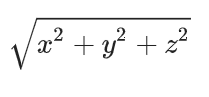
Самостоятельная работа от 12.02.2020

Создайте в Visual Studio проект консольного типа.

Определите в проекте класс, описывающий вектор в трёхмерном пространстве.

Класс должен включать:

* конструктор с параметрами в виде списка координат x, y, z
* метод, вычисляющий длину вектора по формуле:   
  
* метод, вычисляющий скалярное произведение этого вектора с другим по формуле:  
  
* метод, вычисляющий векторное произведение с другим вектором:  
  
* Переопределение метода ToString(), возвращающий строковое представление вектора в формате: vec (x:число ; y:число ; z:число)

В методе Main() создайте два экземпляра класса, описывающего вектор. Используя конструктор задайте для каждого экземпляра, его каждому полю/свойству значения. Используя метод ToString() получите строковое представление каждого объекта. Выведите в консоль полученные строковые представления объектов.  
Рассчитайте модули (длины) векторов и выведите в консоль результат с текстовым сообщением, поясняющим, что это длины векторов.  
Рассчитайте скалярное произведение векторов и выведите в консоль результат с текстовым сообщением, поясняющим, что это результат скалярного произведения векторов.

Рассчитайте векторное произведение векторов и выведите в консоль результат с текстовым сообщением, поясняющим, что это результат векторного произведения векторов.

Листинг класса Program:

static void Main()

{

Vector3D vector01 = new Vector3D(5, 10, 3);

Console.WriteLine(vector01.ToString());

Vector3D vector02 = new Vector3D(15, 2, 25);

Console.WriteLine(vector02.ToString());

Console.WriteLine();

Console.WriteLine($"Длина 1-го вектора: {vector01.Length()}; Длина 2-го вектора: {vector02.Length()}");

Console.WriteLine($"Скалярное произведение векторов: {vector01.ScalarProductVectors(vector02)}");

Console.WriteLine($"Векторное произведение векторов: {vector01.VectorsProduct(vector02)}");

Console.ReadKey();

}

Листинг класса 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 double Length()

{

return Math.Sqrt((Math.Pow(this.X, 2)) + Math.Pow(this.Y, 2) + Math.Pow(this.Z, 2));

}

public double ScalarProductVectors(Vector3D other)

{

return (this.X \* other.X) + (this.Y \* other.Y) + (this.Z \* other.Z);

}

public Vector3D VectorsProduct(Vector3D other)

{

return new Vector3D ( (this.Y \* other.Z) - (this.Z \* other.Y),

(this.Z \* other.X) - (this.X \* other.Z),

(this.X \* other.Y) - (this.Y \* other.X) );

}

public override string ToString()

{

return $"vec (x:{X} ; y:{Y} ; z:{Z})";

}

Результат работы программы:

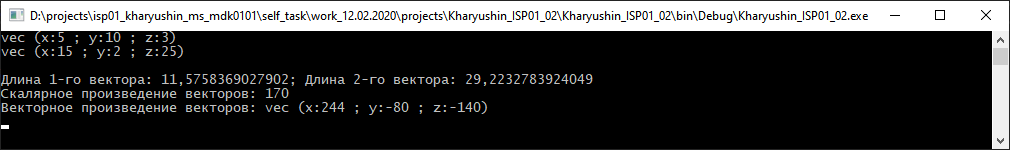


Рисунок 1 - Результат работы программы