

Módulo Imperativo

Recursión.

Ejercicio Resuelto de Repaso

Autores

Leandro Romanut

Alejandro Héctor González

Resumen

En esta clase se trabaja con un enunciado de sobre la lectura de una cadena de caracteres. El mismo tiene naturaleza recursiva y se propone una posible solución en Pascal

Palabras clave

caracter, lectura, recursión, calculo

Ejercicio

Se lee una palabra de no más de 50 caracteres.
Determinar si la misma es palíndromo (se lee del mismo modo de izquierda a derecha y viceversa).

La lectura de la palabra finaliza cuando llega el carácter blanco o se alcance su longitud máxima (50).

Analizando el problema

¿En qué estructura almacenaremos la palabra?

R	E	C	O	N	O	C	E	R						...														
---	---	---	---	---	---	---	---	---	--	--	--	--	--	-----	--	--	--	--	--	--	--	--	--	--	--	--	--	--

¿Qué información necesitaremos de la palabra?

- Longitud = dimensión lógica

Analizando el problema

¿Cómo determinar si es un palíndromo?

NEUQUEN
↑ ↑

EUQUE
↑ ↑

UQU
↑ ↑

Q
↑↑

ERRE
↑ ↑

RR
↑ ↑

REIR
↑ ↑

EI
↑ ↑

Comparamos el primer y último carácter.
Luego, "reducimos" el problema.

¿Cuáles serían los casos base?

Solución

```
program ejercicio;
```

```
CONST
```

```
    long_max = 50;
```

```
TYPE
```

```
    letras = Array[1..long_max] of char;
```

```
    palabra = record
```

```
        p : letras;
```

```
        long : integer;
```

```
    end;
```

```
{Proceso que carga la palabra en el vector}
```

```
procedure cargarPalabra (var pal: palabra);
```

```
var
```

```
    car: char;
```

```
begin
```

```
    pal.long := 0;
```

```
    read(car);
```

```
    while (car <> ' ') and (pal.long < long_max) do
```

```
    begin
```

```
        pal.long := pal.long + 1;
```

```
        pal.p[pal.long] := car;
```

```
        read(car);
```

```
    end;
```

```
end;
```

```
{VARIABLES DEL PROGRAMA PRINCIPAL}
```

```
var
```

```
    pal: palabra;
```

```
begin {PROGRAMA PRINCIPAL}
```

```
    cargarPalabra(Pal);
```

```
    if (palindromo(pal)) then
```

```
        writeln ('---- Es palindromo ----')
```

```
    else
```

```
        writeln ('---- NO es palindromo ----');
```

```
end.
```


Solución

{FUNCIONES Y PROCEDIMIENTOS}

```
function verificarPalindromo (pal: letras; ini, fin: integer):boolean;  
Begin  
    if (ini = fin) or (fin < ini) then  
        //hay un solo caracter o se cruzan los índices  
        verificarPalindromo := true  
    else  
        if (pal[ini] <> pal[fin]) then  
            //deja de ser palíndromo porque las letras no coinciden  
            verificarPalindromo := false;  
        else begin  
            //llamado recursivo  
            ini := ini + 1;  
            fin := fin - 1;  
            verificarPalindromo := verificarPalindromo (pal, ini, fin);  
        end  
    end;  
end;
```

{Inicializa los límites e invoca a la función verificarPalindromo}

```
function palindromo (pal: palabra): boolean;  
var  
    ini, fin: integer;  
begin  
    ini := 1;  
    fin := pal.long;  
    palindromo := verificarPalindromo (pal.p, ini, fin);  
end;
```