

1. Responde brevemente a las siguientes preguntas:

- a) Define cinco criterios de calidad del software que influyan de forma directa en el aspecto final del código fuente. (1 pt.)
- b) Sea el siguiente bucle:

```

N x,y,z
MIENTRAS guarda(x,y) HACER
    S1(x,z)
    S2(y,z)
    S3(y)
FIN MIENTRAS
    
```

donde `guarda` es una función Booleana y `Si(v)` representa una sentencia que accede (lee o escribe) a la variable `v`. Escribe un procedimiento recursivo equivalente. (1 pt.)

2. Escribir un algoritmo que diga si una frase contenida en un array es un palíndromo o no. La frase puede contener letras y otros caracteres. Sólo se tienen en cuenta las letras para ver si es un palíndromo. (2 pts.)

```

A R R O Z , A L A Z O R R A \0
    
```

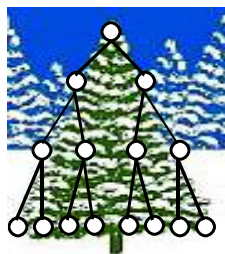
La frase anterior sí es un palíndromo cuando no se tienen en cuenta ni la coma ni los espacios en blanco

3. Se quiere utilizar un árbol binario para gestionar el encendido de las luces de un Árbol de Navidad. Se supone que el árbol está lleno, es decir, que todos los niveles están completos. Cada una de las bolas se identificará por dos números (nivel, posición en el nivel). Así, el nivel 0 tendrá la bola (0,1) y en el nivel 2 las bolas (2,1), (2,2),(2,3),(2,4). Se trabajará a nivel de implementación. Los tipos a utilizar son los siguientes:

```

TIPOS
REGISTRO NodoArbolBin
    B encendido //falso=Apagada/cierto=Encendido
    NodoArbolBin * izq
    NodoArbolBin * Der
FIN
NodoArbolBin * ArbolBin

REGISTRO TCoordenada
    N nivel
    N posicion
FIN
TCoordenada TGuirnalda[1..MAXBOLAS]
    
```



Nivel 0
Nivel 1
Nivel 2
Nivel 3

Se piden los algoritmos:

```

o // Recibe un arbol y las coordenadas de la bola que debe ser encendida
  Algoritmo EncenderBola(ES ArbolBin arbol, E N nivel, E N posicion)
    
```

(2 pts.)

(1 pt.)

```
o // Recibe un array de bolas que deben ser encendidas.  
  // Por ejemplo: (3,1),(2,1),(3,3),(2,3),(3,6),(3,8)  
  Algoritmo Guirnalda(ES ArbolBin arbol, E TGuirnalda guirnalda)
```

4. Se tiene una función del sistema que devuelve el primer ítem *de un directorio* dado `PrimerItem(E Cadena dado, S Cadena resul)`. Si no hay ninguno se devuelve en `resul` una cadena vacía. Para seguir obteniendo los demás elementos hay que ir llamando a `DameSiguiente(S Cadena resul)` sucesivamente hasta que ya no haya más elementos en el directorio dado y se obtenga en `result` una cadena vacía. Cuando uno de los elementos devueltos es a su vez un directorio el propio nombre del ítem acaba en `/`. Hacer una función que, (**utilizando una cola**), muestre el contenido *completo* (todos los elementos, subdirectorios y contenidos de éstos) dentro de un directorio inicial dado. (3 pt.)
- Suponer que existe también una función `Escribe(E Cadena s)` capaz de escribir la cadena `s` en la pantalla de un terminal.