

Universidad Nacional de Jujuy
Facultad de Ingeniería
Metodología De La Programación
Trabajo Práctico N° 6

Comisión N°5

Grupo N°5

Gutiérrez Karen

Paredes Mariana

Rivera Lucas

Ejercicio propuesto por el profesor

1. Diseñar un algoritmo modular que permita: a. Cargar 2 listas de ventas de lotes productos, las mismas están caracterizadas por:
i. Número de venta (Entero) ii. Nombre del lote (Cadena) iii. Costo del lote (Real) b. Mostrar las listas cargadas c. Ordenar las
listas y fusionar considerando un orden descendente por el costo del lote.

```

1. program TP06P1;
2. uses
3.     crt;
4. const
5.     MAX=100;
6. type
7.     TNombre = String[50];
8.     TProducto = record
9.         NroVenta : Integer;
10.        NombreLote : TNombre;
11.        CostoLote : Real;
12.    end;
13.    AProductos = Array[1..MAX] of TProducto;
14.
15.
16. (*-----ORDENACIONES-----*)
17.
18. procedure Intercambiar(var X:TProducto;var Y:TProducto);
19. var
20.     Temp:TProducto;
21. begin
22.     Temp:=X;
23.     X:=Y;
24.     Y:=Temp;
25. end;
26.
27. //seleccion
28. procedure OrdenarPorPrecioDesc(var Productos:AProductos; var N:integer);
29. var
30.     i, j,Minimo: Integer;
31. begin
32.     for i := 1 to N - 1 do
33.         begin
34.             Minimo:=i;
35.             for J := (I + 1) to N Do
36.                 if (Productos[j].CostoLote > Productos[Minimo].CostoLote) then
37.                     Minimo:=j;
38.             Intercambiar(Productos[i],Productos[Minimo]);
39.         end
40.     end;
41.
42.
43. (*-----CARGA Y UNION DE VECTORES-----*)
44.
45.
46. procedure CargarVentas (var lista:AProductos;var N:integer);
47. var
48.     i,CodigoError:integer;
49.     opcion:char;
50.     CosteLoteString:TNombre;
51.
52. begin
53.     I:=N+1;
54.     repeat
55.         with lista[i] do
56.             begin
57.                 NroVenta:=i;//nro de venta se genera automaticamente
58.                 Writeln('Venta Nro. ',i);
59.                 Write('Ingrese el nombre del lote -> ');
60.                 ReadLn(NombreLote);
61.                 Write('Ingrese el costo del lote -> ');
62.                 ReadLn(CosteLoteString);
63.                 Val(CosteLoteString,CostoLote,CodigoError);
64.                 while (CodigoError<>0) do
65.                     begin
66.                         write('Coste incorrecto, por favor ingrese un valor real->');
67.                         ReadLn(CosteLoteString);
68.                         Val(CosteLoteString,CostoLote,CodigoError);
69.                     end;
70.             end;

```

```

71.     inc(N);
72.     inc(i);
73.
74.     WriteLn();
75.     WriteLn('Desea ingresar otra venta? S/N ->');
76.     ReadLn(Opcion);
77.     Opcion := upCase(Opcion);
78.     until Opcion='N';
79. end;
80.
81.
82. (*-----MUESTRAS POR PANTALLA-----*)
83.
84. procedure MostrarCabecera();
85. begin
86.   Write('Nro Venta');GotoXY(20,1);
87.   Write('Nombre Lote');GotoXY(40,1);
88.   WriteLn('Costo Lote');
89. end;
90.
91. procedure MostrarVentaTabla(Producto:TProducto;y:integer);
92. begin
93.   with Producto do
94.     begin
95.       Write(NroVenta);GotoXY(20,y);
96.       Write(NombreLote);GotoXY(40,y);
97.       WriteLn(CostoLote:5:2);
98.     end;
99. end;
100.
101. procedure MostrarVentas (Lista:AProductos;N:integer);
102. var
103.   i:integer;
104. begin
105.   MostrarCabecera();
106.   for i:=1 to n do
107.     begin
108.       MostrarVentaTabla(Lista[i],i+1);
109.     end;
110. end;
111.
112. procedure MostrarListasFusionadas(var v1,v2:AProductos; N1,N2:integer);
113. var
114.   v3:AProductos;
115.   i,j,k,z,N3:integer;
116. begin
117.   i := 1; j:=1 ; k := 1;
118.   if(N1>= 2) and (N2>=2) then
119.     OrdenarPorPrecioDesc(v1,n1);
120.     OrdenarPorPrecioDesc(v2,n2);
121.     while (i<=N1) and (j<=N2) do
122.       begin
123.         if (v1[i].CostoLote> v2[j].CostoLote) then
124.           begin
125.             v3[k] := v1[i];
126.             i:= i+1;
127.           end
128.         else
129.           begin
130.             v3[k] := v2[j];
131.             j:= j+1;
132.           end;
133.         k:=k+1;
134.       end;
135.       if (i>N1) then
136.         for z:=j to N2 do
137.           begin
138.             v3[k]:=v2[z];
139.             k:=k+1;
140.           end
141.         else

```

```

142.     for z:=i to N1 do
143.     begin
144.         v3[k]:=v2[z];
145.         k:=k+1;
146.     end;
147.     N3:=N1+N2;
148.     MostrarVentas(V3,N3);
149. end;
150.
151. function MostrarMenu ():integer;
152. begin
153.     WriteLn('MENU');
154.     WriteLn('1- Cargar ventas de lista 1 o lista 2');
155.     WriteLn('2- Mostrar ventas de lista 1 o lista 2');
156.     WriteLn('3- Fusionar listas en una sola');
157.     WriteLn('0- Salir');
158.     Write('->');
159.     ReadLn(MostrarMenu);
160. end;
161.
162.
163. (*-----PRINCIPAL-----*)
164.
165. var
166.     Lista1,Lista2:AProductos;
167.     N1,N2:integer;
168.     Opcion:integer;
169. begin
170.     N1:=0;
171.     N2:=0;
172.     repeat
173.         ClrScr;
174.         Opcion:=MostrarMenu();
175.         case Opcion of
176.             1:
177.                 begin
178.                     WriteLn('Que lista desea cargar? 1 o 2?');
179.                     ReadLn(Opcion);
180.                     while (opcion<>1) and (Opcion<>2) do
181.                         begin
182.                             WriteLn('opcion incorrecta, debe ser 1 o 2');
183.                             ReadLn(Opcion);
184.                         end;
185.                     if (opcion=1) then
186.                         CargarVentas(Lista1,N1)
187.                     else
188.                         CargarVentas(Lista2,N2);
189.                     end;
190.                 2:
191.                 begin
192.                     WriteLn('Que lista desea mostrar? 1 o 2?');
193.                     ReadLn(Opcion);
194.                     while (opcion<>1) and (Opcion<>2) do
195.                         begin
196.                             WriteLn('opcion incorrecta, debe ser 1 o 2');
197.                             ReadLn(Opcion);
198.                         end;
199.                     if (opcion=1) then
200.                         begin
201.                             ClrScr;
202.                             MostrarVentas(Lista1,N1)
203.                         end
204.                     else
205.                         begin
206.                             ClrScr;
207.                             MostrarVentas(Lista2,N2);
208.                         end;
209.                     end;
210.                 3:
211.                 begin
212.                     ClrScr;

```

```

213.     MostrarListasFusionadas(lista1,lista2,n1,n2);
214.     end;
215.     0:writeln('Fin del programa');
216.     else
217.         writeln('Opcion incorrecta');
218.     end;
219.     Writeln;
220.     write('Presione una tecla para continuar...');
221.     ReadKey;
222.     until Opcion=0;
223. end.

```

3. Diseñar un programa que mediante un menú de opciones resuelva los ítems de más abajo. Siga la siguiente estrategia: el programa debe funcionar desde el principio, primero sin módulos y luego Ud. va agregando uno a uno los módulos, de menor a mayor complejidad y de tal manera que si se produce un error este estará en el último módulo agregado: a. Ingresar una frase en una cadena de caracteres. b. Mostrar cual es la letra que más se repite y el porcentaje de la misma en la frase. c. Cargar en un vector auxiliar las palabras de la frase y mostrar las palabras ordenadas de menor a mayor. d. Mostrar la suma aritmética de las letras que representan números enteros dentro de la frase. e. Elegir una palabra dentro de la frase original y sustituirla por otra palabra o frase deseada

```

1.  program TP06p3;
2.  uses
3.      crt,
4.      SysUtils, StrUtils;
5.  type
6.      Cadena = String[100];
7.      Vector = Array[1..100] of char;
8.  procedure MostrarLetraMasRepetida(var Cad:Cadena);
9.  var
10.     Porcentaje:real;
11.     MasRepetida,i:char;
12.     j,Apariciones,Max:integer;
13.  begin
14.     Max:=0;
15.     for i:=' ' to 'z' do
16.         begin
17.             apariciones:=0;
18.             for j:=1 to Length(cad) do
19.                 begin
20.                     if(cad[j]=i)then
21.                         inc(Apariciones);
22.                     end;
23.                     if (apariciones>max) then
24.                         begin
25.                             max:=Apariciones;
26.                             MasRepetida:= i;
27.                         end;
28.                     end;
29.             Porcentaje := Max/Length(cad)*100;
30.             Writeln('El caracter mas repetido es: ',MasRepetida,' con ',Porcentaje:4:2,'% de apariciones');
31.         end;
32.
33.  procedure Intercambiar(var X:char;var Y:char);
34.  var
35.     Temp:char;
36.  begin
37.     Temp:=X;
38.     X:=Y;
39.     Y:=Temp;
40.  end;
41.
42.  procedure OrdenarPorCaracter(var v:Vector;var n:integer);
43.  var
44.     i, j,Minimo: Integer;
45.  begin
46.     for i := 1 to N - 1 do
47.         begin
48.             Minimo:=i;
49.             for J := (I + 1) to N Do

```

```

50.         if (v[j] < v[Minimo]) then
51.             Minimo:=j;
52.             Intercambiar(v[i],v[Minimo]);
53.         end
54.     end;
55.
56. procedure MostrarFraseOrdenadaPorLetra (cad:Cadena);
57. var
58.     v:Vector;
59.     i,n:integer;
60. begin
61.     n:= Length(cad) ;
62.     for I:=1 to N do
63.         v[i]:=cad[i];
64.
65.         OrdenarPorCaracter(v,N);
66.         for I:=1 to N do
67.             write(v[i]);
68.         writeln;
69.     end;
70.
71. procedure MostrarSumaAritmetica(cad:Cadena);
72. var
73.     i,suma,ascii:integer;
74. begin
75.     suma:=0;
76.     for i:=1 to length(cad) do
77.         begin
78.             ascii:=Ord(cad[i]);
79.             if (ascii>=48) and (ascii<=57) then
80.                 suma:=suma + (ascii - 48);
81.             end;
82.             writeln('La suma aritmetica de los valores enteros es: ',suma);
83.         end;
84.
85.     var
86.         cad,subcad,sustituto:Cadena;
87.     begin
88.         WriteLn('Ingrese una cadena:');
89.         ReadLn(cad);
90.         MostrarLetraMasRepetida(cad);
91.         writeln;
92.         MostrarFraseOrdenadaPorLetra(cad);
93.         writeln;
94.         MostrarSumaAritmetica(cad);
95.         ReadKey;
96.         WriteLn('Ingrese la porcion de la cadena que desea sustituir');
97.         readln(subcad);
98.         WriteLn('Ingrese el texto que se reemplazara en la cadena');
99.         readln(sustituto);
100.        cad:= StringReplace(cad,subcad,sustituto, [rfReplaceAll, rfIgnoreCase]);
101.        Writeln('La cadena reemplazada es: ',cad);
102.        readkey;
103.
104.    end.

```