

Apellido y Nombre

Por favor, no escribir sobre esta fotocopia, resolver los ejercicios en hojas aparte.

1.- Se lee información de productos. De cada producto se lee código de producto, nombre del rubro y stock. Se pide realizar un programa que genere una estructura donde se almacenen, para cada rubro leído, su nombre, cantidad de productos del rubro y stock total, para los productos con stock distinto de cero. La lectura finaliza cuando se lee el código -1 y la lectura viene ordenada por nombre de rubro.

2. Teniendo en cuenta la siguiente declaración de tipos (type) y los siguientes procesos (A y B), indique para cada uno de los procesos si son correctos. El objetivo es eliminar el último nodo de la lista "l" recibida como parámetro. Justifique su respuesta (en caso de considerar algún proceso NO correcto, indicar todos sus errores).

```
type lista = ^nodo;
nodo = record
    dato: integer;
    sig: lista;
end;
miLista = record
    pri: lista;
    ult: lista;
end;
```

A	B
<pre>procedure uno (var l: miLista); var ant: lista; begin while (l.pri^.sig <> nil) do begin ant:= l.pri; l.pri:= l.pri^.sig; end; ant^.sig:= nil; dispose (l.ult); l.ult:= ant; end;</pre>	<pre>procedure dos (l: miLista); var ant: lista; begin while (l.pri <> l.ult) do begin ant:= l.pri; l.pri:= l.pri^.sig; end; ant^.sig:= nil; dispose (l.ult); l.ult:= ant; end;</pre>

3.- Dado el siguiente programa indique qué imprime en cada sentencia write. Justifique su respuesta.

```
program tres;
var c, d: integer;

procedure calcular (var a: integer; b: integer; var x: integer);
var d: integer;
begin
    d:= (b MOD 4) + c;
    b:= (d MOD 10) + d;
    if ((a + b) > 25) then b:= b + (a * 4)
                       else x:= (b + a) * 3;

    c:= (a-b) + c;
    writeln ('Valor a: ', a, ' Valor b: ', b, ' Valor x: ', x, ' Valor d: ', d);
end;

var a, b: integer;
begin
    a:= 4; b:= 6; c:= 3; d:= 12;
    calcular(d, c, a);
    writeln ('Valor a: ', a, ' Valor b: ', b, ' Valor c: ', c, ' Valor d: ', d);
end
```

4.- Indique Verdadero o Falso. Justifique en todos los casos:

- Si una variable declarada en el programa es de tipo simple, a la misma puede aplicarse la estructura de control CASE.
- Dados un vector y su dimensión lógica, siempre se puede utilizar un repeat until para recorrerlo e imprimir sus elementos.
- Buscar un elemento en una lista ordenada es más eficiente en tiempo de ejecución que buscar el mismo elemento en un arreglo ordenado.
- La cantidad de memoria total (estática + dinámica) requerida por el programa "progra" es de 187 bytes.
- El tiempo de ejecución del programa "progra" es de 123 UT.

```

program progra;
const
  dimF = 5;
type
  vector = array [1..dimF] of real;
  info = record
    nombre: string[15];
    edad: integer;
    montos: ^vector;
  end;
  lista = ^nodo;
  nodo = record
    dato: info;
    sig: lista;
  end;

var v: vector; i: integer; e: info;
    ok: boolean; monto: real; l, nue: lista;
begin
  l:= nil;
  read (e.nombre);
  ok:= true;
  while (e.nombre <> 'ZZZ') and (ok) do begin
    read (e.edad);
    if (e.edad MOD 2 = 0) then begin
      new(e.montos);
      for i:= 1 to dimF do begin
        read (monto);
        e.montos[i]:= monto;
      end;
      new(nue); nue^.dato:= e; nue^.sig:= l;
      l:= nue;
      read(e.nombre);
    end
    else ok:= false;
  end;
end.

```

Char	1 byte
Integer	6 bytes
Real	8 bytes
Boolean	1 byte
String	Longitud + 1 byte
Puntero	4 bytes