Examen de **Fundamentos de Informática**

Grado de Tecnologías Industriales

Escuela de Ingeniería y Arquitectura

Soluciones examen 16 de Junio de 2016

Departamento de Informática e Ingeniería de Sistemas

Universidad Zaragoza

Ejercicio 1

[3 puntos]

```
program anemometro;
const N=3;
type tpFichero = file of real;
     tpDatos   = array[1..N] of real;
var fin,fout:tpFichero; amediar:tpDatos;  sum:real; posicion:integer;
begin
   assign(fin,'velocidades.dat');
   reset(fin);

   assign(fout,'velocidadesfiltradas.dat');
   rewrite(fout);

   {Leemos todos y los vamos sumando}
   posicion:=1; sum:=0.0;
   while (posicion<N) and (not eof(fin)) do begin
      read(fin,amediar[posicion]);
      sum:=sum+amediar[posicion];
      posicion:=posicion+1;
   end;

   amediar[N]:=0; { Ponemos este valor para que funcione la primera vez que se
     entra en el bucle }

   {Ahora tenemos todos los elementos para hacer la primera media menos uno,
    salvo que ya hayamos llegado al final del fichero.}
   while not eof(fin) do begin
      {Quitamos de la suma el elemento que vamos a reemplazar}
      sum:=sum-amediar[posicion];
      {Reemplazamos el elemento en la posicion con uno nuevo}
      read(fin,amediar[posicion]);
      {Incluimos el elemento recientemente leido en la suma. De este modo
       sum siempre contiene la suma de todos los elementos de amediar.}
      sum:=sum+amediar[posicion];
      {Avanzamos la posicion ciclicamente}
      posicion:=(posicion mod N)+1;
      {Escribimos la media movil}
      write(fout,sum/N);
   end;
   close(fin);
   close(fout);
end.
```

```
program anemometro;
const N=3;
type tpFichero = file of real;
     tpDatos   = array[1..N] of real;
var fin,fout:tpFichero; amediar:tpDatos;  sum:real; i, leidos:integer;
begin
   assign(fin,'velocidades.dat');
   reset(fin);

   assign(fout,'velocidadesfiltradas.dat');
   rewrite(fout);

   leidos:=1;
   while not eof(fin) do begin
      read(fin,amediar[leidos]);
      if (leidos<N) then leidos:=leidos+1
      else begin
         sum:=0.0;
         for i:=1 to N do sum:=sum+amediar[i];
         write(fout,sum/N);
         for i:=1 to (N-1) do amediar[i]:=amediar[i+1];

      end;
   end;
   close(fin);
   close(fout);
end.
```

Ejercicio 2

[3.5 puntos]

```
program gotv2;
const
        NUM_TEMPORADAS = 6;
        NUM_EPISODIOS = 10;
type
        tpVistos = array[1..NUM_TEMPORADAS, 1..NUM_EPISODIOS] of boolean;

var
        f: text;
        vistos: tpVistos;
        temporada, episodio, stark, lann, barath, totalVistos, contNoVistosTemp: integer;
        espacio, casa: char;
begin
        { Indicadores de episodios no vistos por temporada }
        for temporada := 1 to NUM_TEMPORADAS do
                for episodio := 1 to NUM_EPISODIOS do
                        vistos[temporada, episodio] := FALSE;
        { Contadores de muertes por casa }
        stark := 0;
        lann := 0;
        barath := 0;

        { Numero de episodios vistos }
        totalVistos := 0;

        assign(f, 'got.txt');
        reset(f);
        while not eof(f) do begin
                read(f, temporada);
                read(f, episodio);
                vistos[temporada, episodio] := TRUE;
                while not eoln(f) do begin
                        read(f, espacio);  { Espacio separador (necesario para leer correctamente la letra) }
                        read(f, casa);
                        if casa = 'S' then
                                stark := stark + 1
                        else if casa = 'L' then
                                lann := lann + 1
                        else if casa = 'B' then
                                barath := barath + 1;
                end;
                readln(f);
        end;
        close(f);

        { Escribe datos sobre la serie }
        for temporada := 1 to NUM_TEMPORADAS do begin
                contNoVistosTemp := 0;
                for episodio := 1 to NUM_EPISODIOS do
                        if vistos[temporada, episodio] then
                                totalVistos := totalVistos + 1
                        else
                                contNoVistosTemp := contNoVistosTemp + 1;
                writeln('Faltan por ver ', contNoVistosTemp, ' episodios de la temporada ', temporada);
        end;
        writeln('Media de muertes: ', ((stark + lann + barath)/totalVistos):0:2);
        write('Casa con mas muertes: ');
        if (stark >= lann) and (stark >= barath) then
                writeln('Stark')
        else if (lann >= stark) and (lann >= barath) then
                writeln('Lannister')
        else
                writeln('Baratheon');

end.
```

```
program gotv3;
const
        NUM_TEMPORADAS = 6;
        NUM_EPISODIOS = 10;
type
        tpContadorNoVistos = array[1..NUM_TEMPORADAS] of integer;

var
        f: text;
        noVistos: tpContadorNoVistos;
        temporada, episodio, stark, lann, barath, totalVistos: integer;
        espacio, casa: char;
begin
        { Contadores de episodios no vistos por temporada }
        for temporada := 1 to NUM_TEMPORADAS do
                noVistos[temporada] := NUM_EPISODIOS;

        { Contadores de muertes por casa }
        stark := 0;
        lann := 0;
        barath := 0;

        { Numero de episodios vistos }
        totalVistos := 0;

        assign(f, 'got.txt');
        reset(f);

        while not eof(f) do begin
                read(f, temporada);
                read(f, episodio);
                noVistos[temporada] := noVistos[temporada] - 1;
                totalVistos := totalVistos + 1;
                while not eoln(f) do begin
                        read(f, espacio);  { Espacio separador (necesario para leer correctamente la letra) }
                        read(f, casa);
                        if casa = 'S' then
                                stark := stark + 1
                        else if casa = 'L' then
                                lann := lann + 1
                        else if casa = 'B' then
                                barath := barath + 1;
                end;
                readln(f);
        end;

        close(f);

        { Escribe datos sobre la serie }
        for temporada := 1 to NUM_TEMPORADAS do
                if noVistos[temporada] > 0 then
                        writeln('Faltan por ver ', noVistos[temporada], ' episodios de la temporada ', temporada);

        writeln('Media de muertes: ', ((stark + lann + barath)/totalVistos):0:2);
        write('Casa con mas muertes: ');
        if (stark >= lann) and (stark >= barath) then
                writeln('Stark')
        else if (lann >= stark) and (lann >= barath) then
                writeln('Lannister')
        else
                writeln('Baratheon');

end.
```

Ejercicio 3 [3.5 puntos]

```
const
   MAXFIL = 1024;
   MAXCOL = 1024;
   MAXNIVEL = 255;

type
   tpPixel = 0..MAXNIVEL;
   tpImagen = record
      nFils: 1..MAXFIL;
      nCols: 1..MAXCOL;
      dat: array[1..MAXFIL,1..MAXCOL] of tpPixel;
   end;
   tpHistograma = array[tpPixel] of integer;
```

```
procedure inicializaHistograma(var hist: tpHistograma);
var
   i: tpPixel;
begin
   for i:=0 to MAXNIVEL do
      hist[i]:= 0;
end;

procedure calcularHistograma(imagen: tpImagen; var hist: tpHistograma);
var
   i,j: integer;
begin
   inicializaHistograma(hist);
   for i:=1 to imagen.nFils do
      for j:=1 to imagen.nCols do
         hist[imagen.dat[i,j]] := hist[imagen.dat[i,j]] + 1;
end;
```

```
function calculaValDelMaximo(hist: tpHistograma): tpPixel;
var
   max: integer;
   valMax: tpPixel;
   i: tpPixel;
begin
   max := 0;    {no puede haber negativos}
   valMax := 0; {por defecto 0}
   for i:=0 to MAXNIVEL do
      if max < hist[i] then
      begin
         max := hist[i];
         valMax := i;
      end;
   calculaValDelMaximo := valMax;
end;

procedure umbralizar(var imagenAUmb: tpImagen);
var
   h: tpHistograma;
   umbral: tpPixel;
   i, j: integer;
begin
   calcularHistograma(imagenAUmb, h);
   umbral := calculaValDelMaximo(h);
   for i:=1 to imagenAUmb.nFils do
      for j:=1 to imagenAUmb.nCols do
         if imagenAUmb.dat[i,j] > umbral then
            imagenAUmb.dat[i,j] := MAXNIVEL
         else
            imagenAUmb.dat[i,j] := 0;
end;
```