

→ O tipo estruturado: Conjunto (SET)

O tipo de dados estruturado SET (conjunto) define um conjunto de elementos que é manipulado como um todo.

Suponha as seguintes declarações:

```

...
Type
    bola = 1..90;
    saco = set of bola;
Var
    bingo : saco;
...

```

Para atribuir valores a uma variável do tipo conjunto, são válidos os comandos:

```

bingo:= [ ];      {o valor da variável é o conjunto vazio}
bingo:= [1..90];  { os valores do conjunto são definidos por intervalo}
bingo:= [1,2,3];  { os valores do conjunto são definidos por enumeração}
bingo:= [1,5, 10..20, 50];

```

Supondo que **a** e **b** são variáveis do tipo **SET of T** e **x** uma variável do tipo **T**, são válidos os seguintes operadores:

```

+ reunião { a+b contém todos os elementos de a e/ou b }
- diferença { a-b contém os elementos de a que não pertencem a b }
* intersecção { a*b contém os elementos que pertencem a a e a b }
= igualdade { a=b é TRUE se a e b contêm os mesmos elementos }
<> desigualdade { a<>b é TRUE se a=b é FALSE }
<=          { a<=b é TRUE se a está contido em b }
>=          { a>=b é TRUE se b está contido em a }
in        { x in b é TRUE se o elemento x pertence a b }

```

1 - Considerando os seguintes conjuntos:

$$X:=[1]; \quad Y:=[1,2,3]; \quad Z:=[4,6,8,9]; \quad U:=[8,9]; \quad V:=[\];$$

- Qual o resultado das operações que se seguem?

- a) $X+Y$
- b) $X-Y$
- c) $(X+Y)-U$
- d) $(Z*Y)+X$
- e) $2 \text{ in } Y$
- f) $U \leq Z$

2- Construir um programa que leia uma linha de caracteres para um array e conte o número de vogais existentes nessa linha. Suponha que a linha tem no máximo 80 caracteres e que termina com um ponto final. Deverá utilizar as seguintes declarações de variáveis, e instrução de atribuição:

Type

conjunto = set of char;

Var

linha : array [1..80] of char;

vogais: conjunto;

...

Begin

vogais:= ['a','e','i','o','u','A','E','I','O','U']

...

→ O tipo estruturado: Registo (RECORD)

Uma variável do tipo RECORD (registo) é constituída por um conjunto fixo de elementos (designados por campos). Os vários campos podem ter diferentes tipos e cada campo tem um nome próprio (identificador do campo).

Exemplo:

```

...
Type
    complexo=record
        re:real;
        im:real
    end;
var
    x,y:complexo;
begin
    x.re:=1;
    x.im:=0;
    y.re:=0;
    y.im:=1;
    writeln (" Complexo x: ", x.re, x.im);
    writeln (" Complexo y: ", y.re, y.im);
end.

```

3- Considerando o tipo Complexo definido no exemplo acima construa procedimentos que calculem:

- i) A soma de dois complexos
- ii) O produto de dois complexos
- iii) A divisão de dois complexos
- iv) O conjugado de um complexo

- Implementar um programa que permita testar os quatro procedimentos anteriores.