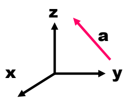
Вектор

Вектор е група от данни, която съдържа алгебричен елемент – дължина (модул) и геометричен елемент – направление.

Координати на вектор – геометричните проекции на вектор върху трите координатни оси.



$a = \{a_x, a_y, a_z\}$

Приложение

- **аритметика с полиноми:**
  изчисляване и интерполация на полином;
  умножение на полиноми;
  аритметични операции с големи числа, представени чрез полином;
- **матрична аритметика.**

Логическо описание

Съединение на елементи от тип РЕАЛНО, наречени координати на вектор.

данни: тип РЕАЛНО

тип ВЕКТОР$_{\text{РЕАЛНО}}$=(координата$_x$; координата$_y$; координата$_z$)

Операции

1. **Създаване на вектор**
2. **Произведение на вектор със скалар**
3. **Събиране на вектори**
4. **Скаларно произведение**
5. **Векторно произведение**
6. **Смесено произведение на три вектора**
7. **Двойно векторно произведение**

Физическо представяне

1. **Непрекъснато представяне**

   – **чрез масив**

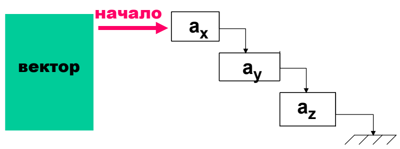
вектор	координата$_z$
	координата$_y$
	координата$_x$

координата$_x$ – **тип** РЕАЛНО
координата$_y$ – **тип** РЕАЛНО
координата$_z$ – **тип** РЕАЛНО

– **чрез структура**

тип ВЕКТОР = (координата$_x$: РЕАЛНО;
координата$_y$: РЕАЛНО;
координата$_z$: РЕАЛНО)

2. Верижно представяне



Дефиниране

```
struct vector                    // вектор
{
   double vx,                    // x координата
          vy,                    // y координата
          vz;                    // z координата
};
typedef struct vector VECTOR;
```

1. Създаване на вектор

Алгоритъм

$$v = (x)i + (y)j + (z)k$$

```
VECTOR create (double x, double y, double z)
{
   VECTOR v;
   v.vx = x;
   v.vy = y;
   v.vz = z;
   return v;
}
```

2. Произведение на вектор със скалар

Алгоритъм

$$v.c = (cv_x)i + (cv_y)j + (cv_z)k$$

```
VECTOR product (VECTOR v, double c)
{
   VECTOR v3;
   v3.vx = c*v.vx;
   v3.vy = c*v.vy;
   v3.vz = c*v.vz;
   return v3;
}
```

3. Събиране на вектори

Алгоритъм

$$v_1 + v_2 = (v_{1x} + v_{2x})i + (v_{1y} + v_{2y})j + (v_{1z} + v_{2z})k$$

```
VECTOR add (VECTOR v1, VECTOR v2)
{
   VECTOR v3;
   v3.vx = v1.vx + v2.vx;
   v3.vy = v1.vy + v2.vy;
   v3.vz = v1.vz + v2.vz;
   return v3;
}
```

4. Скаларно произведение

Алгоритъм

$$v_1 \bullet v_2 = v_{1x}.v_{2x} + v_{1y}.v_{2y} + v_{1z}.v_{2z}$$

```
double dot (VECTOR v1, VECTOR v2)
{
   double d;
   d = v1.vx*v2.vx + v1.vy*v2.vy + v1.vz*v2.vz;
   return d;
}
```

5. Векторно произведение

Алгоритъм

$$v_1 \times v_2 = (v_{1y}v_{2z} - v_{1z}v_{2y})i + (-v_{1x}v_{2z} + v_{1z}v_{2x})j + (v_{1x}v_{2y} - v_{1y}v_{2x})k$$

```
VECTOR cross (VECTOR v1, VECTOR v2)
{
  VECTOR v3;
  v3.vx = v1.vy*v2.vz - v1.vz*v2.vy;
  v3.vy = -v1.vx*v2.vz + v1.vz*v2.vx;
  v3.vz = v1.vx*v2.vy - v1.vy*v2.vx;
  return v3;
}
```

6. Смесено произведение на три вектора

Алгоритъм

$$v_1 \bullet (v_2 \times v_3) = (v_{2y}v_{3z} - v_{2z}v_{3y})v_{1x} + (-v_{2x}v_{3z} + v_{2z}v_{3x})v_{1y} + (v_{2x}v_{3y} - v_{2y}v_{3x})v_{1z}$$

```
double mixedProduct (VECTOR v1, VECTOR v2, VECTOR v3)
{
  VECTOR vv;
  double d;
  vv = cross (v2, v3);
  d = dot (v1, vv);
  return d;
}
```

7. Двойно векторно произведение на три вектора

Алгоритъм

$$v_1 \times (v_2 \times v_3) = (v_1 \bullet v_3) \bullet v_2 - (v_1 \bullet v_2) v_3$$

```
VECTOR doubleCross(VECTOR v1,VECTOR v2,VECTOR v3)
{
  VECTOR vv1, vv2;
  vv1 = cross (v2, v3);
  vv2 = cross (v1, vv1);
  return vv2;
}
```

Задача: **Структура данни** ВЕКТОР

1. **Напишете функции** X(), Y() **и** Z()**, които осигуряват достъп съответно до x-координата, y-координата и z-координата на даден вектор.**
2. **Напишете функция** divide() **за деление на вектор със скалар.**

$$v.c = (v_x / c)i + (v_y / c)j + (v_z / c)k$$

3. **Напишете функция** length() **за изчисляване дължина (модул) на вектор и функция** angle() **за изчисляване на ъгъл между два вектора.**

$$l = \sqrt{v_x^2 + v_y^2 + v_z^2}$$

$$\cos(v_1, v_2) = \frac{v_{1x}v_{2x} + v_{1y}v_{2y} + v_{1z}v_{2z}}{\sqrt{v_{1x}^2 + v_{1y}^2 + v_{1z}^2}\sqrt{v_{2x}^2 + v_{2y}^2 + v_{2z}^2}}$$

Задача: **Представете структура данни** ВЕКТОР **чрез масив. Напишете функция** add()**, която изчислява сумата на два полинома.**

$$r(x) = p(x) + q(x)$$

$$p(x) = p_o + p_1 x + \ldots + p_{N-1} x^{N-1}$$

$$q(x) = q_o + q_1 x + \ldots + q_{N-1} x^{N-1}$$

$$r(x) = (p_0 + q_0) + (p_1 + q_1)x + \ldots + (p_{N-1} + q_{N-1})x^{N-1}$$