“Вектор”

Листинг Vector

using System;

namespace Work2

{

class Vector

{

private double x;

private double y;

private double z;

public double X

{

get { return x; }

set { x = value;}

}

public double Y

{

get { return y; }

set { y = value; }

}

public double Z

{

get { return z; }

set { z = value; }

}

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

{

this.x = x;

this.y = y;

this.z = z;

}

//длинна вектора

public double Length()

{

double resultLen = 0;

resultLen = Math.Sqrt(Math.Pow(x, 2) + Math.Pow(y, 2) + Math.Pow(z, 2));

return resultLen;

}

//скалярное произведение векторов

public double scalarProduct(Vector vec02)

{

double resultеScal = 0;

resultеScal = (x \* vec02.x) + (y \* vec02.y) + (z \* vec02.z);

return resultеScal;

}

//метод вычисляющий векторное произведение с другим

public Vector vectorProduct (Vector vec02)

{

Vector vec = new Vector((y \* vec02.z - z \* vec02.y), (z \* vec02.x - x \* vec02.z), (x \* vec02.y - y \* vec02.x));

return vec;

}

public override string ToString()

{

return $"X:{X}, Y:{Y}, Z:{Z}";

}

}

}

Листинг Program

using System;

namespace Work2

{

class Program

{

static void Main(string[] args)

{

Vector vector1 = new Vector(121, 441, 512);

Vector vector2 = new Vector(11, 41, 52);

Console.WriteLine($"Длина вектора-- ({vector1.Length()})");

Console.WriteLine();

Console.WriteLine($"Скалярное произведение векторов-- ({vector2.Length()})");

Console.WriteLine();

Console.WriteLine($"Векторное произведение с другим вектором-- " +

$"({vector1.vectorProduct(vector2)})");

Console.WriteLine();

Console.WriteLine($"Вектор 1-- ({vector1.ToString()})");

Console.ReadKey();

}

}

}

Снимок результата работы программы

