

## Lista 7 – Vetores

- 1) Crie um algoritmo que leia 10 números e armazene-os, depois apresente todos os números e o seu dobro. O dobro também deve estar armazenado em um vetor.
- 2) Crie um algoritmo que leia 20 números, armazene-os e apresente ao final:
  - a. O maior número digitado;
  - b. A quantidade de números negativos;
- 3) Crie um algoritmo que calcule os 10 primeiros termos da sequência de Fibonacci sem a utilização de vetor, e depois crie uma versão com a utilização de vetor. Sequência: 1, 1, 2, 3, 5, 8, 13, 21, 34, 55
- 4) Realize o teste de mesa dos algoritmos a seguir, e exiba o que será armazenado na memória e o que será escrito na tela.

### ALGORITMO Vetor1

```
VAR
    i, x, y, num[10]: INTEIRO;
INICIO
    x ← 1;
    y ← 2;
    num[x] ← y * y;
    num[y] ← x;
    num[x + y] ← num[x];
    num[6] ← num[1] + num[3];
    num[num[x]] ← num[6] - num[y];
    num[x + y] ← x * y;
    num[3 + 4] ← num[6] * 2;
    num[x - 1] ← x + y;
    num[num[x] + 1] ← num[x] + num[y];
    num[num[6] + 1] ← num[6 + 1];
    num[4 * 2] ← num[9] - num[7] + x;
    PARA (i ← 0; i < 10; i ← i + 1) FAÇA
        ESCREVA (num[i]);
    FIMPARA;
```

**FIM**

### ALGORITMO Vetor2

```
VAR
    i, x, y, num[10] = {3,4,6,2,1,6,8,2,9,5}: INTEIRO;
INICIO
    x ← 2;
    y ← 3;
    num[x] ← num[x * y];
    num[num[x]] ← num[x + y];
    num[y + 1] ← x + y;
    num[y - x] ← num[num[y]];
    num[y] ← num[y] + num[x];
    num[x - 2] ← num[y] + x;
    PARA (i ← 0; i < 10; i ← i + 1) FAÇA
        ESCREVA (num[i]);
```

FIM\_PARA

**Fim.**