

# I.T. Informática.

## Programación I. Curso 2001-2002

### Práctica 2

## 1 Especificación del problema

Para estudiar los sistemas de representación, un alumno decide hacer un programa que muestre tablas de ejemplos de números positivos en diferentes bases. Las tablas mostrarán la representación de una serie de números (hasta 10) en todas las bases entre 2 y 16. Una tabla de ejemplos tendrá 15 filas y 10 columnas. Cada fila corresponderá a una base y cada columna a un número. En una columna previa se especificarán las bases correspondientes (ver figuras).

Los números a representar se podrán elegir de dos maneras: especificando un rango de enteros, o dando valores sueltos de números que pueden ser reales. Para ello, al pedir los datos, el programa presentará las opciones:

- r) Representar un rango de enteros
- s) Representar algunos números reales

En el caso **r** el programa pedirá dos enteros para formar un rango de números, y mostrará una tabla con tantas columnas como indique el rango y filas para las bases 2 a 16. Por ejemplo, para la entrada `4 12` mostrará la tabla que se ve en la figura 1. Se repetirá la petición de rango hasta que sea válido.

En el caso **s** el programa pedirá hasta 10 números (en decimal) al usuario. La serie acabará cuando el usuario introduzca un 0. Para números no válidos, se repetirá la petición (hasta uno positivo ó 0). Igualmente el programa mostrará la tabla, como la de la figura 1 para la entrada `1.5 0.5 0`

Una vez mostrada la tabla (en cualquiera de los dos casos), se presentará un menú que permitirá:

- a) Eliminar una fila
- b) Insertar una fila (para una base nueva o repetida)
- c) Eliminar una columna
- d) Insertar una columna (para un número nuevo o repetido)
- e) Imprimir la tabla en un fichero de texto
- f) Construir otra tabla
- g) Terminar el programa

Se pedirá:

- En los casos de inserción, el elemento nuevo (base o número) y el lugar de inserción. En ningún caso se permitirá la inserción si la tabla tiene ya el tamaño máximo.
- en los casos de eliminación, la posición de la fila o columna que se elimina.
- en el caso de impresión en fichero, el nombre del mismo, de salida, que se supone que el usuario proporcionará y que tendrá a lo sumo 12 caracteres (no es necesario validarlo).

Además de las especificaciones anteriores, el programa deberá solicitar, antes que otra cosa, el fichero del que se leerán las entradas, mediante opción:

- t) Teclado
- f) Fichero de texto

En el segundo caso, las entradas se leerán del fichero 'entrada.txt', en el que los datos pedidos se encontrarán dispuestos en líneas como muestra el ejemplo siguiente:

|             |   |
|-------------|---|
| r           | <i>rango de números</i>                   |
| 4 -12       | <i>extremos del rango, incorrectos</i>    |
| 5 13        | <i>extremos del rango, correctos</i>      |
| d           | <i>insertar una columna, imposible</i>    |
| c           | <i>eliminar una columna</i>               |
| 10          | <i>columna que se elimina</i>             |
| d           | <i>insertar una columna</i>               |
| 4           | <i>número que se inserta</i>              |
| 1           | <i>posición en la que se inserta</i>      |
| a           | <i>eliminar una fila</i>                  |
| 14          | <i>fila que se elimina</i>                |
| a           | <i>eliminar una fila</i>                  |
| 15          | <i>fila que se elimina, imposible</i>     |
| a           | <i>eliminar una fila</i>                  |
| 14          | <i>fila que se elimina</i>                |
| a           | <i>eliminar una fila</i>                  |
| 13          | <i>fila que se elimina</i>                |
| e           | <i>imprimir en un fichero</i>             |
| salida1.txt | <i>nombre del fichero en que se graba</i> |
| f           | <i>construir otra tabla</i>               |
| s           | <i>valores sueltos</i>                    |
| 1.5         | <i>primer valor</i>                       |
| 0.5         | <i>segundo valor</i>                      |
| -3          | <i>valor incorrecto</i>                   |
| 0           | <i>fin de la lista</i>                    |
| f           | <i>imprimir en un fichero</i>             |
| salida2.txt | <i>nombre del fichero en que se graba</i> |
| g           | <i>terminar el programa</i>               |

y cuyas salidas serán, además de los mensajes pertinentes (por pantalla) los ficheros 'salida1.txt' y 'salida2.txt', cuyos contenidos deben ser, respectivamente, el de la figura 1 sin las últimas 3 filas, y el de la figura 1.

Puede suponerse que

1. Los nombres de los ficheros que se pedirán son siempre válidos (es decir, cadenas de a lo sumo 12 caracteres que pueden ser nombres de ficheros)
2. cuando el programa espera entradas numéricas, lo serán.

**Se pide**

1. Realizar el análisis, diseño modular e implementación de un algoritmo que resuelva este problema.
2. Codificar el programa en Pascal

## 2 Parte opcional

Los alumnos que deseen mejorar el programa, incluirán en las validaciones la capacidad de la pantalla, de forma que no se permitan tablas que no quepan en la pantalla. También podrán permitir que el usuario elija el número de decimales con el que desea que se muestre cada número, para toda la tabla o para columnas o filas concretas. En este caso, presentará **además** del programa que resuelve las especificaciones anteriores, la versión mejorada.

## 3 Normas de entrega

Se debe entregar un documento de diseño y la codificación del programa en Pascal.

## 3.1 Documento de diseño

### 3.1.1 Identificación

- Nombre de los alumnos que la realizan (máximo 2).
- Nombre de usuario de cada uno
- Grupo de teoría al que pertenecen
- Subgrupo de prácticas al que pertenecen

### 3.1.2 Análisis

- Descripción breve del problema y aspectos importantes (máximo: 1 página).
- Elementos:
  - Entradas
  - Salidas
  - Suposiciones
  - ...

### 3.1.3 Diseño

- Planteamiento general de la solución y estructuras de datos utilizadas (máximo: 1 página)
- Diseño modular:
  - Esquema de módulos
  - Especificación de cada módulo definido (parámetros de entrada, salida, valor devuelto, y objetivo, al menos)

### 3.1.4 Programa

Deberá llamarse `p2.pas`. Las cuatro primeras líneas deberán tener el siguiente formato:

```
(* Usuario uuuuuuu *)  
(* Apellidos, Nombre *)  
(* Usuario uuuuu *)  
(* Apellidos, Nombre *)  
(* Grupo: Ln *)  
(* dd-mm-aaaa *)
```

donde el grupo `Ln` podrá ser `A1`, `A2`, ..., `C2`, `S1`, `S2`, `S3`. En el caso de presentar la parte opcional, el programa mejorado se llamará `p3.pas`, con la misma identificación inicial.

## 3.2 Criterios de evaluación

Es necesario obtener la calificación de “apto” para aprobar las prácticas. El “apto” se consigue entregando el programa fuente y la documentación detallada en este documento, ambos cumpliendo unos requisitos mínimos de calidad y pasando las baterías de pruebas.

## 3.3 Fechas

Entrega de documentación (al profesor de prácticas) y programa fuente en la cuenta del alumno: antes del 25 de enero de 2002 a las 20 h.

Examen de prácticas: se convocará oportunamente.

|    |     |     |     |     |      |      |      |      |      |
|----|-----|-----|-----|-----|------|------|------|------|------|
| 2  | 100 | 101 | 110 | 111 | 1000 | 1001 | 1010 | 1011 | 1100 |
| 3  | 11  | 12  | 20  | 21  | 22   | 100  | 101  | 102  | 110  |
| 4  | 10  | 11  | 12  | 13  | 20   | 21   | 22   | 23   | 30   |
| 5  | 4   | 10  | 11  | 12  | 13   | 14   | 20   | 21   | 22   |
| 6  | 4   | 5   | 10  | 11  | 12   | 13   | 14   | 14   | 20   |
| 7  | 4   | 5   | 6   | 10  | 11   | 12   | 13   | 14   | 15   |
| 8  | 4   | 5   | 6   | 7   | 10   | 11   | 12   | 13   | 14   |
| 9  | 4   | 5   | 6   | 7   | 8    | 10   | 11   | 12   | 13   |
| 10 | 4   | 5   | 6   | 7   | 8    | 9    | 10   | 11   | 12   |
| 11 | 4   | 5   | 6   | 7   | 8    | 9    | A    | 10   | 11   |
| 12 | 4   | 5   | 6   | 7   | 8    | 9    | A    | B    | 10   |
| 13 | 4   | 5   | 6   | 7   | 8    | 9    | A    | B    | C    |
| 14 | 4   | 5   | 6   | 7   | 8    | 9    | A    | B    | C    |
| 15 | 4   | 5   | 6   | 7   | 8    | 9    | A    | B    | C    |

Figura 1: Ejemplo de salida para rango de enteros

|    |          |          |
|----|----------|----------|
| 3  | 1.111111 | 0.111111 |
| 4  | 1.200000 | 0.200000 |
| 5  | 1.222222 | 0.222222 |
| 6  | 1.300000 | 0.300000 |
| 7  | 1.333333 | 0.333333 |
| 8  | 1.400000 | 0.400000 |
| 9  | 1.444444 | 0.444444 |
| 10 | 1.500000 | 0.500000 |
| 11 | 1.555555 | 0.555555 |
| 12 | 1.600000 | 0.600000 |
| 13 | 1.666666 | 0.666666 |
| 14 | 1.700000 | 0.700000 |
| 15 | 1.777777 | 0.777777 |
| 16 | 1.800000 | 0.800000 |

Figura 2: Ejemplo de salida con números sueltos