

# Lliçons de TurboPascal.

Per David Obrador Sala

Font:        - Manual de Nacho Cabanes  
              - Pascal y TurboPascal I. J.L. Isabel Fernández

## Paraules claus

BEGIN                    (no distingim majúscules ni minúscules)

Atenció cal emprar el punt i coma ; a final de cada línia.

Amb certes excepcions

END.

Exemple de programa:

---

```
PROGRAM HOLA MUNDO  
  
    BEGIN  
  
        WRITE (hola mundo);  
  
    END
```

---

Què cal fer ara?

1. Menú FILE i SAVE AS i guardar l'arxiu en una carpeta personal (es crearan arxius amb extensions PAS i EXE que convé controlar d'ubicació)
2. Menú RUN i opció RUN (o CTRL + F9) i anar traient conclusions.

3. L'etiqueta d'encapçalament és un nom sense espais ni caràcters especials.  
Ha d'acabar en ;
4. Dins del WRITE hi van cometes simples
5. Cal END.
6. Amb ALT + F5 canviem a la pantalla de resultats
7. Arreglar el programa per tal que funcioni

Programa arreglat:

---

```
PROGRAM HOLA_MUNDO;  
  
    BEGIN  
  
        WRITE ('hola mundo');  
  
    END.
```

---

**Variables**

Poden prendre valors numèrics, text, ...

Nom d'una variable:

text, combinacions de text i caràcters i numèrics.

| <b>Exemples vàlids</b> | <b>Exemples no vàlids</b> |
|------------------------|---------------------------|
| Nom_De_Programa        | 1Programa, 1_Programa     |
| Programa_2             | 2ºPrograma                |
| Exemple_3              | exemple 3 de TP           |
| Practica_4             | Pràctica_4                |

## Instrucció Write

Permet escriure en pantalla, cal emprar la sintaxi

```
WRITE (      );
```

### Observacions:

El programa HOLA\_MUNDO també pot ser:

---

```
PROGRAM HOLA_MUNDO;

    BEGIN

        WRITE ('hola mundo')
        (*AQUÍ comentaris*)

        { WRITE sense ; punt i coma al
          final }

    END.
```

Però hem d'anar en compte perquè si volem afegir una comanda, tenim:

---

```
PROGRAM HOLA_MUNDO;

    BEGIN

        WRITE ('hola mundo')

        WRITE ('com anem?')

        (*el primer WRITE sense ; al final
          ens donarà errors*)

    END.
```

---

### Observacions:

1. Com introduir comentaris en un arxiu de TurboPascal.

2. Quins canvis tinc en els arxius \*.EXE tot i no guardar canvis ens els arxius \*.PAS?
3. Escriptura INDENTADA.
4. Menú HELP per obtenir ajut sobre funcions.
5. Buscar Informació amb el menú HELP sobre quins tipus d'enters pot gestionar TurboPascal (en anglès INTEGER).

Ara veurem un programa que permet fer canvis de valors en una variable, proveu-lo:

---

```
PROGRAM NOU_VALOR;

VAR

    NUMERO: INTEGER;

BEGIN

    NUMERO:=25;

    WRITELN('LA VARIABLE VAL', NUMERO);

    NUMERO:=50;

    WRITELN('ARA EN CANVI VAL', NUMERO);

    NUMERO:= NUMERO + 10;

    WRITELN(' I ARA VAL', NUMERO);

    WRITELN('ENTRA ARA TU MATEIX EL VALOR',NUMERO);

    READLN( NUMERO);

    WRITELN('I PER ACABAR ARA VAL', NUMERO);

END.
```

---

### Observacions:

1. Quin és el funcionament de les noves funcions WRITELN i READLN?

2. Què vol dir definir una variable de tipus INTEGER? Com es defineixen variables (abans o després del BEGIN?, amb " : " o bé amb " := "?)
3. Com es modifica el valor d'una variable?
4. Quines diferències hi ha entre les funcions WRITE i WRITELN? (pots emprar el menú HELP)

## Format d'entrada i sortida de dades

En aquesta part veurem amb un exemple com podem treure valors en pantalla controlant quantes xifres volem i quants decimals.

---

```
PROGRAM WRITE2;

VAR

    NUM: REAL;

BEGIN

    NUM:= 1234567.89;

    WRITELN(NUM); (* s'escriu en el format per
defecte que és exponencial *)

    WRITELN(NUM:20:3);      (* amb 3 decimals *)
    WRITELN(NUM:7:2);      (* amb *** decimals *)
    WRITELN(NUM:4:1);      (* amb *** decimals *)
    WRITELN(NUM:3:0);      (* amb *** decimals *)
    WRITELN(NUM:5);        (* amb *** decimals *)

END.
```

---

### Observacions:

1. Quins són els resultats a la pantalla de resultats?
2. Confirma si podem extreure les següents conclusions:
  - En no indicar cap format, s'empra l'exponencial.
  - Si sobra amplada s'omple amb espais per l'esquerra.
  - Si és menor, no es "talla" el nombre.
  - Si el nombre de decimals és major, què passa?
  - Si el nombre de decimals és menor, s'arrodoneix?
  - Si indiquem format però no la quantitat de decimals què fa?

## Sentència IF

Hi ha una sentència que determina l'execució d'una sèrie d'instruccions o d'unes altres segons el resultat d'una o diverses comparacions. És la sentència **IF**.

El format general d'utilització de la sentència if és el següent:

```
if <expressió lògica>  
then <sentència 1>  
else <sentència 2>;
```

Quan el resultat de l'expressió lògica sigui cert, s'executarà la sentència1, en canvi quan sigui fals s'executarà la sentència 2.

Recordeu amb l'EXCEL les instruccions

```
IF( a2>=5 ; "aprovat" ; "suspès" )
```

O bé un exemple amb TurboPascal:

```
write ( ' Introdueix la nota de l'examen: ' );  
readln(nota);  
  
if ( nota <0 ) or ( nota >10 )  
then writeln ( 'Nota incorrecta' )  
else writeln ( 'Nota correcta' );
```

### Observacions:

1. Completa el text anterior per tal que funcioni el programa, és a dir, fes-ne



la capçalera, declara les variables que calguin,...

2. La sentència `if ** then ** else` és una de les excepcions de la utilització dels punts i comes a final de línia.

3. Es poden "encaixar" diferents sentències `if ** then ** else`. Completa el següent exemple:

---

```
write ( ' Introdueix la nota de l'examen: ');
readln(nota);
if ( nota <0 ) or ( nota >10 )
then
    begin
        writeln ( 'Nota incorrecta');
        writeln ( 'ha de ser entre 0 i 10');
    end
else
    begin
        writeln ( 'Nota correcta');
        if (nota >=5 )
        then writeln ( 'aprovat')
        else writeln ('suspès');
    end;
end;
```

---

**Sentència for \*\* to \*\* do**

La sentència `for ** to ** do` s'utilitza per a repetir una mateixa acció un nombre determinat de vegades. Cerca al menú HELP el seu funcionament i realitza el programa de càlcul del factorial d'un nombre enter.

**Altres sentències, instrucció case**

---

```
program case1;
var
    lletra: char;
begin
    WriteLn('Escriu un símbol');
    ReadLn(lletra);
    case lletra of
        ' ':           WriteLn('Un espai');
        'A'..'Z', 'a'..'z': WriteLn('Una lletra');
        '0'..'9':       WriteLn('Un dígit');
        '+', '-', '*', '/': WriteLn('Un operador');
    else
        WriteLn('No es espai, ni lletra, ni dígit, ni
operador');
    end;
end.
```

---

**Programes per anar fent proves**

**Prog 1.** Programa que demani a l'usuari el nom i l'edat i respongui

```
"Benvolgut senyor ***** vostè té ** anys".
```

**Prog 2.** Programa per a calcular àrees de triangles. El programa demanarà la base i l'altura i les unitats de mesura.

**Prog 3.** Programa per a calcular longituds de circumferències. El programa demanarà el radi i les unitats de mesura.

**Prog 4.** Programa per a calcular àrees de cercles. El programa demanarà el radi i les unitats de mesura.

**Prog 5.** Programa que demani a l'usuari un nombre enter i en retorni el seu quadrat, el seu cub i la seva quarta potència.

**Prog 6.** Programa que demani a l'usuari dos nombres i en retorni la seva suma, el seu producte, el seu quocient (el més gran entre el més petit), i el residu del seu quocient.

**Prog 7.** Programa per a calcular arrels quadrades de nombres. El programa demanarà el nombre i respondrà:

" L'arrel del nombre \*\*\*\*\* és \*\*\*\*\*"

**Prog 8.** Programa que demani a l'usuari un nombre enter i respongui si és parell o senar (en anglès **even** i **odd**)

**Prog 9.** Programa per a resoldre equacions de segon grau. El programa demanarà els coeficients de l'equació i en donarà les possibles solucions.

**Prog 10.** Programa per a calcular el factorial d'un nombre.

Per exemple  $7! = 7*6*5*4*3*2*1 = 5040$ .

**Prog 11.** Programa per a calcular la suma de nombres enters consecutius. El programa ha de demanar a l'usuari fins a quin nombre enter cal fer la suma. Per exemple:

fins a 5, aleshores  $\text{Suma} = 1 + 2 + 3 + 4 + 5 = 15$ .

**Prog 12.** Programa per a calcular la suma de nombres enters senars consecutius. El programa ha de demanar a l'usuari fins a quin nombre enter cal fer la suma. Per exemple:

fins a 5, aleshores  $\text{SumaSenars} = 1 + 3 + 5 = 9$ .

**Prog 13.** Programa per a calcular la suma dels quadrats de nombres enters consecutius. El programa ha de demanar a l'usuari fins a quin nombre enter cal fer la suma. Per exemple:

fins a 5, aleshores

$$\text{SumaQuadrats} = 1^2 + 2^2 + 3^2 + 4^2 + 5^2 = \dots$$

**Prog 14.** Programa de conversió de graus Celsius a graus Kelvin.

**Prog 15.** Programa de conversió de mesures angulars en radians a graus.

**Prog 16.** Programa d'endevinar un nombre aleatori entre el 0 i el 100 en menys de 10 intents.

**Prog 17.** Càlcul del terme n-èssim de la successió de Fibonacci. El programa demanarà fins a quin nombre cal arribar

$$a_1 = 1, a_2 = 1, a_{n+1} = a_n + a_{n-1}$$

i obtenim: 1 , 1 , 2 , 3 , 5 , 8 , 13 , 21 , ...

si  $n = 5$ , aleshores  $a_5 = 5$

si  $n = 6$ , aleshores  $a_6 = 8$

si  $n = 9$ , aleshores  $a_9 = \dots$

si  $n = 10$ , aleshores  $a_{10} = \dots$

**taula de continguts**

|                                                 |           |
|-------------------------------------------------|-----------|
| <b>Paraules claus .....</b>                     | <b>1</b>  |
| <b>Variables .....</b>                          | <b>3</b>  |
| <b>Instrucció Write.....</b>                    | <b>4</b>  |
| <b>Format d'entrada i sortida de dades.....</b> | <b>7</b>  |
| <b>Sentència IF .....</b>                       | <b>8</b>  |
| <b>Sentència     for ** to ** do .....</b>      | <b>10</b> |
| <b>Altres sentències, instrucció case.....</b>  | <b>10</b> |
| <b>Programes per anar fent proves.....</b>      | <b>11</b> |
| <b>taula de continguts.....</b>                 | <b>14</b> |