

1.- Después de la ejecución de P(x,y), obtener los valores de x e y:

```
VAR x,y:INTEGER;
PROCEDURE P(VAR y:INTEGER; z:INTEGER);
  BEGIN
    x:=y-1;
    y:=z;
    z:=y;
  END P;
```

```
...
  x:=2;
  y:=1;
  P(x,y);
...
```

2.- Dado el siguiente código, calcular el valor final de A:

```
...
A:=5;
INC(DEC(A)+1);
...
```

3.- Indicar la expresión que se produce al ejecutar el siguiente código:

```
cont:=0;
WHILE cont<2 DO
  FOR cont:=1 TO 3 DO
    WriteInt(cont,2);
  END;
  INC(cont);
END;
```

4.- Después de la ejecución de P(x,x), obtener el valor de x:

```
VAR x:INTEGER;
PROCEDURE P(y:INTEGER; VAR z:INTEGER);
  BEGIN
    x:=3;
    y:=y+3;
    z:=z+1;
  END P;
```

```
...
  x:=2;
  P(x,x);
...
```

5.- ¿Cuántas veces se ejecuta la sentencia IF en el siguiente bucle?

```
A:=FALSE;
B:=TRUE;
LOOP
  IF NOT(A OR B) THEN
    EXIT
  END;
  B:=A AND B;
END;
```

6.- Realizar un TAD con una tabla de 10 puntos en el plano (x,y) y con las siguientes operaciones: Suma de la longitud de los segmentos entre los puntos sucesivos de la tabla desde un punto inicial a otro final.

7.- Implementar el TAD que representa un vector de longitud 3, tal que en los extremos contenga colores básicos (rojo, amarillo, azul) y en el centro almacena la mezcla de los colores de las otras 2 posiciones ($r+am=naranja$, $r+az=violeta$, $am+az=verde$).

```

DEFINITION MODULE plano;
  TYPE
    tipopunto=RECORD
      x,y:REAL;
    END;
    tipotabla=ARRAY [1..10] OF tipopunto;
  PROCEDURE suma(a:tipotabla;pi,pf:INTEGER):REAL;
END plano.

```

```

IMPLEMENTATION MODULE plano;
FROM MathLib0 IMPORT sqrt;
  PROCEDURE suma(a:tipotabla;pi,pf:INTEGER):REAL;
  VAR
    i:INTEGER;
    acumulado,segmento:REAL;
  BEGIN
    acumulado:=0.0;
    FOR i:=pi TO pf-1 DO
      segmento:=sqrt((a[i].x-a[i+1].x)*(a[i].x-a[i+1].x)+
        (a[i].y-a[i+1].y)*(a[i].y-a[i+1].y));
      acumulado:=acumulado + segmento;
    END;
    RETURN acumulado;
  END suma;
END plano.

```

```

MODULE prinpl;
FROM InOut IMPORT WriteInt, WriteLn, WriteString, ReadInt;
FROM RealInOut IMPORT WriteReal;
FROM plano IMPORT tipotabla, tipopunto, suma;
VAR
  t:tipotabla;
  i,pi,pf:INTEGER;
  sumatotal:REAL;
BEGIN
  FOR i:=1 TO 10 DO
    t[i].x:=FLOAT(i)+1.0;
    t[i].y:=FLOAT(i)+2.5;
    WriteString("Punto ");
    WriteInt(i,1);
    WriteString(": ");
    WriteReal(t[i].x,5);
    WriteString(" , ");
    WriteReal(t[i].y,5);
    WriteLn;
  END;
  WriteString("Introducir punto inicial del 1 al 10: ");
  ReadInt(pi);
  WriteLn;
  WriteString("Introducir punto final del 1 al 10: ");
  ReadInt(pf);
  WriteLn;
  sumatotal:=suma(t,pi,pf);
  WriteReal(sumatotal,10);
  WriteLn;
END prinpl.

```

```

DEFINITION MODULE colores;
  TYPE
    tipocolor=(rojo,amarillo,azul,naranja,violeta,verde);
    vectorcolor=ARRAY [1..3] of tipocolor;
  PROCEDURE mezclarcolores(VAR a:vectorcolor);
END colores.

IMPLEMENTATION MODULE colores;
PROCEDURE mezclarcolores(VAR a:vectorcolor);
BEGIN
  IF (a[1]=rojo) OR (a[3]=rojo) THEN
    IF (a[1]=amarillo) OR (a[3]=amarillo) THEN
      a[2]:=naranja;
    ELSE
      a[2]:=violeta;
    END;
  END;
  IF (a[1]=amarillo) OR (a[3]=amarillo) THEN
    IF (a[1]=azul) OR (a[3]=azul) THEN
      a[2]:=verde;
    END;
  END;
END mezclarcolores;
END colores.

MODULE principal;
FROM colores IMPORT tipocolor, vectorcolor, mezclarcolores;
FROM InOut IMPORT WriteString,WriteLn;
VAR
  x:vectorcolor;
  y,z:tipocolor;
PROCEDURE imprimir(a:tipocolor);
BEGIN
  CASE a OF
    amarillo: WriteString("amarillo")|
    azul: WriteString("azul")|
    verde: WriteString("verde")|
    naranja: WriteString("naranja")|
    rojo: WriteString("rojo")|
    violeta: WriteString("violeta")
  END;
END imprimir;
BEGIN
  FOR y:=rojo TO azul DO
    x[1]:=y;
    FOR z:=rojo TO azul DO
      x[3]:=z;
      IF x[1]<>x[3] THEN
        mezclarcolores(x);
        imprimir(x[1]);
        WriteString("-");
        imprimir(x[2]);
        WriteString("-");
        imprimir(x[3]);
        WriteLn;
      END;
    END;
  END;
END principal.

```

1.- Cuál de las líneas de código es correcta si se ejecuta el siguiente fragmento de código: `PROCEDURE f(x:REAL):INTEGER;`

- a) `f(3.0)`
- b) `v:=2*f(3.0)`
- c) `v:=2.0*f(3.0)`
- d) `v:=2*f(3)`

2.- Es correcto? `VAR Diametro:Radio*2`

3.- `sqrt` es:

- a) un procedimiento de `MathLib0`
- b) no existe
- c) una función para `x` de tipo real
- d) una sentencia para `x` de tipo real

4.- Cuál es falsa?

- a) todo dato encapsulado es abstracto
- b) todo tipo opaco es abstracto
- c) todo tipo abstracto es dato encapsulado
- d) con los módulos se definen tipos abstractos

5.- En el siguiente fragmento: `Algo.Local; Otro.Remoto;`

- a) `Local` puede ser una variable
- b) `Remoto` es una función
- c) `Otro` es un módulo
- d) `Algo` puede ser un registro

6.- Cuál de las sentencias es correcta, dada la declaración:

```
TYPE TipoUno = (A,B,C,D);
TIPODOS = SET OF TipoUno;
```

- a) `TipoDos:=TipoDos+TipoUno{A}`
- b) `VAR Variable:TipoDos{}`
- c) `CONST Constante=TipoDos{B,D}`
- d) `INCL(TipoDos,E)`

7.- Cuál de las asignaciones es correcta?

```
TYPE Colores=(Rojo,Verde,Azul);
TYPE Letras=["a".."z"];
VAR lista:ARRAY Colores OF Letras;
```

- a) `lista["a"]:=Azul`
- b) `lista[1]:="a"`
- c) `lista[Rojo]:="m"`
- d) `lista[Verde]:=b`

8.- Escribir una expresión que seleccione el campo y del tercer elemento de una variable `v` de tipo `Vector`:

```
TYPE TipoPunto= RECORD
    x,y:REAL;
END;
tipoVector=ARRAY [1..5] OF TipoPunto;
```

9.- Es correcto? `TYPE vocales=("A","E","I","O","U");`

10.- Es correcto `semana:=TipoDias{L,M};?`

```
TYPE TipoDias=(L,M,X,J);
VAR semana: SET OF TipoDias;
```

11.- Obtener el resultado final:

```
VAR c1,c2:CARDINAL;  
FOR c1:=-1 TO 1 DO  
  FOR c2:= 1 TO 2 DO  
    WriteString("Hola");  
  END;  
END;
```

12.- Es correcto? PROCEDURE Simetrico (texto:ARRAY OF CHAR)

13.- Es correcto? PROCEDURE Imprimir (VAR A: SET OF CHAR)

14.-Cuál de las afirmaciones es correcta?

```
DEFINITION MODULE Primero;  
TYPE TipoDato;  
END Primero.  
IMPLEMENTATION MODULE Primero;  
PROCEDURE Magico(dato:TipoDato);  
END Primero.
```

- a) Magico es anónimo
- b) Magico es opaco
- c) Nadie podrá usar Magico
- d) No tiene sentido definir Magico