

**Nombre y Apellidos:**  
**Especialidad y Grupo:**

## FUNDAMENTOS DE INFORMÁTICA

15-septiembre-2003

### PROGRAMA de Estructuras repetitivas y alternativas (2 puntos)

1. Dado el módulo EjForAni.bas, modificarlo para visualizar en pantalla una serie de números, utilizando escritura con formato (#) y tecleando previamente un número de línea cualquiera [1,17]. Si por ejemplo la línea introducida es 5, la salida por pantalla será la siguiente:

```
col
0      1      2      3      4      5      6      7
123456789012345678901234567890123456789012345678901234567890
0                                           = 0
0 3                                           = 3
0 3 6                                           = 9
0 3 6 9                                           = 18
0 3 6 9 12                                           = 30
```

```
DIM f AS INTEGER, c AS INTEGER
DIM numLineas AS INTEGER
DIM num AS INTEGER
INPUT "Teclear el número de líneas: ", numLineas
IF numLineas >= 5 AND numLineas <= 10 THEN
  FOR f = 1 TO numLineas
    FOR c = 1 TO numLineas
      PRINT num,
      num = num + 2
    NEXT c
  NEXT f
ELSE
  PRINT "ERROR en el número de líneas"
END IF
```

```
DIM f AS INTEGER, c AS INTEGER
DIM numLineas AS INTEGER
DIM num AS INTEGER
INPUT "Teclee el número de líneas [1,17]: ", numLineas
IF numLineas >= 1 AND numLineas <= 17 THEN
  FOR f = 1 TO numLineas
    num = 0
    suma = 0
    FOR c = 1 TO f
      PRINT USING "####"; num;
      suma = suma + num
      num = num + 3
    NEXT c
    PRINT TAB(70); "=";
    PRINT USING "#####"; suma
  NEXT f
ELSE
  PRINT "ERROR en el número de líneas"
END IF
```

**PROGRAMA de Arrays y Registros (3 puntos)**

2. Codificar un programa en QB para obtener el camino que sigue un móvil sobre un tablero según las indicaciones de la ruta. El móvil se desplaza por el tablero saltando de una posición a otra contigua en dirección Norte, Sur, Este u Oeste. Para ello, se dispone del siguiente array:

- Array bidimensional estático, llamado *arrayAlfa*, de  $20 \times 20 = 400$  elementos de tipo alfanumérico. Cada elemento contiene 1 único carácter.

Para la obtención del camino del móvil se deben seguir estos pasos:

- Declarar el array bidimensional *arrayAlfa*.
- Cargar los elementos en dicho array con una letra cualquiera, pero de la siguiente forma:
  - Declarar e inicializar una cadena alfanumérica con el siguiente contenido:  
cadena = "0123456789ZYXWVUTSRQPOÑNMLKJIHGFEDCBA".
  - Generar un número al azar entre [1,37].
  - Este número será la posición que ocupa la letra en la cadena anterior. Dicha letra se cargará en la fila y en la columna del array de forma consecutiva.
- Introducir los datos de la ruta a seguir según el siguiente formato de pantalla:  
Fila inicio de ruta: \_\_\_  
Colu inicio de ruta: \_\_\_  
Ruta: \_\_\_\_\_

Los datos anteriores pueden ser incorrectos. Por ello, controlar que:

- La fila y la columna de inicio de ruta sean valores pertenecientes a [1,20].
- La ruta sólo contenga caracteres del conjunto {N,S,E,O}, que indican los puntos cardinales.
- Repetir la entrada de dichos datos hasta que sean correctos, visualizando el mensaje de error que corresponda en cada caso.
- Visualizar el camino del móvil, empezando por la fila y columna de inicio y siguiendo el desplazamiento según los puntos cardinales: N (arriba), S (abajo), E (derecha) y O (izquierda).
- Por ejemplo, si fila = 3, columna = 7 y ruta = {SENNOOSSE}:

*arrayLetras* 10 filas x 10 columnas

A	B	C	D	2	Q	W	E	R	U
B	W	E	S	3	Y	T	E	U	W
Q	E	5	R	T	Y	Y	Y	U	I
Y	R	R	B	A	H	R	9	Z	O
Y	T	I	R	A	A	S	D	G	H
E	R	T	Y	U	Q	Y	U	I	O
Q	S	D	F	G	J	K	4	K	I
X	V	B	6	7	J	K	N	O	0
H	J	R	E	E	4	G	E	S	2
1	2	3	6	7	8	9	T	0	R

Camino del móvil = YR9YETY3TAH

```
DIM arrayAlfa(1 TO 20, 1 TO 20) AS STRING * 1
```

```

DIM i AS INTEGER, j AS INTEGER
DIM num AS INTEGER
DIM cadena AS STRING * 37
cadena = "0123456789ZYXVWUTSRQPONMLKJIHGFEDCBA"
RANDOMIZE TIMER
FOR i = 1 TO 20
    FOR j = 1 TO 20
        num = INT(RND * 37) + 1
        arrayAlfa(i, j) = MID$(cadena, num, 1)
    NEXT j
NEXT i

CLS
FOR i = 1 TO 20
    FOR j = 1 TO 20
        PRINT arrayAlfa(i, j);
    NEXT j
    PRINT
NEXT i
PRINT

DIM fila AS INTEGER, colu AS INTEGER
DO
    INPUT "Fila inicio de ruta: ", fila
    INPUT "Colu inicio de ruta: ", colu
    IF fila < 1 OR fila > 20 OR colu < 1 OR colu > 20 THEN
        PRINT "Fila y Columna [1,20]. Repetir"
    END IF
LOOP UNTIL fila >= 1 AND fila <= 20 AND colu >= 1 AND colu <= 20
DIM ruta AS STRING
DIM letra AS STRING * 1
DIM erro AS STRING * 2
DO
    INPUT "Ruta: ", ruta
    ruta = UCASE$(ruta)
    erro = "NO"
    FOR i = 1 TO LEN(ruta)
        letra = MID$(ruta, i, 1)
        IF INSTR("NSEO", letra) = 0 THEN
            erro = "SI"
        END IF
    NEXT i
    IF erro = "SI" THEN
        PRINT "Ruta {N,S,E,O}. Repetir"
    END IF
LOOP UNTIL erro = "NO"

DIM punto AS STRING * 1
PRINT "Camino del móvil = "; arrayAlfa(fila, colu);
FOR i = 1 TO LEN(ruta)
    punto = MID$(ruta, i, 1)
    SELECT CASE punto
        CASE IS = "N"
            fila = fila - 1
        CASE IS = "S"
            fila = fila + 1
        CASE IS = "E"
            colu = colu + 1
        CASE IS = "O"
            colu = colu - 1
    END SELECT
    PRINT arrayAlfa(fila, colu);
NEXT i

```

### PROGRAMA de Funciones y Procedimientos (2.5 puntos)

3. Calcular el dígito de control de una cuenta bancaria. Para ello:

- Programa Principal
  - Teclear un número de cuenta bancaria de 8 dígitos en una variable STRING. Se supone que la longitud del dato es correcta.
  - Llamar al procedimiento *calcularDigitoControl* con el fin de que devuelva el dígito de control de la cuenta.
  - Visualizar en pantalla el número de cuenta y su dígito. Por ejemplo:  
12345678      4
- Procedimiento *calcularDigitoControl*
  - El parámetro formal de entrada es el número de cuenta y el de salida es su dígito de control.
  - Para obtener el dígito de control vamos a utilizar el siguiente ejemplo. Si el número de cuenta es 12345678, su dígito de control es 4.  
 $8 \times 1 + 7 \times 3 + 6 \times 5 + 5 \times 7 + 4 \times 9 + 3 \times 11 + 2 \times 13 + 1 \times 15 = 204$   
resto de la división de 204 entre 10 es 4.

```
REM Visualizar el dígito de control de  
REM una cuenta bancaria.  
REM
```

```
DECLARE SUB calcularDigitoControl (cuenta AS STRING, digito AS  
INTEGER)
```

```
CLS  
DIM cuentaBanco AS STRING * 8  
DIM digitoControl AS INTEGER
```

```
INPUT "Teclear el número de cuenta: ", cuentaBanco  
CALL calcularDigitoControl(cuentaBanco, digitoControl)  
PRINT  
PRINT cuentaBanco; SPC(5);  
COLOR 10  
PRINT digitoControl  
COLOR 7
```

```
SUB calcularDigitoControl (cuenta AS STRING, digito AS INTEGER)  
DIM i AS INTEGER  
DIM d AS INTEGER  
DIM suma AS INTEGER  
DIM num AS INTEGER  
num = 1  
FOR i = LEN(cuenta) TO 1 STEP -1  
d = VAL(MID$(cuenta, i, 1))  
suma = suma + d * num  
num = num + 2  
NEXT i  
digito = suma MOD 10  
END SUB
```

**PREGUNTAS DE TEORÍA DE VISUAL BASIC (2.5 puntos)**

- ¿Qué significa paso de parámetros por referencia y por valor tanto en QB como en VB? ¿Cuál es la utilidad del paso por referencia frente al paso por valor?
- Explicar las diferencias de las funciones y de los procedimientos entre QB y VB.
- Codificar ejemplos muy claros que reflejen las diferencias anteriores.