

Ingeniería T. de Informática Sistemas F. y Gestión
Examen de Programación 1. 7 de febrero del 2002

Apellidos _____

Nombre _____

Grupo _____

--	--	--	--	--	--	--	--	--

- | | |
|---|--|
| <ul style="list-style-type: none">• Duración del examen: 4 horas• Poner el nombre y apellidos en todas las hojas del examen. | <ul style="list-style-type: none">• Realizar cada problema en hojas diferentes• Se valorará la presentación |
|---|--|

Cuestiones

1. (1 pto) Realice la traza del siguiente fragmento de programa, usando los símbolos “-“ para indicar una variable no existente y “?” para indicar un valor desconocido. Además debe indicarse de qué tipo es cada variable y las acciones no recomendables que hay en el fragmento.

```
1. c1:=0; c2:=2;
2. new(pc);
3. pc^ := c1;
4. new(pc);
5. pc^ := c2;
6. write(pc^,c1);
7. qc := pc;
8. qc^ := c1;
9. write(pc^,c2);
10.new(qc);
11.qc^ := c2;
12.dispose(pc);
13.new(pc);
14.pc^ := c1;
15.pc := qc;
16.writeln(':',c1+c2+pc^+qc^+1,'/',pc^);
```

2. (1.25 ptos) Diseñe un subprograma que permita convertir un conjunto de elementos del tipo Tyele, en otro conjunto en el que se hayan eliminado los elementos mayor y menor. Obténgase el diagrama modular de la solución.

```
Type Tyele = ini..fin;
```

3. (1 pto) Especificuense, para las acciones que se enumeran, según proceda:
Si deberían ser función o procedimiento, parámetros de entrada, salida y/o entrada-salida (su significado y tipo), valor devuelto (significado y tipo), condiciones de funcionamiento (precondición), declaración en Pascal (cabecera, NO contenido) y ejemplo de llamada (uso) en Pascal
1. Suma de dos horas, expresadas en horas, minutos y segundos (enteros)
 2. Petición al usuario de una opción de menú como carácter, validando que esté dentro de un rango de caracteres de opciones válidas (de 'a' a 'h', por ejemplo)
 3. Escritura de una línea (de a lo suma 255 caracteres) en un fichero de texto, ajustada a una anchura determinada (insertando blancos adecuadamente).

4. (1 pto)¿Qué estructuras de datos serían idóneas para almacenar y manipular las siguientes informaciones?. Detállense las definiciones de los tipos en Pascal.
- 1 Las características de edad, peso, altura, nivel de actividad (0 a 5) y cantidad recomendada de ingesta calórica (en calorías/día) de un adolescente en la ficha de su médico.
 - 2 Un cartón de un juego de bingo, en el que, eventualmente, habrá que ‘tachar’ elementos. Un cartón contiene un número fijo de líneas, cada una de las cuales tiene un número fijo de casillas (por ejemplo, 3 líneas de 5 casillas cada una). En las casillas hay, aleatoriamente distribuidos, números enteros del 1 al 99. El jugador va tachando de su cartón los números que salen en un bombo, con el objeto de tachar antes que otro jugador todos los de alguna línea (cantar línea) o todos los de su cartón (cantar bingo).
 - 3 Todas las fichas del médico del apartado 1.

Ingeniería T. de Informática Sistemas F. y Gestión
Examen de Programación 1. 7 de febrero del 2002

5. (0.75 ptos) Diseñe un subprograma recursivo para obtener el producto de dos números enteros positivos. Especifíquense las pre y post condiciones, y estúdiense la finitud del algoritmo.

Nota: $a * b = a$ si $b=1$
 $a * b = a * (b-1) + a$ si $b>1$

Problema 1 (2.5 ptos)

Para hacer el cambio de pesetas a euros, utilizando exclusivamente monedas, un programador decide utilizar los siguientes tipos de datos:

```
TYPE
  TIndicePesetas = (p1, p5, p10, p25, p50, p100, p200, p500);
  TIndiceEuros = (c1, c2, c5, c10, c20, c50, e1, e2);
  TVectorPesetas = ARRAY [TIndicePesetas] OF INTEGER;
  TVectorEuros = ARRAY [TIndiceEuros] OF INTEGER;
```

Donde se supone que una variable de tipo TVectorPesetas contiene en cada posición el número de monedas que hay de cada valor de TIndicePesetas (lo mismo para el tipo TVectorEuros).

Desarrollar de forma modular un subprograma PASCAL que dada una cantidad en pesetas en el formato TVectorPesetas, devuelva esa misma cantidad en euros en el formato TVectorEuros, de forma que la cantidad de monedas utilizadas sea la mínima posible. Por ejemplo si la entrada del subprograma fuese:

p1	p5	p10	p25	p50	p100	p200	p500
2	0	0	0	0	1	2	2

Es decir, 2 monedas de 1 peseta, 1 moneda de 100 pesetas, 2 monedas de 200 pesetas y 2 monedas de 500 pesetas, en total 1502 pesetas, la salida correspondiente, (9,03 euros), debería ser 1 moneda de un céntimo de euro, 1 moneda de 2 céntimos de euro, 1 moneda de 1 euro y 4 de 2:

c1	c2	c5	c10	c20	c50	e1	e2
1	1	0	0	0	0	1	4

NOTA: Se recuerda que para realizar el cambio de pesetas a euros la equivalencia que se debe utilizar es: **1 euro = 166,386 pesetas**, aunque el resultado debe redondearse para manejar solo 2 dígitos decimales.

Se recuerda también que la declaración de la función ROUND es: ROUND (x:REAL): INTEGER.

Problema 2 (2.5 ptos)

Dado un fichero de texto (entrada.txt) que contiene una frase por línea, obtener un fichero de salida (salida.txt) de texto en el que cada palabra esté en orden inverso, conservando los blancos. Obténgase la solución en Pascal y hágase el diagrama modular de la solución planteada. Considérese que cada línea del fichero tiene palabras completas, y que cada línea no tiene más de 232 caracteres.

En el ejemplo siguiente cada • representa un espacio en blanco.

<p><i>Entrada.txt</i></p> <p>••invertir••esta• frase•• esto•es•un••••ejemplo••a</p>

<p><i>Salida.txt</i></p> <p>••ritrevni ••atse•esarf•• otse•se•nu••••olpmeje••a</p>
--