

Slide 1

TEMA 11: TIPOS ESTRUCTURADOS: CADENA. CONJUNTO.

Índice

11.1. Cadenas de caracteres	1
11.2. Tipo “String”	9
11.3. Conjuntos	13

Slide 2

TIPO CADENA DE CARACTERES

valores: cadenas de caracteres, con tamaño dinámico

operaciones: ■ acceso a cada carácter individual

- asignación “cómoda”
- comparación (orden lexicográfico)
- concatenación
- lectura y escritura “cómodas”
- paso como parámetro
- extracción, inserción y borrado de subcadenas
- ...

Slide 3

TIPO CADENA DE CARACTERES: IMPLANTACIÓN

cadena de caracteres = vector de caracteres + tamaño efectivo

- vector de caracteres de tamaño que prevea la longitud máxima posible
- tamaño efectivo en alguna otra variable, para modificación dinámicamente:
 - para cada variable de tipo cadena otra variable con el tamaño efectivo
 - registro con dos campos: uno para el tamaño efectivo (entero) y otro para vector de caracteres
- o bien: carácter especial de fin efectivo de la cadena.
- procedimientos asociados que actualicen el vector y su tamaño efectivo

Slide 4

```
const LONGMAX=255;
type tcadena = record
    long : integer;
    valor : array [1..LONGMAX] of char;
end;

function longitud (x:tcadena): integer;
begin
    longitud := x.long
end;

procedure aVacia (var x: tcadena);
begin
    x.long := 0;
end;
```

Slide 5

```
procedure asigna (x: tcadena; var y:tcadena);
(* y := x *)
var i : integer;
begin
    for i := 1 to x.long do
        y.valor[i]:=x.valor[i];
    y.long := x.long
end;
procedure leecadena (var x:tcadena);
...
procedure leecadenaln (var x:tcadena);
...
function anterior (x, y : tcadena): boolean;
...
```

Slide 6

```
const LONGMAX=25; EOC=chr(0);
type tcadena = array [1..LONGMAX+1] of char;

procedure aVacia (var x : tcadena);
begin
    x[1] := EOC;
end;

function longitud (x:tcadena): integer;
(* Pre: EOC aparece en x *)
var i: integer;
begin
    i := 1;
    while x[i]<> EOC do i := i+1;
    longitud := i-1
end;
```

Slide 7

```

procedure asigna (x:tcadena; var y : tcadena);
var i, l: integer;
begin
    l := longitud(x);
    for i := 1 to l do
        y[i]:=x[i];
    y[l+1] := EOC
end;
procedure leecadena (var x:tcadena);
...
procedure leecadenaln (var x:tcadena);
...
function anterior (x, y : tcadena): boolean;
...

```

Slide 8

TIPO CADENA: **string** ó **string** [*< exp cte>*] (no estándar)

valores: cadenas de caracteres, con tamaño dinámico

operaciones:

- acceso a cada carácter individual como vector
- asignación **a := b**
- relacionales (orden lexicográfico) **<, <=, >, >=, = <>**
- concatenación: operador **+**
- **read, readln, write, writeln**
- paso como parámetro e incluso como valor devuelto

	length		delete
funciones:	concat	procedimientos:	insert
	copy		str
	pos		val
			...

Slide 9

longitud:	length (st)
concatenación:	concat (s1, s2, s3, ...)
extracción de subcadena:	copy (s, p, n)
posición de subcadena:	pos (ss,s)
borrado:	delete (s, p, n)
inserción:	insert (ss, s, p)
conversión número a cadena:	str (x, s) ó str (x:m:n, s)
conversión cadena a número:	val (s, v, c)
...	

Slide 10

TIPO CONJUNTO: **set of** *<tipo ordinal>*

valores: subconjuntos posibles del tipo ordinal base (conjunto universal)

constantes: [*<elemento>* {, *<elemento>* }] En particular, conjunto vacío: []

operaciones:

- asignación **a := b**
- 4 relacionales (orden de inclusión) **<= >= = <>**
- pertenencia de un elemento al conjunto: **in**
- unión : operador **+**
- intersección : operador *****
- diferencia: operador **-**
- paso como parámetro e incluso como valor devuelto