

Trabajo Práctico N° 1.3: **Estructuras de Control (Adicionales).**

Ejercicio 1.

Realizar un programa que analice las inversiones de las empresas más grandes del país. Para cada empresa, se lee su código (un número entero), la cantidad de inversiones que tiene y el monto dedicado a cada una de las inversiones. La lectura finaliza al ingresar la empresa con código 100, que debe procesarse. El programa deberá informar:

- Para cada empresa, el monto promedio de sus inversiones.
- Código de la empresa con mayor monto total invertido.
- Cantidad de empresas con inversiones de más de \$50.000.

```
program TP1_E1;
{$codepage UTF8}
uses crt;
const
    empresa_salida=100;
    monto_corte=50000.0;
var
    i: int8;
    j, empresa, inversiones, empresa_max, empresas_corte: int16;
    monto, monto_total, monto_max: real;
begin
    randomize;
    monto_max:=-9999999; empresa_max:=0;
    empresas_corte:=0;
    repeat
        i:=random(100);
        if (i=0) then
            empresa:=empresa_salida
        else
            empresa:=1+random(high(int16));
            inversiones:=1+random(1000);
            monto_total:=0;
            for j:= 1 to inversiones do
                begin
                    monto:=1+random(1000);
                    monto_total:=monto_total+monto;
                end;
            textcolor(green); write('El monto promedio de las inversiones de la empresa ');
            textcolor(yellow); write(empresa); textcolor(green); write(' es '); textcolor(red);
            writeln(monto_total/inversiones:0:2);
            if (monto_total>monto_max) then
                begin
                    monto_max:=monto_total;
                    empresa_max:=empresa;
                end;
            if (monto_total>monto_corte) then
                empresas_corte:=empresas_corte+1;
            until (empresa=empresa_salida);
            textcolor(green); write('El código de la empresa con mayor monto total invertido es ');
            textcolor(red); writeln(empresa_max);
            textcolor(green); write('La cantidad de empresas con inversiones de más de $');
            textcolor(yellow); write(monto_corte:0:2); textcolor(green); write(' es '); textcolor(red);
            write(empresas_corte);
    end.
```

Ejercicio 2.

La cátedra de CADP está analizando los resultados de las autoevaluaciones que realizaron los alumnos durante el cuatrimestre. Realizar un programa que lea, para cada alumno, su legajo, su condición (I para INGRESANTE, R para RECURSANTE) y la nota obtenida en cada una de las 5 autoevaluaciones. Si un alumno no realizó alguna autoevaluación en tiempo y forma, se le cargará la nota -1. La lectura finaliza al ingresar el legajo -1. Una vez ingresados todos los datos, el programa debe informar:

- Cantidad de alumnos INGRESANTES en condiciones de rendir el parcial y porcentaje sobre el total de alumnos INGRESANTES.
- Cantidad de alumnos RECURSANTES en condiciones de rendir el parcial y porcentaje sobre el total de alumnos RECURSANTES.
- Cantidad de alumnos que aprobaron todas las autoevaluaciones.
- Cantidad de alumnos cuya nota promedio fue mayor a 6.5 puntos.
- Cantidad de alumnos que obtuvieron cero puntos en, al menos, una autoevaluación.
- Código de los dos alumnos con mayor cantidad de autoevaluaciones con nota 10 (diez).
- Código de los dos alumnos con mayor cantidad de autoevaluaciones con nota 0 (cero).

Nota: Recordar que, para poder rendir el EXAMEN PARCIAL, el alumno deberá obtener “Presente” en, al menos, el 75% del total de las autoevaluaciones propuestas. Se considera “Presente” la autoevaluación que se entrega en tiempo y forma y con, al menos, el 40% de respuestas correctas.

```
program TP1_E2;
{$codepage UTF8}
uses crt;
const
  condicion_i='I'; condicion_r='R';
  autoeva_total=5;
  nota_incumple=-1;
  legajo_salida=-1;
  nota_corte=4;
  promedio_corte=6.5;
  nota_cero=0;
  nota_diez=10;
  presente_corte=0.75;
var
  vector_condiciones: array[1..2] of char=(condicion_i, condicion_r);
  i, j, nota, presente, nota_total, notas_cero, notas_diez, notas_diez_max1, notas_diez_max2,
  notas_cero_max1, notas_cero_max2: int8;
  legajo, ingresantes_parcial, ingresantes_total, recursantes_parcial, recursantes_total,
  alumnos_autoeva, alumnos_corte, alumnos_cero, legajo_diez_max1, legajo_diez_max2,
  legajo_cero_max1, legajo_cero_max2: int16;
  condicion: char;
begin
  randomize;
  ingresantes_parcial:=0; ingresantes_total:=0;
  recursantes_parcial:=0; recursantes_total:=0;
  alumnos_autoeva:=0;
  alumnos_corte:=0;
  alumnos_cero:=0;
  notas_diez_max1:=0; notas_diez_max2:=0; legajo_diez_max1:=0; legajo_diez_max2:=0;
  notas_cero_max1:=0; notas_cero_max2:=0; legajo_cero_max1:=0; legajo_cero_max2:=0;
```

```

{Introducir legajo del alumno}
i:=random(100);
if (i=0) then
    legajo:=legajo_salida
else
    legajo:=1+random(high(int16));
while (legajo<>legajo_salida) do
begin
    {Introducir condición (I para INGRESANTE, R para RECURSANTE) del alumno}
    condicion:=vector_condiciones[1+random(2)];
    {Nota obtenida por el alumno en cada una de las 5 autoevaluaciones}
    presente:=0; nota_total:=0; notas_cero:=0; notas_diez:=0;
    for j:= 1 to autoeva_total do
    begin
        nota:=nota_incumple+random(12);
        if ((nota>nota_incumple) and (nota>=nota_corte)) then
            presente:=presente+1;
        if (nota<>nota_incumple) then
            nota_total:=nota_total+nota;
        if (nota=nota_cero) then
            notas_cero:=notas_cero+1;
        if (nota=nota_diez) then
            notas_diez:=notas_diez+1;
    end;
    {Cantidad de alumnos INGRESANTES en condiciones de rendir el parcial}
    if (condicion=condicion_i) then
    begin
        if (presente>=presente_corte*autoeva_total) then
            ingresantes_parcial:=ingresantes_parcial+1;
            ingresantes_total:=ingresantes_total+1;
        end
    {Cantidad de alumnos RECURSANTES en condiciones de rendir el parcial}
    else
    begin
        if (presente>=presente_corte*autoeva_total) then
            recursantes_parcial:=recursantes_parcial+1;
            recursantes_total:=recursantes_total+1;
        end;
    {Cantidad de alumnos que aprobaron todas las autoevaluaciones}
    if (presente=autoeva_total) then
        alumnos_autoeva:=alumnos_autoeva+1;
    {Cantidad de alumnos cuya nota promedio fue mayor a 6.5 puntos}
    if (nota_total/autoeva_total>promedio_corte) then
        alumnos_corte:=alumnos_corte+1;
    {Cantidad de alumnos que obtuvieron cero puntos en, al menos, una autoevaluación}
    if (notas_cero>=1) then
        alumnos_cero:=alumnos_cero+1;
    {Código de los dos alumnos con mayor cantidad de autoevaluaciones con nota 10 (diez)}
    if (notas_diez>notas_diez_max1) then
    begin
        notas_diez_max2:=notas_diez_max1;
        legajo_diez_max2:=legajo_diez_max1;
        notas_diez_max1:=notas_diez;
        legajo_diez_max1:=legajo;
    end
    else
        if (notas_diez>notas_diez_max2) then
        begin
            notas_diez_max2:=notas_diez;
            legajo_diez_max2:=legajo;
        end;
    {Código de los dos alumnos con mayor cantidad de autoevaluaciones con nota 0 (cero)}
    if (notas_cero>notas_cero_max1) then
    begin
        notas_cero_max2:=notas_cero_max1;
        legajo_cero_max2:=legajo_cero_max1;

```

```
    notas_cero_max1:=notas_cero;
    legajo_cero_max1:=legajo;
end
else
    if (notas_cero>notas_cero_max2) then
        begin
            notas_cero_max2:=notas_cero;
            legajo_cero_max2:=legajo;
        end;
    {Introducir legajo del alumno}
    i:=random(100);
    if (i=0) then
        legajo:=legajo_salida
    else
        legajo:=1+random(high(int16));
    end;
    if ((ingresantes_total>0) or (recursantes_total>0)) then
        begin
            if (ingresantes_total>0) then
                begin
                    textcolor(green); write('La cantidad de alumnos INGRESANTES en condiciones de rendir el
parcial y el porcentaje sobre el total de alumnos INGRESANTES son '); textcolor(red);
write(ingresantes_parcial); textcolor(green); write(' y '); textcolor(red);
write(ingresantes_parcial/ingresantes_total*100:0:2); textcolor(green); writeln('%',
respectivamente');
                end
            else
                begin
                    textcolor(red); writeln('No hay alumnos INGRESANTES (I)');
                end;
            if (recursantes_total>0) then
                begin
                    textcolor(green); write('La cantidad de alumnos RECURSANTES en condiciones de rendir el
parcial y el porcentaje sobre el total de alumnos RECURSANTES son '); textcolor(red);
write(recursantes_parcial); textcolor(green); write(' y '); textcolor(red);
write(recursantes_parcial/recursantes_total*100:0:2); textcolor(green); writeln('%',
respectivamente');
                end
            else
                begin
                    textcolor(red); writeln('No hay alumnos RECURSANTES (R)');
                end;
            textcolor(green); write('La cantidad de alumnos que aprobaron todas las autoevaluaciones
es '); textcolor(red); writeln(alumnos_autoeva);
            textcolor(green); write('La cantidad de alumnos cuya nota promedio fue mayor a ');
textcolor(yellow); write(promedio_corte:0:2); textcolor(green); write(' puntos es ');
textcolor(red); writeln(alumnos_corte);
            textcolor(green); write('La cantidad de alumnos que obtuvieron cero puntos en, al menos,
una autoevaluación es '); textcolor(red); writeln(alumnos_cero);
            textcolor(green); write('Los legajos de los dos alumnos con mayor cantidad de
autoevaluaciones con nota 10 (diez) son '); textcolor(red); write(legajo_diez_max1);
textcolor(green); write(' y '); textcolor(red); writeln(legajo_diez_max2);
            textcolor(green); write('Los legajos de los dos alumnos con mayor cantidad de
autoevaluaciones con nota 0 (cero) son '); textcolor(red); write(legajo_cero_max1);
textcolor(green); write(' y '); textcolor(red); write(legajo_cero_max2);
        end
    else
        begin
            textcolor(red); write('No hay alumnos INGRESANTES (I) o RECURSANTES (R)');
        end;
    end.
end.
```

Ejercicio 3.

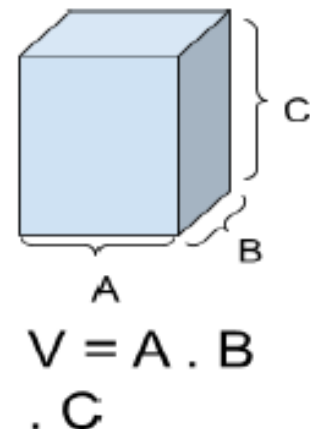
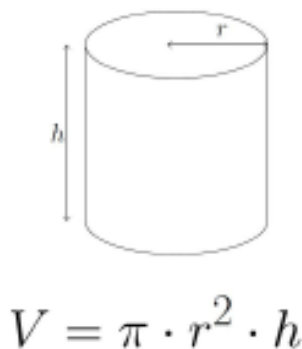
Un fabricante de tanques de agua está analizando las ventas de sus tanques durante el 2020. La empresa fabrica tanques a medida, que pueden ser rectangulares (tanques “R”) o cilíndricos (tanques “C”).

- De cada tanque R, se conoce su ancho (A), su largo (B) y su alto (C).
- De cada tanque C, se conoce su radio y su alto.

Todas las medidas se ingresan en metros. Realizar un programa que lea la información de los tanques vendidos por la empresa. La lectura finaliza al ingresar un tanque de tipo “Z”. Al finalizar la lectura, el programa debe informar:

- Volumen de los dos mayores tanques vendidos.
- Volumen promedio de todos los tanques cilíndricos vendidos.
- Volumen promedio de todos los tanques rectangulares vendidos.
- Cantidad de tanques cuyo alto sea menor a 1.40 metros.
- Cantidad de tanques cuyo volumen sea menor a 800 metros cúbicos.

Recordar: las fórmulas para el cálculo de volumen (V) del cilindro y del paralelepípedo rectangular son:



```
program TP1_E3;
{$codepage UTF8}
uses crt;
const
  tanque_r='R'; tanque_c='C';
  tanque_salida='Z';
  alto_corte=1.40;
  volumen_corte=800.0;
var
  vector_tanques: array[1..2] of char=(tanque_r, tanque_c);
  i, tanques_c, tanques_r, tanques_corte_alto, tanques_corte_volumen: int8;
  radio, alto, ancho, largo, volumen, volumen_max1, volumen_max2, volumen_total_c,
  volumen_total_r: real;
  tanque: char;
begin
  randomize;
  volumen_max1:=0; volumen_max2:=0;
```

```
volumen_total_c:=0; tanques_c:=0;
volumen_total_r:=0; tanques_r:=0;
tanques_corte_alto:=0;
tanques_corte_volumen:=0;
i:=random(100);
if (i=0) then
    tanque:=tanque_salida
else
    tanque:=vector_tanques[1+random(2)];
while (tanque<>tanque_salida) do
begin
    if (tanque=tanque_r) then
    begin
        ancho:=1+random(391)/10;
        largo:=1+random(391)/10;
        alto:=1+random(21)/10;
        volumen:=ancho*largo*alto;
        volumen_total_r:=volumen_total_r+volumen;
        tanques_r:=tanques_r+1;
    end
    else
    begin
        radio:=1+random(391)/10;
        alto:=1+random(21)/10;
        volumen:=pi*radio*radio*alto;
        volumen_total_c:=volumen_total_c+volumen;
        tanques_c:=tanques_c+1;
    end;
    if (volumen>volumen_max1) then
    begin
        volumen_max2:=volumen_max1;
        volumen_max1:=volumen;
    end
    else
        if (volumen>volumen_max2) then
            volumen_max2:=volumen;
    if (alto<alto_corte) then
        tanques_corte_alto:=tanques_corte_alto+1;
    if (volumen<volumen_corte) then
        tanques_corte_volumen:=tanques_corte_volumen+1;
    i:=random(100);
    if (i=0) then
        tanque:=tanque_salida
    else
        tanque:=vector_tanques[1+random(2)];
    end;
    if ((tanques_c>0) or (tanques_r>0)) then
    begin
        textcolor(green); write('El volumen de los mayores tanques vendidos es '); textcolor(red);
        write(volumen_max1:0:2); textcolor(green); write(' y '); textcolor(red);
        writeln(volumen_max2:0:2);
        if (tanques_c>0) then
        begin
            textcolor(green); write('El volumen promedio de todos los tanques cilíndricos (C)
            vendidos es '); textcolor(red); writeln(volumen_total_c/tanques_c:0:2);
        end
        else
        begin
            textcolor(red); writeln('No hay tanques cilíndricos (C) vendidos');
        end;
        if (tanques_r>0) then
        begin
            textcolor(green); write('El volumen promedio de todos los tanques rectangulares (R)
            vendidos es '); textcolor(red); writeln(volumen_total_r/tanques_r:0:2);
        end
        else
```

```
begin
    textcolor(red); writeln('No hay tanques rectangulares (R) vendidos');
end;
textcolor(green); write('La cantidad de tanques cuyo alto es menor a ');
textcolor(yellow); write(alto_corte:0:2); textcolor(green); write(' metros es ');
textcolor(red); writeln(tanques_corte_alto);
    textcolor(green); write('La cantidad de tanques cuyo volumen es menor a ');
textcolor(yellow); write(volumen_corte:0:2); textcolor(green); write(' metros cúbicos es ');
textcolor(red); write(tanques_corte_volumen);
end
else
begin
    textcolor(red); write('No hay tanques cilíndricos (C) o rectangulares (R) vendidos');
end;
end.
end.
```