

Fundamentos de la programación y la informática
Examen de ejercicios de laboratorios
21 de junio de 2023
Grados en ingeniería aeroespacial. Turno de tarde
Universidad Rey Juan Carlos

- Ejecuta en un terminal la orden
`~mortuno/prepara`
- Para entregar el ejercicio, ejecuta
`~mortuno/entrega`
Conviene que relices varias entregas durante el examen, como medida de seguridad. La última elimina todas las anteriores.
- Recuerda que si alguno de los programas no compila, su nota será nula.

Ejercicio 1 (7 puntos)

En el fichero `~/fpi.junio.23/filtra.TULOGIN.pas` encontrarás el siguiente código

```
{ $mode objfpc } { $H- } { $R+ } { $T+ } { $Q+ } { $V+ } { $D+ } { $X- } { $warnings on }
```

```
program filtra_tabla;
const
  Elementos = 10;
  Frontera = 4;
type
  TipoTabla = array[1..Elementos] of integer;
function tira_dado(caras_dado: integer): integer;
begin
  result := random(caras_dado) + 1;
end;
```

Como puedes ver, se trata de la declaración de una tabla de números enteros. Añade a este fichero los subprogramas indicados a continuación. (Cada subprograma puede llamar a otros, si lo ves conveniente)

- Un subprograma llamado *inicia_tabla* que reciba una tabla (posiblemente vacía, aunque tu programa no tendrá esto en cuenta) de tipo `TipoTabla` y la devuelva rellena con los valores de la tirada de un dado. Pon al dado el número de caras que quieras, por ejemplo 6.
- Un subprograma llamado *escribe_tabla* que reciba una tabla de tipo `TipoTabla` y escriba sus valores en pantalla.
- Un subprograma llamado *filtra*, que reciba una tabla de tipo `TipoTabla` y devuelva otra tabla, del mismo tipo, pero con la siguiente modificación:
 - Aquellos valores mayores o iguales que la constante *Frontera*, serán reemplazados por la constante *Frontera*. Ejemplo: si *Frontera* vale 4, los valores 5 y 6 serán reemplazados por el valor 4.
 - Los valores que no cumplan la condición anterior, permanecerán inalterados.

Escribe además el código necesario para que el programa ejecute una vez los subprogramas anteriores, generando una salida similar a esta:

```
Tabla:
 5 1 6 2 6 6 3 2 4 3
Tabla Filtrada:
 4 1 4 2 4 4 3 2 4 3
```

Solución

```
{ $mode objfpc } { $H- } { $R+ } { $T+ } { $Q+ } { $V+ } { $D+ } { $X- } { $warnings on }
```

```
program filtra_tabla;
const
  Elementos = 10;
  Frontera = 4;
type
  TipoTabla = array[1..Elementos] of integer;

function tira_dado(caras_dado:integer):integer;
begin
  result := random(caras_dado) + 1;
end;

procedure inicia_tabla(var tabla:TipoTabla; caras_dado: integer);
var
  i : integer;
begin
  for i := 1 to Elementos do
    tabla[i] := tira_dado(caras_dado);
  end;

procedure escribe_tabla(tabla:TipoTabla);
var i: integer;
begin
  for i := 1 to Elementos do
    write(tabla[i]:3);
    writeln;
  end;

function filtra(var tabla_entrada:TipoTabla):TipoTabla;
var i: integer;
begin
  for i := 1 to Elementos do begin
    if (tabla_entrada[i] >= Frontera)
    then result[i] := Frontera
    else result[i] := tabla_entrada[i];
  end;
end;

var
  tabla, tabla_filtrada : Tipotabla;
const
  CarasDado = 6 ;
begin
  randomize();
  inicia_tabla(tabla, CarasDado);
  writeln('Tabla:');
  escribe_tabla(tabla);
  writeln('Tabla Filtrada:');
  tabla_filtrada := filtra(tabla);
  escribe_tabla(tabla_filtrada);
end.
```

Ejercicio 1 (3 puntos)

En el fichero `~/fpi.junio.23/huevos.TULOGIN.pas` encontrarás la solución propuesta para el ejercicio 2 del examen de ejercicios de enero de este año. Modifícalo de forma que:

1. El subprograma `clasifica_huevo` no sea una función, sino un procedimiento, que haga lo mismo. Naturalmente deberás hacer todo lo necesario para que el código siga funcionando.
2. Tras mostrar todos los pesos de los huevos, el programa indicará cuántos han resultado defectuosos, esto es, de la categoría `ERROR`. También el porcentaje, sin decimales. Ejemplo:

```
55.2 M
38.9 ERROR
[... etc]
59.9 M
46.9 S
Huevos defectuosos: 3 (10 %)
```

Solución

```
{ $mode objfpc } { $H- } { $R+ } { $T+ } { $Q+ } { $V+ } { $D+ } { $X- } { $warnings on }
```

```
{ $mode objfpc } { $H- } { $R+ } { $T+ } { $Q+ } { $V+ } { $D+ } { $X- } { $warnings on }
```

```
program huevos;
uses math;

function pesaje():real;
const
  Media = 63;
  DesvTip = 15;
begin
  result := randg(Media, DesvTip);
end;

procedure clasifica_huevo(peso:real; var categoria:string);
begin
  if (43 <= peso) and (peso < 53) then
    categoria := 'S'
  else if (53 <= peso) and (peso < 63) then
    categoria := 'M'
  else if (63 <= peso) and (peso <= 73) then
    categoria := 'L'
  else if (73 < peso) and (peso <= 83) then
    categoria := 'XL'
  else
    categoria := 'ERROR'
end;

procedure pesa_y_clasifica_huevos(N: integer);
var
  i: integer;
  peso: real;
  categoria: string;
  defectuosos: integer = 0;
begin
  for i:=1 to N do begin
    peso:= pesaje();
    write(peso:6:1, ' ');
    clasifica_huevo(peso, categoria);
    if (categoria = 'ERROR') then
      defectuosos := defectuosos + 1;
    writeln(categoria);
  end;
  write('Huevos defectuosos: ');
  write(defectuosos);
```

```
    write(' (');  
    write(100*(defectuosos / N):0:0);  
    writeln(' %)');  
end;  
  
const  
    N= 30;  
  
begin  
    randomize;  
    pesa_y_clasifica_huevos(N);  
end.
```