

PRIMER PARCIAL

Pregunta 1

Correcta

Puntúa como 1,00

🚩 Marcar pregunta

Un programa con dos bucles anidados uno dentro del otro, El primero hace n iteraciones aproximadamente y el segundo la mitad, tarda un tiempo

Seleccione una:

- ☐ a. $O(n \log n)$
- ☐ b. $O(n\sqrt{n})$
- ☒ c. $O(n^2)$ ✓

Pregunta 2

Correcta

Puntúa como 1,00

🚩 Marcar pregunta

La versión de *Quicksort* que utiliza como pivote la mediana del vector ...

Seleccione una:

- ☒ a. ... El hecho de que el vector estuviera previamente ordenado o no, no influye en la complejidad temporal de este algoritmo. ✓
- ☐ b. ... se comporta mejor cuando el vector ya está ordenado.
- ☐ c. ... se comporta peor cuando el vector ya está ordenado.

Pregunta 3

Correcta

Puntúa como 1,00

🚩 Marcar pregunta

Indica cuál es la complejidad en el peor caso de la función `replace`:

```
unsigned bound( const vector<int> &v ) {  
    for( unsigned i = 0; i < v.size(); i++ )  
        if( v[i] == '0' )  
            return i;  
    return v.size();  
}  
  
void replace( vector<int>& v, int c ) {  
    for( unsigned i = 0; i < bound(v); i++)  
        v[i] = c;  
}
```

Seleccione una:

- ☐ a. $O(n \log n)$
- ☒ b. $O(n^2)$ ✓
- ☐ c. $O(n)$

Pregunta 4

Incorrecta

Puntúa como 1,00

🚩 Marcar pregunta

Dada la siguiente relación de recurrencia, ¿Qué cota es verdadera?

$$f(n) = \begin{cases} 1 & n=1 \\ \sqrt{n} + 3f(n/3) & n>1 \end{cases}$$

Seleccione una:

- ☒ a. $f(n) \in \Theta(\sqrt{n} \log n)$ ✗
- ☐ b. $f(n) \in \Theta(n^3)$
- ☐ c. $f(n) \in \Theta(n)$

Pregunta 5

Correcta

Puntúa como 1,00

🚩 Marcar pregunta

Un algoritmo recursivo basado en el esquema *divide y vencerás* ...

Seleccione una:

- ☐ a. ... nunca tendrá una complejidad exponencial.
- ☐ b. Las demás opciones son verdaderas.
- ☒ c. ... será más eficiente cuanto más equitativa sea la división en subproblemas. ✓

Pregunta 6

Correcta

Puntúa como 1,00

🚩 Marcar pregunta

Un problema de tamaño n puede transformarse en tiempo $O(n^3)$ en ocho de tamaño $n/2$; por otro lado, la solución al problema cuando la talla es 1 requiere un tiempo constante.

¿cual de estas clases de coste temporal asintótico es la más ajustada?

Seleccione una:

- ☒ a. $O(n^3 \log n)$ ✓
- ☐ b. $O(n^2 \log n)$
- ☐ c. $O(n^3)$

Pregunta 7

Sin contestar

Puntúa como 1,00

🚩 Marcar pregunta

Indicad cuál de estas tres expresiones es cierta:

Seleccione una:

- ☐ a. $O(2^{\log(n)}) \subset O(n^2) \subset O(2^n)$
- ☐ b. $O(n^2) \subset O(2^{\log(n)}) \subseteq O(2^n)$
- ☐ c. $O(n^2) \subset O(2^{\log(n)}) \subset O(2^n)$

Pregunta 8

Correcta

Puntúa como 1,00

🚩 Marcar pregunta

¿Cuál es la complejidad temporal de la siguiente función recursiva?

```
unsigned desperdicio (unsigned n){
    if (n<=1)
        return 0;
    unsigned sum = desperdicio (n/2) + desperdicio (n/2) + desperdicio (n/2);
    for (unsigned i=1; i<n-1; i++)
        for (unsigned j=1; j<=i; j++)
            for (unsigned k=1; k<=j; k++)
                sum+=i*j*k;
    return sum;
}
```

Seleccione una:

- ☒ a. $\Theta(n^3)$ ✓
- ☐ b. $\Theta(3^n)$
- ☐ c. $\Theta(n^3 \log n)$

Pregunta 9

Correcta

Puntúa como 1,00

🚩 Marcar pregunta

Indica cuál es la complejidad en función de n , donde k_c es una constante (no depende de n), del fragmento siguiente :

```
for( int i = k; i < n - k; i++){
    A[i] = 0;
    for( int j = i - k; j < i + k; j++ )
        A[i] += B[j];
}
```

Seleccione una:

- ☐ a. $O(n \log n)$
- ☒ b. $O(n)$ ✓
- ☐ c. $O(n^2)$

Pregunta 10

Correcta

Puntúa como 1,00

🚩 Marcar pregunta

Pertenece $3n^2 + 3$ a $O(n^3)$?

Seleccione una:

- ☐ a. Sólo para $c = 1$ y $n_0 = 5$.
- ☒ b. Sí. ✓
- ☐ c. No.

Pregunta 11

Correcta

Puntúa como 1,00

🚩 Marcar pregunta

Indica cuál es la complejidad, en función de n , del siguiente fragmento de código:

```
s=0; for(i=0;i<n;i++) for(j=i;j<n;j++) s+=i*j;
```

Seleccione una:

- ☒ a. $\Theta(n^2)$ ✓
- ☐ b. $O(n^2)$ pero no $\Omega(n^2)$.
- ☐ c. $\Theta(n)$

Pregunta 12

Incorrecta

Puntúa como 1,00

🚩 Marcar pregunta

Sea $f(n)$ la solución de la relación de recurrencia $f(n) = 2f(n/2) + n$; $f(1) = 1$. Indica cuál de estas tres expresiones es cierta:

Seleccione una:

- ☐ a. $f(n) \in \Theta(n)$
- ☒ b. $f(n) \in \Theta(n^2)$ ✗
- ☐ c. $f(n) \in \Theta(n \log(n))$

SEGUNDO PARCIAL

<p>Pregunta 1</p> <p>Incorrecta</p> <p>Puntúa como 1,00</p> <p>🚩 Marcar pregunta</p>	<p>En el método voraz ...</p> <p>Seleccione una:</p> <p><input type="radio"/> a. ... es habitual preparar los datos para disminuir el coste temporal de la función que determina cuál es la siguiente decisión a tomar.</p> <p><input type="radio"/> b. ... el dominio de las decisiones sólo pueden ser conjuntos discretos o discretizables.</p> <p><input checked="" type="radio"/> c. ... siempre se encuentra solución pero puede que no sea la óptima. ✖</p>
<p>Pregunta 2</p> <p>Incorrecta</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 $O(n)$ a $O(1)$, 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 $O(n^2)$ a $O(n \log n)$, 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 3</p> <p>Sin contestar</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 complejidad espacial que se puede conseguir?</p> <p>Seleccione una:</p> <p><input type="radio"/> a. $O(x^2)$</p> <p><input type="radio"/> b. $O(x)$</p> <p><input type="radio"/> c. $O(1)$</p>
<p>Pregunta 4</p> <p>Incorrecta</p> <p>Puntúa 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> <p><input type="radio"/> a. Meter primero los elementos de mayor valor específico o valor por unidad de peso.</p> <p><input type="radio"/> b. Meter primero los elementos de mayor valor.</p> <p><input checked="" type="radio"/> c. Meter primero los elementos de menor peso. ✖</p>
<p>Pregunta 5</p> <p>Incorrecta</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 type="radio"/> b. Ninguna de las otras dos opciones es cierta.</p> <p><input checked="" type="radio"/> c. ... al menos una solución factible está garantizada. ✖</p>
<p>Pregunta 6</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 type="radio"/> a. El árbol de cobertura de coste mínimo de un grafo conexo.</p> <p><input checked="" type="radio"/> b. El problema de la mochila discreta o sin fraccionamiento. ✔</p> <p><input type="radio"/> c. El problema de la mochila continua o con fraccionamiento.</p>

Pregunta 7

Correcta

Puntúa como 1,00

🚩 Marcar pregunta

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?

Seleccione una:

- ☐ a. La recursión puede ser infinita y por tanto es necesario organizarla según el esquema iterativo de programación dinámica.
- ☐ b. Se repiten muchos cálculos y ello se puede evitar haciendo uso de una estrategia voraz.
- ☒ c. Se repiten muchos cálculos y ello se puede evitar usando programación dinámica. ✓

Pregunta 8

Correcta

Puntúa como 1,00

🚩 Marcar pregunta

Los algoritmos de programación dinámica hacen uso ...

Seleccione una:

- ☐ a. ... de que la solución óptima se puede construir añadiendo a la solución el elemento óptimo de los elementos restantes, uno a uno.
- ☒ b. ... de que se puede ahorrar cálculos guardando resultados anteriores en un almacén. ✓
- ☐ c. ... de una estrategia trivial consistente en examinar todas las soluciones posibles.

Pregunta 9

Incorrecta

Puntúa como 1,00

🚩 Marcar pregunta

Se pretende implementar mediante programación dinámica iterativa la función recursiva:

```
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;  
}
```

¿Cuál es la mejor estructura para el almacén?

Seleccione una:

- ☐ a. int A[]
- ☒ b. int A[][] ✗
- ☐ c. int A

Pregunta 10

Correcta

Puntúa como 1,00

🚩 Marcar pregunta

De los problemas siguientes, indicad cuál no se puede tratar eficientemente como los otros dos:

Seleccione una:

- ☐ a. El problema del cambio, o sea, el de encontrar la manera de entregar una cantidad de dinero usando el mínimo de monedas posibles.
- ☒ b. El problema de la mochila sin fraccionamiento y sin restricciones en cuanto al dominio de los pesos de los objetos y de sus valores. ✓
- ☐ c. El problema de cortar un tubo de forma que se obtenga el máximo beneficio posible.

Pregunta 11

Incorrecta

Puntúa como 1,00

🚩 Marcar pregunta

Cuando la descomposición recursiva de un problema da lugar a subproblemas de tamaño similar, ¿qué esquema promete ser más apropiado?

Seleccione una:

- ☒ a. El método voraz ✗
- ☐ b. Programación dinámica.
- ☐ c. Divide y vencerás, siempre que se garantice que los subproblemas no son del mismo tamaño.

Pregunta 12

Incorrecta

Puntúa como 1,00

🚩 Marcar pregunta

El problema de encontrar el árbol de recubrimiento de coste mínimo para un grafo no dirigido, conexo y ponderado ...

Seleccione una:

- ☐ a. ... se puede resolver siempre con una estrategia voraz.
- ☒ b. sólo se puede resolver con una estrategia voraz si existe una arista para cualquier par de vértices del grafo. ✗
- ☐ c. ... no se puede resolver en general con una estrategia voraz.

TERCER PARCIAL

Pregunta 1 Correcta Puntúa como 1,00 🚩 Marcar pregunta	<p>La estrategia de <i>ramificación y poda</i> genera las soluciones posibles al problema mediante ...</p> <p>Seleccione una:</p> <ul style="list-style-type: none"><input checked="" type="radio"/> a. ... un recorrido guiado por estimaciones de las mejores ramas del árbol que representa el espacio de soluciones. ✓<input type="radio"/> b. ... un recorrido en profundidad del árbol que representa el espacio de soluciones.<input type="radio"/> c. ... un recorrido en anchura del árbol que representa el espacio de soluciones.
Pregunta 2 Correcta Puntúa como 1,00 🚩 Marcar pregunta	<p>En los algoritmos de <i>ramificación y poda</i>, ¿el valor de una cota pesimista es mayor que el valor de una cota optimista? (entendiendo que ambas cotas se aplican sobre el mismo nodo)</p> <p>Seleccione una:</p> <ul style="list-style-type: none"><input type="radio"/> a. No, nunca es así.<input type="radio"/> b. En general sí, si se trata de un problema de maximización, aunque en ocasiones ambos valores pueden coincidir.<input checked="" type="radio"/> c. En general sí, si se trata de un problema de minimización, aunque en ocasiones ambos valores pueden coincidir. ✓
Pregunta 3 Correcta Puntúa como 1,00 🚩 Marcar pregunta	<p>Decid cuál de estas tres es la cota pesimista más ajustada al valor óptimo de la mochila discreta:</p> <p>Seleccione una:</p> <ul style="list-style-type: none"><input type="radio"/> a. El valor de una mochila que contiene todos los objetos aunque se pase del peso máximo permitido.<input type="radio"/> b. El valor de la mochila continua correspondiente.<input checked="" type="radio"/> c. El valor de la mochila discreta que se obtiene usando un algoritmo voraz basado en el valor específico de los objetos. ✓
Pregunta 4 Incorrecta Puntúa como 1,00 🚩 Marcar pregunta	<p>Al resolver el problema del viajante de comercio mediante <i>vuelta atrás</i>, ¿cuál de estas cotas optimistas se espera que puede mejorar el árbol de búsqueda?</p> <p>Seleccione una:</p> <ul style="list-style-type: none"><input checked="" type="radio"/> a. Se resuelve el resto del problema usando un algoritmo voraz que añade cada vez al camino el vértice más cercano al último añadido. ✗<input type="radio"/> b. Se multiplica k_c por la distancia de la arista más corta que nos queda por considerar, donde k_c es el número de saltos que nos quedan por dar.<input type="radio"/> c. Se ordenan las aristas restantes de menor a mayor distancia y se calcula la suma de las k_c aristas más cortas, donde k_c es el número de saltos que nos quedan por dar.
Pregunta 5 Incorrecta Puntúa como 1,00 🚩 Marcar pregunta	<p>En los algoritmos de <i>ramificación y poda</i> ...</p> <p>Seleccione una:</p> <ul style="list-style-type: none"><input type="radio"/> a. Una cota pesimista es el beneficio esperado de cualquier nodo factible que no es el óptimo.<input checked="" type="radio"/> b. Una cota optimista es necesariamente un valor alcanzable, de no ser así no está garantizado que se encuentre la solución óptima. ✗<input type="radio"/> c. Una cota optimista es necesariamente un valor insuperable, de no ser así se podría podar el nodo que conduce a la solución óptima.
Pregunta 6 Correcta Puntúa como 1,00 🚩 Marcar pregunta	<p>Se desea obtener todas las permutaciones de una lista compuesta por n elementos. ¿Qué esquema es el más adecuado?</p> <p>Seleccione una:</p> <ul style="list-style-type: none"><input type="radio"/> a. <i>Ramificación y poda</i>, puesto que con buenas funciones de cota es más eficiente para este problema que <i>vuelta atrás</i>.<input checked="" type="radio"/> b. <i>Vuelta atrás</i>, para este problema no hay un esquema más eficiente. ✓<input type="radio"/> c. <i>Divide y vencerás</i>, puesto que la división en sublistas se podría hacer en tiempo constante.

Pregunta 7 Incorrecta Puntúa como 1,00 🚩 Marcar pregunta	<p>Se desea encontrar el camino mas corto entre dos ciudades. Para ello se dispone de una tabla con la distancia entre los pares de ciudades en los que hay carreteras o un valor centinela (por ejemplo, -1) si no hay, por lo que para ir de la ciudad inicial a la final es posible que haya que pasar por varias ciudades. También se conocen las coordenadas geográficas de cada ciudad y por tanto la distancia geográfica (en línea recta) entre cada par de ciudades. Para limitar la búsqueda en un algoritmo de <i>vuelta atrás</i>, se utiliza la solución de un algoritmo <i>voraz</i> basado en moverse en cada paso a la ciudad, de entre las posibles según el mapa de carreteras, que esté más cercana al destino según su distancia geográfica. Este algoritmo voraz, ¿serviría como cota pesimista?</p> <p>Seleccione una:</p> <p><input type="radio"/> a. No, ya que en algunos casos puede dar distancias menores que la óptima.</p> <p><input type="radio"/> b. No, ya que no asegura que se encuentre una solución factible.</p> <p><input checked="" type="radio"/> c. Sí, puesto que la distancia geográfica asegura que otra solución mejor no es posible. X</p>
Pregunta 8 Incorrecta Puntúa como 1,00 🚩 Marcar pregunta	<p>El uso de funciones de cota en ramificación y poda ...</p> <p>Seleccione una:</p> <p><input type="radio"/> a. ... garantiza que el algoritmo va a ser más eficiente ante cualquier instancia del problema.</p> <p><input checked="" type="radio"/> b. ... transforma en polinómicas complejidades que antes eran exponenciales. X</p> <p><input type="radio"/> c. ... puede reducir el número de instancias del problema que pertenecen al caso peor.</p>
Pregunta 9 Correcta Puntúa como 1,00 🚩 Marcar pregunta	<p>Cuando se resuelve usando un algoritmo de <i>vuelta atrás</i> un problema de n decisiones, en el que siempre hay como mínimo dos opciones para cada decisión, ¿cuál de las siguientes complejidades en el caso peor es la mejor que nos podemos encontrar?</p> <p>Seleccione una:</p> <p><input type="radio"/> a. $O(n!)$</p> <p><input type="radio"/> b. $O(n^2)$</p> <p><input checked="" type="radio"/> c. $O(2^n)$ ✓</p>
Pregunta 10 Sin contestar Puntúa como 1,00 🚩 Marcar pregunta	<p>La complejidad en el mejor de los casos de un algoritmo de <i>ramificación y poda</i>...</p> <p>Seleccione una:</p> <p><input type="radio"/> a. ... es siempre exponencial con el número de decisiones a tomar.</p> <p><input type="radio"/> b. ... suele ser polinómica con el número de alternativas por cada decisión</p> <p><input type="radio"/> c. ... puede ser polinómica con el número de decisiones a tomar.</p>
Pregunta 11 Incorrecta Puntúa como 1,00 🚩 Marcar pregunta	<p>¿Cuál es la diferencia principal entre una solución de <i>vuelta atrás</i> y una solución de <i>ramificación y poda</i> para el problema de la mochila?</p> <p>Seleccione una:</p> <p><input type="radio"/> a. El orden de exploración de las soluciones.</p> <p><input checked="" type="radio"/> b. El hecho que la solución de <i>ramificación y poda</i> puede empezar con una solución subóptima voraz y la de <i>vuelta atrás</i> no. X</p> <p><input type="radio"/> c. El coste asintótico en el caso peor.</p>
Pregunta 12 Incorrecta Puntúa como 1,00 🚩 Marcar pregunta	<p>Dado un problema de optimización cualquiera, ¿la estrategia de <i>vuelta atrás</i> garantiza la solución óptima?</p> <p>Seleccione una:</p> <p><input type="radio"/> a. Es condición necesaria que el dominio de las decisiones sea discreto o discretizable y que el número de decisiones a tomar esté acotado.</p> <p><input checked="" type="radio"/> b. Sí, puesto que ese método analiza todas las posibilidades. X</p> <p><input type="radio"/> c. Sí, siempre que el dominio de las decisiones sea discreto o discretizable y además se empleen mecanismos de poda basados en la mejor solución hasta el momento.</p>