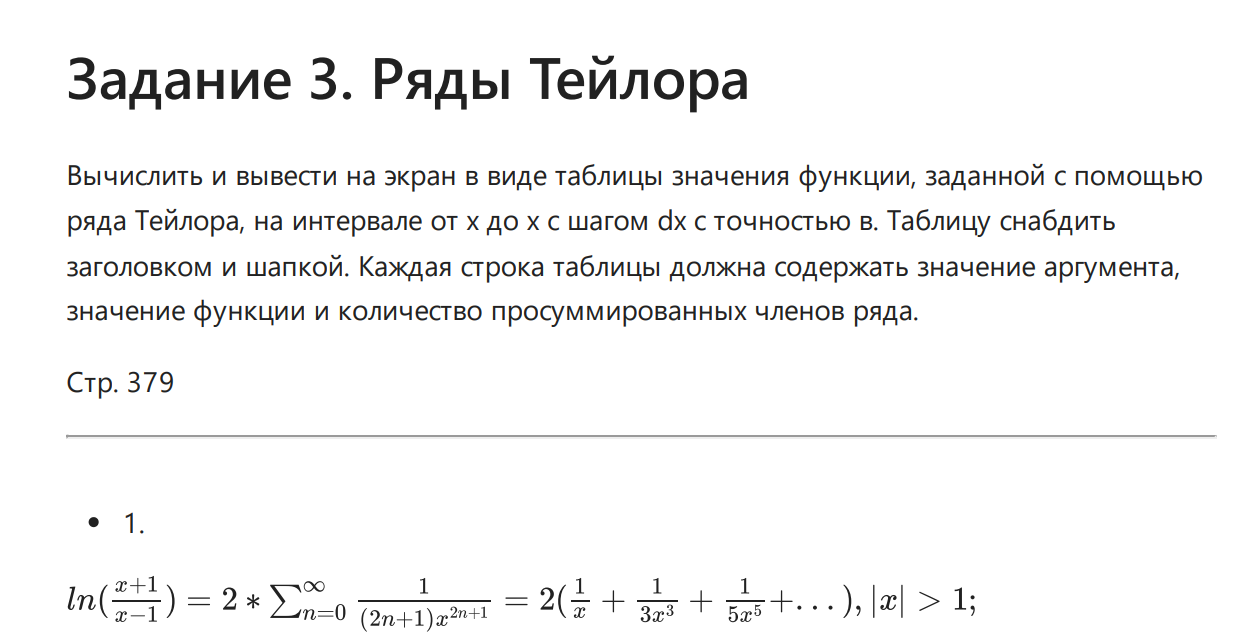
if (angle < 0) angle += 360;

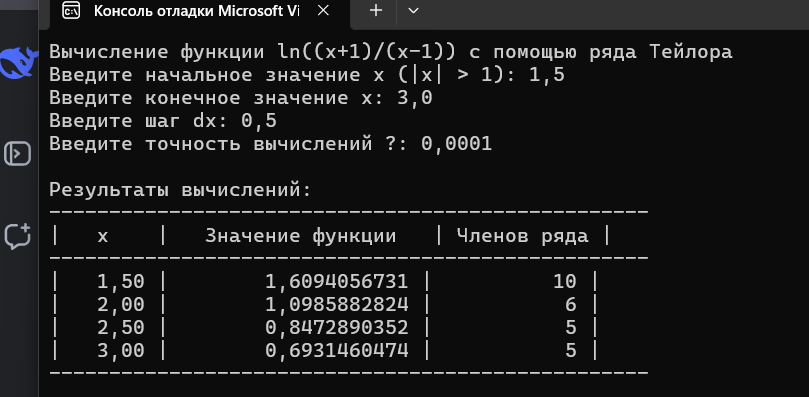
// Проверяем попадание в сектор (два случая из-за симметрии)

return (angle <= angleDegrees / 2) || (angle >= 360 - angleDegrees / 2);

}

} 





using System;

class TaylorSeriesCalculator

{

static void Main()

{

Console.WriteLine("Вычисление функции ln((x+1)/(x-1)) с помощью ряда Тейлора");

// Ввод параметров

Console.Write("Введите начальное значение x (|x| > 1): ");

double xStart = double.Parse(Console.ReadLine());

Console.Write("Введите конечное значение x: ");

double xEnd = double.Parse(Console.ReadLine());

Console.Write("Введите шаг dx: ");

double dx = double.Parse(Console.ReadLine());

Console.Write("Введите точность вычислений ε: ");

double epsilon = double.Parse(Console.ReadLine());

// Проверка корректности ввода

if (Math.Abs(xStart) <= 1 || Math.Abs(xEnd) <= 1)

{

Console.WriteLine("Ошибка: |x| должен быть больше 1");

return;

}

// Вывод шапки таблицы

Console.WriteLine("\nРезультаты вычислений:");

Console.WriteLine("--------------------------------------------------");

Console.WriteLine("| x | Значение функции | Членов ряда |");

Console.WriteLine("--------------------------------------------------");

// Цикл по значениям x

for (double x = xStart; x <= xEnd; x += dx)

{

double sum = 0;

double term;

int n = 0;

// Вычисление суммы ряда

do

{

double denominator = 2 \* n + 1;

double xPower = Math.Pow(x, 2 \* n + 1);

term = 2 / (denominator \* xPower);

sum += term;

n++;

}

while (Math.Abs(term) >= epsilon);

// Вывод строки таблицы

Console.WriteLine($"| {x,6:F2} | {sum,19:F10} | {n,11} |");

}

Console.WriteLine("--------------------------------------------------");

}

}