

## TEMA 2-CADP 2024 8/6//2024

Un fabricante de dispositivos electrónicos desea procesar información de los repuestos que compró. De cada repuesto conoce su código, precio, código de marca (entre 1 y 130) y nombre del país del que proviene. El fabricante dispone de una estructura de datos con la información de los repuestos, ordenados por nombre de país.

Realizar un programa que:

- a. Lea el código y el nombre de las 130 marcas con las que trabaja, y las almacene en una estructura de datos. La información se ingresa sin ningún orden en particular,
- b. Una vez completada la carga, procese la información de los repuestos e informe:
  1. La cantidad de países a los que se le compró más de 100 repuestos.
  2. Para cada marca, nombre de la marca y el precio del producto más barato.
  3. COMPLETO: La cantidad de repuestos que no poseen ningún cero en su código.

Implemente el programa principal que invoque a los módulos de los incisos a y b, e imprima los resultados.

### Resolución:

```
program parcial;
const
    DF=130;
type
    rango=1..DF;
    cadena15=string[15];

    repuesto=record
        cod:integer;
        precio:real;
        codM:rango;
        nomP:cadena15;
    end;
    lista=^nodo;
    nodo=record
        datos: repuesto;
```

```

        sig: lista;
    end;

    vectorMarcas = array [rango] of string;

    vectorPrecios = array [rango] of real;

procedure CargarLista(var l:lista); //se dispone

procedure LeerMarca(var cod:integer;var nombre:cadena15);
begin
    readln(cod);
    readln(nombre);
end;
procedure CargarVector(var v:vectorMarcas);
var
    i:rango;
    cod:integer;
    nombre:cadena15;
begin
    for i:= 1 to DF do
    begin
        LeerMarca(cod,nombre);
        v[cod]:=nombre;
    end;
end;

function SinCeros (cod:integer):boolean;
var
    dig:integer;
    stop:boolean;
begin
    stop:=false;
    while( ( cod <> 0) and (not stop) )do
    begin
        dig:= cod mod 10;
        cod:=cod div 10;
        if ( dig = 0 )then stop:=true;
    end;
    Sinceros:= not stop;
end;

procedure InicializarVector(var vec:vectorPrecios);
var
    i:rango;
begin

```

```

        for i:= 1 to DF do vec[i]:=9999;
end;

procedure actualizarMinimo(var minimo:real;nuevoValor:real);
begin
    if ( nuevoValor < minimo) then minimo:=nuevoValor;
end;

procedure Recorrer(l:lista; var vec:vectorPrecios;var
cantMas100,cantSinCeros:integer);
var
    actualPais:cadena15;
    repuestosPorPais:integer;
begin
    cantSinCeros:=0;
    cantMas100:=0;
    InicializarVector(vec);
    while ( l <> nil) do
        begin
            actualPais:=l^.datos.nomP;
            repuestosPorPais:=0;
            while ( ( l <> nil) and (actualPais = l^.datos.nomP) )do
                begin
                    repuestosPorPais:=repuestosPorPais+1;
                    actualizarMinimo(vec[l^.datos.codM],l^.datos.precio);
                    if ( SinCeros ( l^.datos.cod) )then
cantSinCeros:=cantSinCeros+1;
                        l:=l^.sig;
                    end;
                    if ( repuestosPorPais > 100 )then cantMas100:= cantMas100+1;
                    end;
                end;
        end;

procedure InformarMinimos(vec:vectorPrecios;v:vectorMarcas);
var
    i:rango;
begin
    for i:= 1 to DF do writeln('Para la marca ', v[i], ' el precio
mas barato fue ', vec[i]);
end;

var
    l:lista;
    v:vectorMarcas;
    cantMas100,cantSinCeros:integer;

```

```
        vec:vectorPrecios;
begin
    CargarLista(l); //se dispone
    CargarVector(v);
    Recorrer(l,vec,cantMas100,cantSinCeros);
    writeln('La cantidad de paises a los que se le compro mas de 100
repuestos: ', cantMas100);
    InformarMinimos(vec,v);
    writeln('La cantidad de repuestos que no poseen ningun cero en su
codigo: ', cantSinCeros);
end.
```