

Ingeniería T. de Informática Sistemas F. y Gestión
Examen de Programación I. 6 de Julio del 2001

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.• No se puede escribir a lápiz	<ul style="list-style-type: none">• Realizar cada problema en hojas diferentes• Se valorará la presentación
--	--

Problemas

1. (2 pts) Para ahorrar espacio en el almacenamiento de matrices cuadradas simétricas, de tipo tmatriz, se decide almacenar un vector con los elementos de la diagonal principal y los que están debajo de ella por filas como se indica a continuación, donde dado a_{ij} , i es la fila y j la columna:

$[a_{11}, a_{21}, a_{22}, a_{31}, a_{32}, a_{33}, a_{41}, a_{42}, a_{43}, a_{44}, \dots, a_{nn}]$

Const n = ...; Tmatriz = array [1..n, 1..n] of real;

Escribir un subprograma que permita obtener el vector a partir de la matriz original y otro que permita obtener la matriz original a partir del vector.

2. (2,5 pts) Dado un fichero de enteros, obtener el nº de elementos que aparecen en el fichero antes de que uno de ellos sea un nº de Mersene, mostrando por pantalla el primer número de Mersene y el nº de elementos anteriores a él. Si no hay ninguno se deberá mostrar sólo el nº de elementos del fichero.

Un nº N es de Mersene si y sólo si $N = 2^p - 1$, siendo p un nº primo.

Escribir un programa en Pascal para resolver este problema y **realizar el diagrama modular** del mismo. Suponer que no hay problemas de desbordamiento.

3. (1,5 pts) En un departamento hay 25 docentes. La categoría de cada docente puede ser:

PRAS AYEU AYUN PTEU PTUN CAUN

El sueldo base asignado a cada docente se calcula a partir de una fórmula que depende de la categoría. De cada docente se almacenan la categoría y las horas de docencia y las de investigación trabajadas cada día de la semana. Se necesita mantener la información de las horas trabajadas durante cada una de las semanas de un año.

1. Determinar las estructuras de datos necesarias para almacenar la información anterior, utilizando Pascal.
2. Especificar en Pascal cómo se podría acceder al nº de horas de investigación que trabajó en la semana 30 el docente nº 15.

4. (1,5 pts) Especificar el ámbito de cada identificador de variable y parámetro (lugares donde podría usarse) del siguiente programa y desde que módulos puede ser llamado cada uno de los módulos. Hacer la traza e indicar que prácticas no deseables se han utilizado en este programa.

Ingeniería T. de Informática Sistemas F. y Gestión
Examen de Programación I. 6 de Julio del 2001

```
Program BB (output);
var m, i, p, q : integer ;
function inc (var m : integer) : integer;
begin
  m := m+1 ; inc := m
end;
procedure z (var q, p : integer);
var i : integer;
function r (x, y : integer) : integer;
begin
  i := i+1;
  m := 2*x ;
  r := x+y
end;
begin
  i := q-p ;
  q := r (2, p);
  p := i
end;
BEGIN
  p := 100; q := 200 ; i := 0;
  i := inc (i) ;
  z (p, q);
  writeln (m, i, p, q)
END.
```

5. (1 ptos) Escribir un subprograma recursivo en Pascal que muestre los dígitos de un número entero no negativo en orden inverso. Por ejemplo dado el entero 1234 deberá mostrar 4 3 2 1.
6. (1 ptos) Se quiere obtener el número de elementos de un vector que son múltiplos de K (declarado como constante). Se han obtenido las siguientes soluciones:

```
Tvector = array [ini..fin] of integer;
```

```
Procedure multi1(var vector: tvector; var recuento: integer; var K:integer);
```

```
  Var i: ini..fin;
```

```
  Begin
```

```
  While i <= fin do
```

```
    begin
```

```
      if vector[i] mod k = 0 then recuento:=recuento+1;
```

```
      i:= i+1;
```

```
  End;
```

```
Function multi2(var v:tvector):integer;
```

```
Var i: ini..fin;
```

```
Begin
```

```
  Multi2:=0;
```

```
  i:=ini;
```

```
  While i<fin do
```

```
    If v[i] mod K=0 then Multi2:= Multi2+1;
```

```
  End;
```

Indique si estas soluciones tienen errores y si le parecen mejorables proponga otra solución.

7. (0,5 ptos) Decir qué ventajas tiene trabajar con módulos o subprogramas.