

## 1 ● El siguiente algoritmo

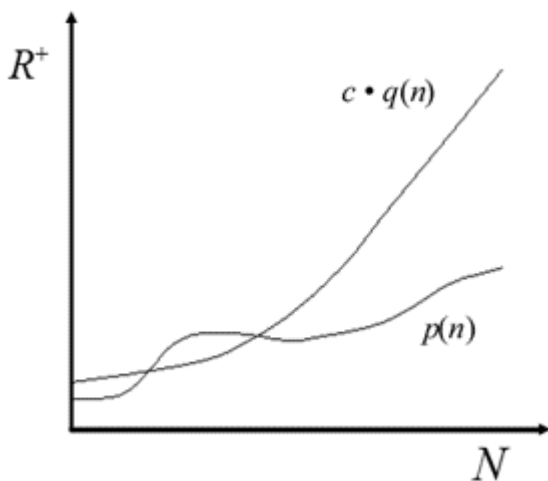
```
función casomejor(A:real[n]):real
  i: entero
  x: real
  x ← 0
  si n ≤ 2
    devolver x
  si no
    para i ← 1 hasta n hacer
      si  $A_i > 2$ 
         $x \leftarrow x + A_x$ 
      fsi
    fpara
  fsi
  devolver x
ffunción
```

- ☒ Tiene caso mejor cuando  $A_i$  es menor o igual que 2.
- ☐ Tiene caso mejor cuando  $n$  es menor o igual que 2.
- ☐ Ambas respuestas son correctas.

## 2 ● ¿Cuál de las siguientes expresiones es falsa?

- ☐  $O(\log n) \subset O(n^2) \subset O(n!)$
- ☒  $O(n^{10}) \subset O(n^n) \subset O(n!)$
- ☐  $O(\log \log n) \subset O(2^n) \subset O(n!)$

## 3 ● Dada la siguiente gráfica donde $p$ y $q$ son funciones y $c$ es una constante decimos que



- ☒  $p(n)$  es de orden superior  $q(n)$
- ☐  $q(n)$  es de orden superior  $p(n)$
- ☐  $p(n)$  es de orden inferior  $q(n)$

4 ● ¿Cuál es la función  $T(n)$  del algoritmo f?

(NOTA: La expresión  $\text{sum\_}\{j=1\}^{\{x\}}[\dots]$  indica el sumatorio desde  $j=1$  hasta  $x$  de todas las instrucciones que se incluyan en los corchetes.)

```
función f(V[n]:entero):entero
    x,i,z:entero
(1)    z <- 0
(2)    para x<-2 hasta n-1 hacer
(3)        i <- 1
(4)        mientras i <= x hacer
(5)            z <- z + Vi - i
(6)            i <- i - 1
(7)        fmientras
(8)    fpara
(9)    devolver z
ffunción
```

- ☐  $T(n) = t(1) + t(2) + (n-2)*t(3) + \text{sum\_}\{x=2\}^{\{n-1\}} [ t(4) + x * ( t(5) + t(6) ) ] + t(9)$
- ☐  $T(n) = t(1) + t(2) + (n-2)*t(3) + \text{sum\_}\{i=1\}^{\{x\}} [ t(4) + x * ( t(5) + t(6) ) ] + t(9)$
- ☒ Ninguna respuesta es correcta.

5 ● ¿Cuál es la complejidad asintótica del siguiente algoritmo? La complejidad asintótica de  $\text{calcular}(n:\text{real}, m:\text{entero}):\text{real}$  es  $O(m)$ .

```
función algoritmo(A:real[x,y]):real
    i,j: entero
    v: real
    v <- 0
    i <- x
    mientras i ≥ 1 hacer
        j <- 1
        mientras j < y hacer
            v <- v + calcular(Ai,j,y)
            j <- j + 2
        fmientras
        i <- i / 2
    fmientras
    devolver v
ffunción
```

- ☐  $O(y * \log x)$
- ☒  $O(y^2 * \log x)$
- ☐ Ninguna respuesta es correcta.

6 ● El algoritmo iterativo correspondiente al algoritmo f1 es

```

función f1(a:natural U {0}, b:natural U {0}):natural U {0}
  si a < b
    devolver b + f1(a + 1, b - a)
  si no
    devolver a * b
  fsi
ffunción

función algoritmo1(a:natural U {0},
  b:natural U {0}):natural U {0}
  a1, b1, s: natural U {0}
  a1 ← a
  b1 ← b
  mientras a1 < b1 hacer
    b1 ← b1 - a1
    a1 ← a1 + 1
  fmientras
  s ← a1 * b1
  mientras !(a1 = a) hacer
    a1 ← a1 - 1
    b1 ← b1 + a1
    s ← b1 + s
  fmientras
  devolver s
ffunción

función algoritmo2(a:natural U {0},
  b:natural U {0}):natural U {0}
  a1, b1, s: natural U {0}
  a1 ← a
  b1 ← b
  mientras a1 < b1 hacer
    a1 ← a1 + 1
    b1 ← b1 - a1
  fmientras
  s ← a1 * b1
  mientras !(a1 = a) hacer
    a1 ← a1 - 1
    b1 ← b1 + a1
    s ← b1 + s
  fmientras
  devolver s
ffunción

```

- ☒ algoritmo1
- ☐ algoritmo2
- ☐ Ninguna respuesta es correcta.

## 7 ● El resultado de la siguiente función para a=6 y b=8 es

```

función calcular(a:natural U {0}, b:natural U {0}):natural U {0}
  si a < b
    devolver b + calcular(a + 1, b - a)
  si no
    devolver a * b
  fsi
ffunción

```

- ☐ 14
- ☒ 22
- ☐ Ninguna respuesta es correcta.

## 8 ● Sean f y g dos funciones definidas de la forma $f(n)=f(n-1)+g(n-2)$ y $g(n)=n/2$ . La función f presenta una recursividad del tipo

- ☐ lineal final.
- ☐ múltiple.
- ☒ Ninguna respuesta es correcta.

## 9 ● El siguiente algoritmo se puede transformar en iterativo

```

función g(n:natural):natural
  si n<1
    devolver n
  si no
    x <- g(n-1)
    devolver g(n-2) + x
  fsi
ffunción

```

- ☐ aplicando el esquema general de transformación de la recursividad lineal no final con inversa.

- ☐ aplicando tanto el esquema general de la recursividad lineal no final con inversa como con pila.
- ☒ Ninguna respuesta es correcta.

## 10 ¿Cuál de estas afirmaciones es verdadera?

- ☐ Los algoritmos recursivos que siguen la estrategia divide y vencerás se pueden convertir a iterativo utilizando la estrategia de programación dinámica.
- ☐ Los algoritmos divide y vencerás tienen como expresión del tiempo de ejecución una expresión del tipo  $T(n) = T(n/b) + c \cdot n^k$ , donde  $n$  es el tamaño del problema,  $n/b$  el tamaño de cada subproblema y  $c \cdot n^k$  el coste de dividir el problema en subproblemas y combinar las soluciones.
- ☒ El algoritmo mergesort tiene una recursividad múltiple y puede implementarse utilizando un único vector auxiliar para dividir el vector original a ordenar en dos partes.

## 11 El algoritmo quicksort en su versión recursiva

- ☐ puede tener en algunos casos una complejidad asintótica menor que la del algoritmo mergesort.
- ☐ puede tener en algunos casos una complejidad asintótica del  $O(2^n)$ .
- ☒ tiene el mismo tipo de recursividad que la del algoritmo mergesort recursivo.

## 12 El resultado de la función combinar del algoritmo mergesort para los vectores $V1=\{10\ 10\ 20\ 30\}$ y $V2=\{20\ 40\ 50\ 60\}$ es

- ☒ 10 10 20 20 30 40 50 60
- ☐ 10 20 30 40 50 60
- ☐ Ninguna respuesta es correcta.

## 13 Para realizar la distribución de partidos de un torneo de pádel utilizando el algoritmo del Torneo suponiendo que hay 13 parejas participantes en el que cada pareja puede tener un partido por día, ha de jugar con todas las parejas una vez y como mucho tiene un día de descanso utilizando el algoritmo del Torneo se tiene que completar un panel para

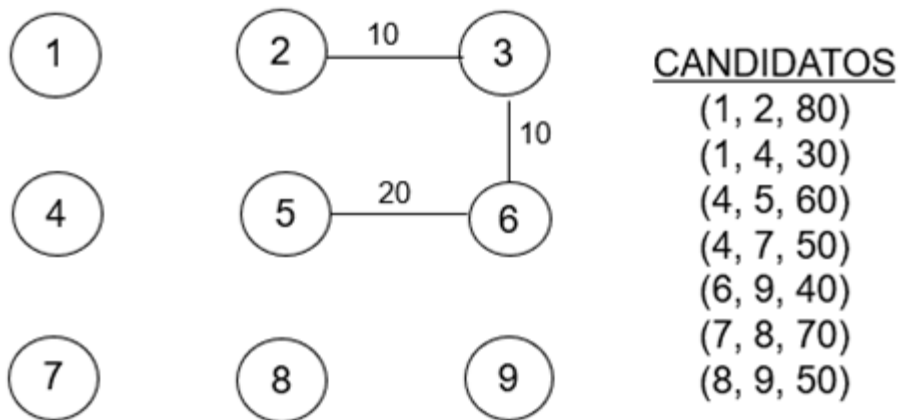
- ☐ 6 parejas.
- ☐ 5 parejas.
- ☒ 4 parejas.

## 14 El algoritmo que obtiene el coste del camino más corto entre un vértice y todos los demás es

- ☐ Viajante de comercio
- ☒ Dijkstra
- ☐ Prim

## 15 Dado el siguiente grafo al que se está aplicando el algoritmo de Prim, ¿cuál de las aristas que hay en el conjunto de candidatos sería la siguiente en seleccionar?

**NOTA:** (a, b, peso) indica una arista que une los vértices a y b con el peso especificado en el último valor.



- ☐ La que une los vértices 1 y 2.
- ☐ La que une los vértices 1 y 4
- ☒ La que une los vértices 6 y 9.

### 16 ¿Cuál de estas afirmaciones es verdadera?

- ☐ El algoritmo original de Dijkstra calcula los vértices que forman parte del camino más corto entre un vértice y todos los demás.
- ☐ El algoritmo de Dijkstra sigue una estrategia divide y vencerás.
- ☒ La complejidad asintótica del algoritmo de Dijkstra es polinomial.

### 17 La programación dinámica

- ☒ se puede utilizar para convertir en iterativo algunos algoritmos con recursividad múltiple.
- ☐ suele ser una estrategia menos eficiente asintóticamente que los algoritmos de ramificación y poda.
- ☐ Ninguna respuesta es correcta.

### 18 El problema de la mochila se puede resolver por programación dinámica cuando

- ☐ los pesos son enteros.
- ☒ los pesos son enteros y los objetos no se fraccionan.
- ☐ los pesos y los valores son enteros, independientemente de que los objetos se puedan fraccionar o no.

### 19 Dada la siguiente matriz resultante de aplicar el algoritmo de Floyd correspondiente al vértice 1, donde INF indica un valor muy alto, ¿cuál es el resultado de la matriz tras aplicarlo para el vértice 2?

0	10	14	5	34
10	0	12	10	24
30	32	0	20	20
23	10	20	0	3
10	20	INF	INF	0

- ☐ Fila 1: 0, 10, 14, 5, 34 ; Fila 2: 10, 0, 12, 10, 24 ; Fila 3: 30, 32, 0, 20, 20 ; Fila 4: 20,10, 20, 0, 3 ; Fila 5: 10, 20, INF, INF, 0
- ☐ Fila 1: 0, 10, 14, 5 y 34 ; Fila 2: 10, 0, 12, 10, 24 ; Fila 3: 30, 30, 0, 20, 20, Fila 4: 23,10, 20, 0, 3; Fila 5: 10, 20, 32, 30, 0
- ☒ Ninguna respuesta es correcta.

**20** • Supongamos que tenemos una mochila con un peso máximo de 11 y que se está resolviendo el problema aplicando el algoritmo de programación dinámica visto en la asignatura. Suponiendo que para el objeto 3 cuyo peso es 5 y valor 8 los valores máximos que se pueden llevar son los que se indican, ¿cuál es el resultado para el objeto 4 con peso 3 y valor 12?

Peso máximo:	1	2	3	4	5	6	7	8	9	10	11
Objeto 3:	0	0	0	7	8	10	10	10	15	17	18

- ☐ 0, 0, 12, 12, 12, 12, 12, 12, 19, 20, 22
- ☒ 0, 0, 12, 12, 12, 12, 19, 20, 22, 22, 22
- ☐ Ninguna respuesta es correcta.

**21** • ¿Cuál de las siguientes afirmaciones es falsa?

- ☐ Los algoritmos de vuelta atrás tienen complejidades muy elevadas.
- ☒ Los algoritmos de vuelta atrás realizan un recorrido en anchura del árbol.
- ☐ Los algoritmos de vuelta atrás son algoritmos que utilizan un método de prueba y error.

**22** • En el problema de la mochila con vuelta atrás para una mochila de objetos con pesos 30, 40, 35, 20 y valores 35, 30 y 20, 40 con capacidad máxima de 60, suponiendo que la solución óptima hasta ese momento es 70, ¿cuántas veces se actualiza la solución óptima en la rama que se expande a partir del nodo (1,0,\_,\_) ?

- ☐ 0
- ☒ 1
- ☐ 2

**23** • ¿A qué función hace referencia la siguiente función correspondiente al problema de la asignación de tareas con coste de penalización?

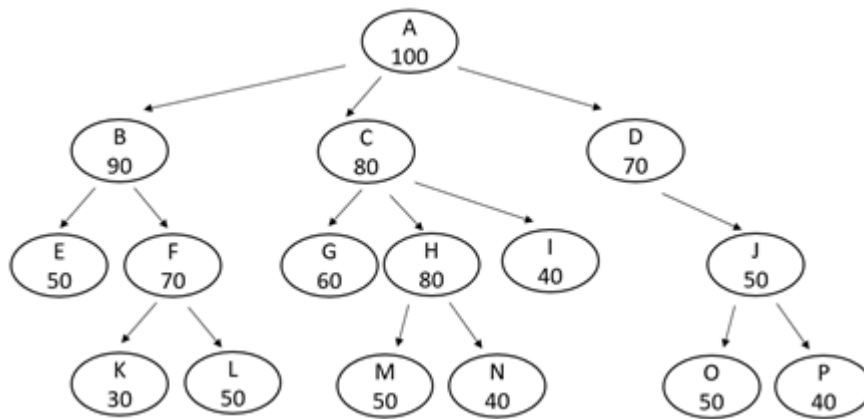
```
función _____(valornodo:entero, valorsolución:entero):booleano
    si valornodo < valorsolución
        devolver VERDADERO
    si no
        devolver FALSO
    fsi
ffunción
```

- ☒ Mejor
- ☐ Admisible
- ☐ Factible

**24** • La poda de un nodo se realiza cuando

- ☐ el valor de la cota del nodo es menor que el de la solución que se tenga en ese momento.
- ☐ un nodo no cumple las restricciones del problema.
- ☒ Ninguna respuesta es correcta.

**25** • Suponiendo que el orden de selección de nodos en el siguiente árbol es A, B, C, H, F, D. ¿Qué estrategia de selección de nodos se ha aplicado?



- ☐ FIFO
- ☐ MC-FIFO
- ☒ MC-LIFO

**26** ¿Qué hace la función Solución en el algoritmo que resuelve el problema de las tareas con coste de penalización?

- ☐ Comprueba que el número de tareas sea menor que el valor de la cota.
- ☐ Comprueba que el número de tareas sea igual a 0.
- ☒ Ninguna respuesta es correcta.

**27** ¿Qué tipo de recursividad presenta el siguiente algoritmo?

función obtener(x:entero, y:entero):entero

z: entero

z ← 0

si x < 10

z ← x + y

si no

si x < 20

z ← x + obtener(x-1,y)

fsi

si y ≤ 0

z ← z + obtener(x-1, y+1)

fsi

fsi

devolver z

ffunción

- ☐ Lineal final.
- ☐ Lineal no final.
- ☒ Múltiple.

**28** ☒ ¿Cuál de estas estrategias voraces obtiene el mejor valor para la mochila discreta?

- ☐ Introducir primero los objetos de mayor valor por unidad de peso.
- ☐ Introducir primero los objetos de mayor valor.
- ☒ Ninguna respuesta es correcta.

**29** ☒ ¿Qué valor es válido para inicializar la solución inicial de un problema cuando el objetivo es maximizar una función objetivo?

- ☐ 0
- ☐ + infinito
- ☒ - infinito