

|   |  |
|---|--|
| <p><b>Pregunta 1</b></p> <p>Correcta</p> <p>Puntúa como 1,00</p> <p>🚩 Marcar pregunta</p> | <p>El valor que se obtiene con el método voraz para el problema de la mochila discreta es ...</p> <p>Seleccione una:</p> <p><input checked="" type="radio"/> a. ... una cota inferior para el valor óptimo que a veces puede ser igual a este. ✓</p> <p><input type="radio"/> b. ... una cota superior para el valor óptimo.</p> <p><input type="radio"/> c. ... una cota inferior para el valor óptimo, pero que nunca coincide con este.</p>   |
| <p><b>Pregunta 2</b></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><b>Pregunta 3</b></p> <p>Correcta</p> <p>Puntúa como 1,00</p> <p>🚩 Marcar pregunta</p> | <p>¿Cuál de estas estrategias para calcular el <math>n</math>-ésimo elemento de la serie de Fibonacci (<math>f(n) = f(n-1) + f(n-2)</math>, <math>f(1) = f(2) = 1</math>) es más eficiente?</p> <p>Seleccione una:</p> <p><input type="radio"/> a. Las dos estrategias citadas serían similares en cuanto a eficiencia.</p> <p><input checked="" type="radio"/> b. Programación dinámica. ✓</p> <p><input type="radio"/> c. La estrategia voraz.</p>   |
| <p><b>Pregunta 4</b></p> <p>Correcta</p> <p>Puntúa como 1,00</p> <p>🚩 Marcar pregunta</p> | <p>¿Cuál de estos tres problemas de optimización no tiene, o no se le conoce, una solución voraz óptima?</p> <p>Seleccione una:</p> <p><input checked="" type="radio"/> a. El problema de la mochila discreta o sin fraccionamiento. ✓</p> <p><input type="radio"/> b. El árbol de cobertura de coste mínimo de un grafo conexo.</p> <p><input type="radio"/> c. El problema de la mochila continua o con fraccionamiento.</p>   |
| <p><b>Pregunta 5</b></p> <p>Correcta</p> <p>Puntúa como 1,00</p> <p>🚩 Marcar pregunta</p> | <p>La eficiencia de los algoritmos voraces se basa en el hecho de que ...</p> <p>Seleccione una:</p> <p><input checked="" type="radio"/> a. ... las decisiones tomadas nunca se reconsideran. ✓</p> <p><input type="radio"/> b. ... antes de tomar una decisión se comprueba si satisface las restricciones del problema.</p> <p><input type="radio"/> c. ... con antelación, las posibles decisiones se ordenan de mejor a peor.</p>  |
| <p><b>Pregunta 6</b></p> <p>Correcta</p> <p>Puntúa como 1,00</p> <p>🚩 Marcar pregunta</p> | <p>Si ante un problema de decisión existe un criterio de selección voraz entonces ...</p> <p>Seleccione una:</p> <p><input type="radio"/> a. ... la solución óptima está garantizada.</p> <p><input checked="" type="radio"/> b. Ninguna de las otras dos opciones es cierta. ✓</p> <p><input type="radio"/> c. ... al menos una solución factible está garantizada.</p>   |
| <p><b>Pregunta 7</b></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 &lt; 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 checked="" type="radio"/> a. int A[] ✓</p> <p><input type="radio"/> b. int A[][]</p> <p><input type="radio"/> c. int A</p>   |
| <p><b>Pregunta 8</b></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 y, unsigned x){ // suponemos y &gt;= x     if (x==0    y==x) return 1;     return f(y-1, x-1) + f(y-1, x); }</pre> <p>¿Cuál es la mejor complejidad espacial que se puede conseguir?</p> <p>Seleccione una:</p> <p><input type="radio"/> a. <math>O(1)</math></p> <p><input checked="" type="radio"/> b. <math>O(y)</math> ✓</p> <p><input type="radio"/> c. <math>O(y^2)</math></p>   |

|   |   |
|---|---|
| <p><b>Pregunta 9</b></p> <p>Incorrecta</p> <p>Puntuación 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> <ul style="list-style-type: none"> <li><input type="radio"/> a. La mochila discreta y la asignación de tareas.</li> <li><input checked="" type="radio"/> b. El fontanero diligente y el problema del cambio. ✖</li> <li><input type="radio"/> c. La mochila continua y la asignación de tareas.</li> </ul>   |
| <p><b>Pregunta 10</b></p> <p>Correcta</p> <p>Puntuación como 1,00</p> <p>▼ Marcar pregunta</p>  | <p>Supongamos que una solución recursiva a un problema de optimización muestra estas dos características: por un lado, se basa en obtener soluciones óptimas a problemas parciales más pequeños; por otro lado, los subproblemas se resuelven más de una vez durante el proceso recursivo. Este problema es candidato a tener una solución alternativa basada en ...</p> <p>Seleccione una:</p> <ul style="list-style-type: none"> <li><input type="radio"/> a. ... un algoritmo del estilo de <i>divide y vencerás</i>.</li> <li><input checked="" type="radio"/> b. ... un algoritmo de programación dinámica. ✔</li> <li><input type="radio"/> c. ... un algoritmo voraz.</li> </ul> |
| <p><b>Pregunta 11</b></p> <p>Correcta</p> <p>Puntuación como 1,00</p> <p>▼ Marcar pregunta</p>  | <p>¿Cuál de estas tres estrategias voraces obtiene un mejor valor para la mochila discreta?</p> <p>Seleccione una:</p> <ul style="list-style-type: none"> <li><input checked="" type="radio"/> a. Meter primero los elementos de mayor valor específico o valor por unidad de peso. ✔</li> <li><input type="radio"/> b. Meter primero los elementos de menor peso.</li> <li><input type="radio"/> c. Meter primero los elementos de mayor valor.</li> </ul>   |
| <p><b>Pregunta 12</b></p> <p>Correcta</p> <p>Puntuación como 1,00</p> <p>▼ Marcar pregunta</p>  | <p>La solución de programación dinámica iterativa del problema de la mochila discreta ...</p> <p>Seleccione una:</p> <ul style="list-style-type: none"> <li><input checked="" type="radio"/> a. ... tiene la restricción de que los valores tienen que ser enteros positivos. ✔</li> <li><input type="radio"/> b. ... tiene un coste temporal asintótico exponencial con respecto al número de objetos.</li> <li><input type="radio"/> c. ... calcula menos veces el valor de la mochila que la correspondiente solución de programación dinámica recursiva.</li> </ul>   |