

Pregunta 1 Incorrecta Puntúa como 1,00 🚩 Marcar pregunta	<p>Dado un problema de optimización, el método voraz ...</p> <p>Seleccione una:</p> <p><input checked="" type="radio"/> a. ... siempre obtiene una solución factible. ✗</p> <p><input type="radio"/> b. Ninguna de las otras dos opciones es cierta.</p> <p><input type="radio"/> c. ... siempre obtiene la solución óptima.</p>
Pregunta 2 Incorrecta Puntúa como 1,00 🚩 Marcar pregunta	<p>En el método voraz ...</p> <p>Seleccione una:</p> <p><input checked="" type="radio"/> a. ... siempre se encuentra solución pero puede que no sea la óptima. ✗</p> <p><input type="radio"/> b. ... el dominio de las decisiones sólo pueden ser conjuntos discretos o discretizables.</p> <p><input type="radio"/> c. ... es habitual preparar los datos para disminuir el coste temporal de la función que determina cuál es la siguiente decisión a tomar.</p>
Pregunta 3 Correcta Puntúa como 1,00 🚩 Marcar pregunta	<p>Dada la suma de la recurrencia</p> $T(n) = \begin{cases} 1 & n = 0 \\ \sum_{k=0}^{n-1} T(k) & n > 0 \end{cases}$ <p>¿cuál de las siguientes afirmaciones es cierta?</p> <p>Seleccione una:</p> <p><input type="radio"/> a. $T(n) \in \Theta(n^2)$</p> <p><input type="radio"/> b. $T(n) \in \Theta(n!)$</p> <p><input checked="" type="radio"/> c. $T(n) \in \Theta(2^n)$ ✓</p>

<p>Pregunta 4</p> <p>Sin contestar</p> <p>Puntúa como 1,00</p> <p>🚩 Marcar pregunta</p>	<p>Cuando se calculan los coeficientes binomiales usando la recursión $\binom{n}{r} = \binom{n-1}{r} + \binom{n-1}{r-1}$, con $\binom{n}{0} = \binom{n}{n} = 1$, qué problema se da y cómo se puede resolver?</p> <p>Seleccione una:</p> <p><input type="radio"/> a. La recursión puede ser infinita y por tanto es necesario organizarla según el esquema iterativo de programación dinámica.</p> <p><input type="radio"/> b. Se repiten muchos cálculos y ello se puede evitar usando programación dinámica.</p> <p><input type="radio"/> c. Se repiten muchos cálculos y ello se puede evitar haciendo uso de una estrategia voraz.</p>
<p>Pregunta 5</p> <p>Correcta</p> <p>Puntúa como 1,00</p> <p>🚩 Marcar pregunta</p>	<p>La programación dinámica...</p> <p>Seleccione una:</p> <p><input type="radio"/> a. ... en algunos casos se puede utilizar para resolver problemas de optimización con dominios continuos pero probablemente pierda su eficacia ya que puede disminuir drásticamente el número de subproblemas repetidos.</p> <p><input checked="" type="radio"/> b. Las otras dos opciones son ciertas. ✓</p> <p><input type="radio"/> c. ... normalmente se usa para resolver problemas de optimización con dominios discretizables puesto que las tablas se han de indexar con este tipo de valores.</p>
<p>Pregunta 6</p> <p>Correcta</p> <p>Puntúa como 1,00</p> <p>🚩 Marcar pregunta</p>	<p>En la solución al problema de la mochila continua ¿por qué es conveniente la ordenación previa de los objetos?</p> <p>Seleccione una:</p> <p><input type="radio"/> a. Para reducir la complejidad temporal en la toma de cada decisión: de $O(n^2)$ a $O(n \log n)$, donde n es el número de objetos a considerar.</p> <p><input checked="" type="radio"/> b. Para reducir la complejidad temporal en la toma de cada decisión: de $O(n)$ a $O(1)$, donde n es el número de objetos a considerar. ✓</p> <p><input type="radio"/> c. Porque si no se hace no es posible garantizar que la toma de decisiones siga un criterio voraz.</p>
<p>Pregunta 7</p> <p>Correcta</p> <p>Puntúa como 1,00</p> <p>🚩 Marcar pregunta</p>	<p>Se pretende implementar mediante programación dinámica iterativa la función recursiva:</p> <pre>unsigned f(unsigned x, unsigned v[]) { if (x==0) return 0; unsigned m = 0; for (unsigned k = 0; k < x; k++) m = max(m, v[k] + f(x-k, v)); return m; }</pre> <p>¿Cuál es la mejor estructura para el almacén?</p> <p>Seleccione una:</p> <p><input type="radio"/> a. int A</p> <p><input checked="" type="radio"/> b. int A[] ✓</p> <p><input type="radio"/> c. int A[][]</p>
<p>Pregunta 8</p> <p>Correcta</p> <p>Puntúa como 1,00</p> <p>🚩 Marcar pregunta</p>	<p>Un informático quiere subir a una montaña y para ello decide que tras cada paso, el siguiente debe tomarlo en la dirección de máxima pendiente hacia arriba. Además, entenderá que ha alcanzado la cima cuando llegue a un punto en el que no haya ninguna dirección que sea cuesta arriba. ¿qué tipo de algoritmo está usando nuestro informático?</p> <p>Seleccione una:</p> <p><input type="radio"/> a. un algoritmo divide y vencerás.</p> <p><input checked="" type="radio"/> b. un algoritmo voraz. ✓</p> <p><input type="radio"/> c. un algoritmo de programación dinámica.</p>

<p>Pregunta 9</p> <p>Correcta</p> <p>Puntúa como 1,00</p> <p>▼ Marcar pregunta</p>	<p>¿Cuál de los siguientes pares de problemas son equivalentes en cuanto al tipo de solución (óptima, factible, etc.) aportada por el método voraz?</p> <p>Seleccione una:</p> <p><input checked="" type="radio"/> a. El fontanero diligente y la mochila continua. ✓</p> <p><input type="radio"/> b. El fontanero diligente y la asignación de tareas.</p> <p><input type="radio"/> c. El fontanero diligente y el problema del cambio.</p>
<p>Pregunta 10</p> <p>Correcta</p> <p>Puntúa como 1,00</p> <p>▼ Marcar pregunta</p>	<p>De los problemas siguientes, indicad cuál no se puede tratar eficientemente como los otros dos:</p> <p>Seleccione una:</p> <p><input checked="" type="radio"/> a. El problema de la mochila sin fraccionamiento y sin restricciones en cuanto al dominio de los pesos de los objetos y de sus valores. ✓</p> <p><input type="radio"/> b. El problema del cambio, o sea, el de encontrar la manera de entregar una cantidad de dinero usando el mínimo de monedas posibles.</p> <p><input type="radio"/> c. El problema de cortar un tubo de forma que se obtenga el máximo beneficio posible.</p>
<p>Pregunta 11</p> <p>Correcta</p> <p>Puntúa como 1,00</p> <p>▼ Marcar pregunta</p>	<p>Se pretende implementar mediante programación dinámica iterativa la función recursiva:</p> <pre>float f(unsigned x, int y){ if(y < 0) return 0; float A = 0.0; if (v1[y] <= x) A = v2[y] + f(x-v1[y], y-1); float B = f(x, y-1); return min(A,2+B); }</pre> <p>¿Cuál es la mejor complejidad espacial que se puede conseguir?</p> <p>Seleccione una:</p> <p><input checked="" type="radio"/> a. $O(y)$ ✓</p> <p><input type="radio"/> b. $O(1)$</p> <p><input type="radio"/> c. $O(y^2)$</p>
<p>Pregunta 12</p> <p>Correcta</p> <p>Puntúa como 1,00</p> <p>▼ Marcar pregunta</p>	<p>Cuando la descomposición recursiva de un problema da lugar a subproblemas de tamaño similar, ¿qué esquema promete ser más apropiado?</p> <p>Seleccione una:</p> <p><input type="radio"/> a. Divide y vencerás, siempre que se garantice que los subproblemas no son del mismo tamaño.</p> <p><input checked="" type="radio"/> b. Programación dinámica. ✓</p> <p><input type="radio"/> c. El método voraz.</p>