

NOMBRE: _____ APELLIDOS: _____

CUESTIONES (1:15 h)

1) Responde a las siguientes cuestiones, indicando si son Verdaderas o Falsas. Las preguntas contestadas erróneamente restan $\frac{1}{2}$ pregunta. (1 pto)

- a) $\frac{n^2}{\sqrt{n}} + n - \frac{1}{n^3} \in \Omega(n)$ [] b) $\lg \lg n^3 \in O(\lg n)$ [] c) $\frac{1}{n} \sin(n^2) \in \Theta(1)$ []
- d) $O(n^2) \subseteq O(n^3)$ [] e) $O(2^{n \lg n}) \subset 2^{O(n \lg n)}$ [] f) $\sqrt{n} \in \Theta(\lg n)$ []
- g) $\Omega(n^2) \subseteq \Omega(n^3)$ [] h) $2^{\log_3 n} \in \Theta(3^{\log_2 n})$ [] i) $\sqrt{\lg n} \in O(\lg \sqrt{n})$ []

2) Considera el problema del cajero automático que tiene cuatro tipos de billetes de 10 €, 5€, 2€ y 1€, en el que se quiere que el cajero distribuya el menor número de billetes. Responde a las siguientes preguntas (1 pto):

a) ¿Puede un algoritmo voraz resolver este problema de forma óptima para cualquier cantidad mayor o igual a 1 €? ¿Y un algoritmo de vuelta a atrás (backtracking)? (0,25 pts)

b) Para esta situación, ¿el coste de un algoritmo de vuelta a atrás con respecto al algoritmo voraz será Mejor, Igual o Peor? Justifica tu respuesta. (0,25 pts)

c) Si no existiesen billetes de 1 €. ¿Qué algoritmo tendríamos que utilizar?. Da una cota superior del coste temporal y espacial de dicho algoritmo. (0,5 pts)

3) Considera el siguiente algoritmo y responde a las preguntas que se indican (2 pts)

Algoritmo MATRIZ

DATOS: M : vector[1..n,1..n] de N

RES: R : N

METODO:

R ← 0

para i ← 1 hasta n hacer

si M[i,i] < 0 entonces

para j ← 1 hasta i-1 hacer R ← R + M[j,i]

para j ← i+1 hasta n hacer R ← R - M[i,j]

fsi

si M[i,i] > 0 entonces

para j ← i+1 hasta n hacer R ← R - M[j,i]

para j ← 1 hasta i-1 hacer R ← R + M[i,j]

fsi

fpara

fMATRIZ

a) ¿Cuál es el tamaño de la entrada? (0,5 pts)

b) ¿Cuál es el número de instancias? (0,5 pts)

c) Calcula el coste temporal exacto en función del número de asignaciones sobre el resultado (variable R) para el caso mejor y peor. Indicando cual es cada uno de los casos. (1 pto)

4) Resuelve la siguiente ec. de recurrencia (2 ptos) $T(n) = \begin{cases} 2 & n \leq 1 \\ 3T(n-1) + 2^n & n > 1 \end{cases}$

NOMBRE: _____ APELLIDOS: _____

PROBLEMAS (45 min.)**(Se pueden utilizar apuntes y libros de clase)****5) Dado el siguiente problema responde a las preguntas que se te plantean (6 pts).**

“Mac Arrón” ha asaltado un almacén de productos químicos. El almacén tiene n componentes (líquidos) cada uno de los cuales posee una densidad d_i y un volumen v_i . “Mac Arrón” sólo dispone de una garrafa de x litros. Su “jefe” sólo le ha dicho que tiene que traer la garrafa llena, pero como “Mac Arrón” es el que tiene que escapar con la garrafa a cuestas cuanto menos pese mejor. (Recuerda que la densidad se define como el peso/volumen).

PREGUNTAS:

- a) Escribe un algoritmo que ayude a “Mac Arrón” solucionar este problema. Utiliza la cabecera que se te adjunta y completa la sección de resultado y de método. (2 pts)

Algoritmo GARRAFA (n: N)

```

DATOS: d: vector[1..n] de R // Vector ordenado de MAYOR A MENOR con la densidad
      // de los elementos.
      v: vector[1..n] de R // Vector con el volumen en litros correspondiente a
      // cada elemento
      G: R // Capacidad de la garrafa en litros.
RES: ¿?
```

MÉTODO: ¿?

fGARRAFA

- b) Define la instrucción crítica y calcula el coste temporal del algoritmo en el caso mejor y peor en función de la I.C.. Indica cual es cada uno de estos casos (2 ptos)
- c) ¿De que tipo de algoritmo se trata?. ¿En este tipo de casos, obtendremos siempre la solución óptima? ¿Le serviría este algoritmo si le piden que asalte el almacén de una mina donde sólo se guardan lingotes de metal de distinto tamaño y “Mac Arrón” debe llenar un cajón?. (1 pto)
- d) Cual sería el coste en el caso medio suponiendo que la distribución de volumen de los líquidos es equiprobable (1 ptos)