**2 Классы**

Задание №1. Составить описание класса для вектора, заданного координатами его концов в трехмерном пространстве. Обеспечить операции сложения и вычитания векторов с получением нового вектора (суммы или разности), вычисления скалярного произведения двух векторов, длины вектора, косинуса угла между векторами. Написать программу, демонстрирующую все разработанные элементы класса.

Листинг программы:

using System;

class Vector3D

{

public double X { get; set; }

public double Y { get; set; }

public double Z { get; set; }

public Vector3D(double x, double y, double z)

{

X = x;

Y = y;

Z = z;

}

public static Vector3D operator +(Vector3D v1, Vector3D v2)

{

return new Vector3D(v1.X + v2.X, v1.Y + v2.Y, v1.Z + v2.Z);

}

public static Vector3D operator -(Vector3D v1, Vector3D v2)

{

return new Vector3D(v1.X - v2.X, v1.Y - v2.Y, v1.Z - v2.Z);

}

public static double DotProduct(Vector3D v1, Vector3D v2)

{

return v1.X \* v2.X + v1.Y \* v2.Y + v1.Z \* v2.Z;

}

public double Length()

{

return Math.Sqrt(X \* X + Y \* Y + Z \* Z);

}

public static double CosAngle(Vector3D v1, Vector3D v2)

{

double dot = DotProduct(v1, v2);

double lengths = v1.Length() \* v2.Length();

return lengths != 0 ? dot / lengths : 0;

}

public override string ToString()

{

return $"({X}, {Y}, {Z})";

}

}

class Program

{

static void Main()

{

Vector3D v1 = new Vector3D(3, 4, 5);

Vector3D v2 = new Vector3D(1, 2, 3);

Console.WriteLine($"Вектор 1: {v1}");

Console.WriteLine($"Вектор 2: {v2}");

Vector3D sum = v1 + v2;

Vector3D diff = v1 - v2;

Console.WriteLine($"Сумма: {sum}");

Console.WriteLine($"Разность: {diff}");

Console.WriteLine($"Скалярное произведение: {Vector3D.DotProduct(v1, v2)}");

Console.WriteLine($"Длина вектора 1: {v1.Length():F3}");

Console.WriteLine($"Длина вектора 2: {v2.Length():F3}");

Console.WriteLine($"Косинус угла между векторами: {Vector3D.CosAngle(v1, v2) :F3}");

}

}

Таблица 1.1 – Входные и выходные данные

|  |  |
| --- | --- |
| Входные данные | Выходные данные |
| (3, 4, 5) (1, 2, 3) | (4, 6, 8) (2, 2, 2) 26 7,071 3,742 0,983 |

Анализ результатов:

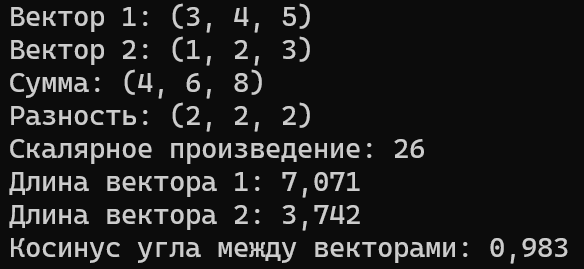


Рисунок 2.1 – Результаты работы программы