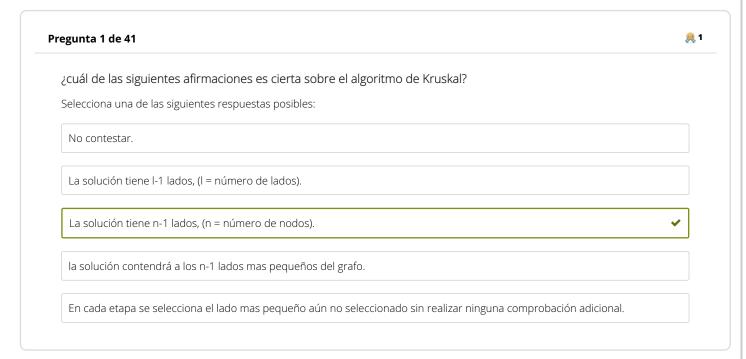
WUOLAH

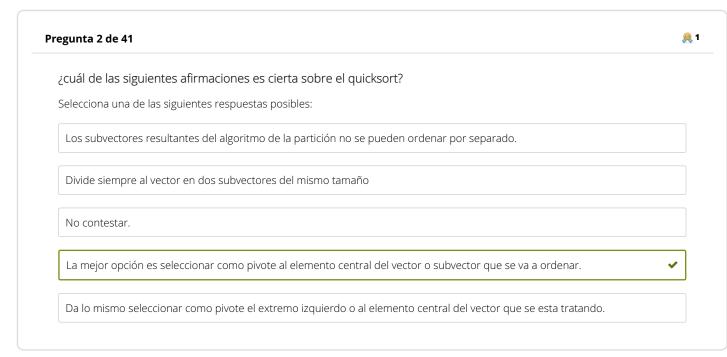


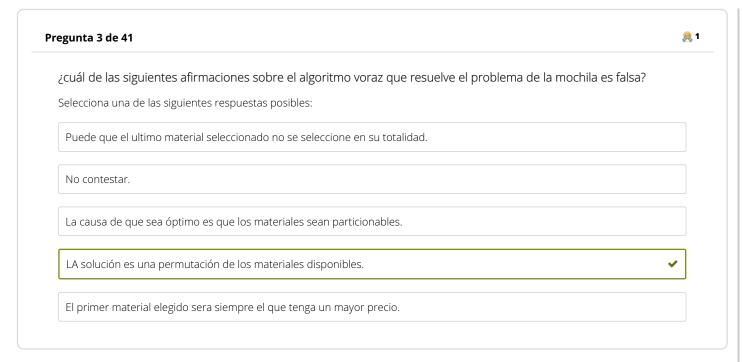
39742

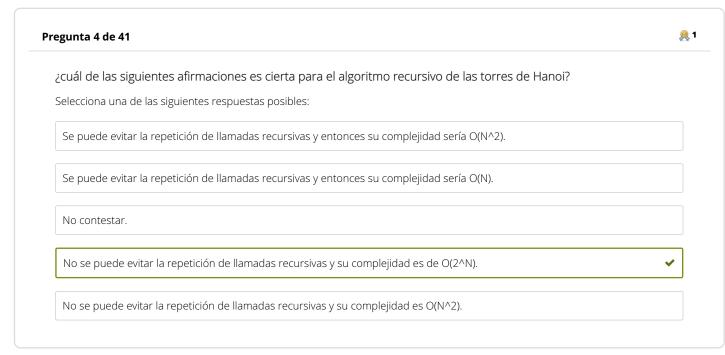
AL 4-5-6 _ Test.pdf Test 2° Bloque Temas 4.5.6

- 3° Algorítmica
- Escuela Politécnica Superior de Córdoba UCO Universidad de Córdoba

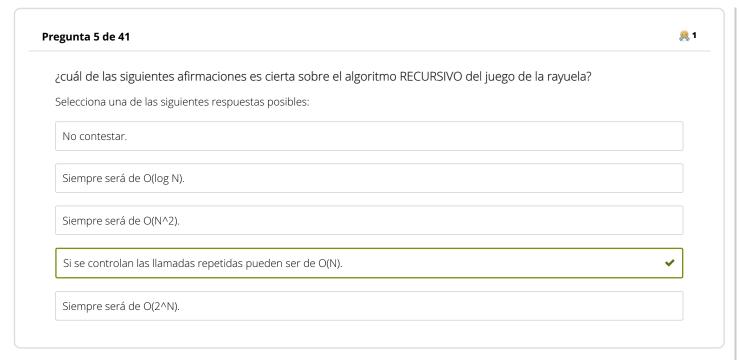


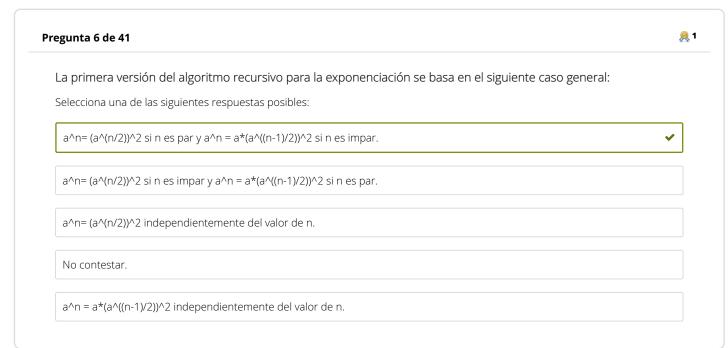






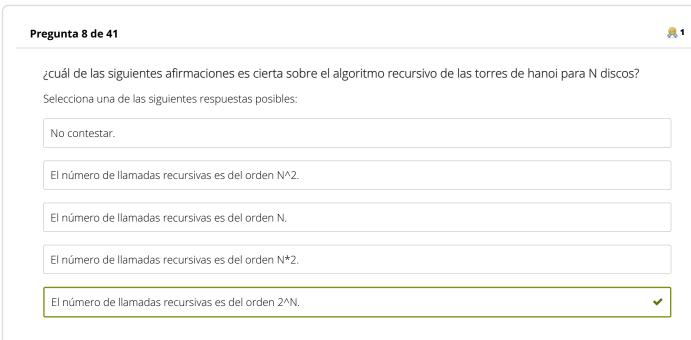




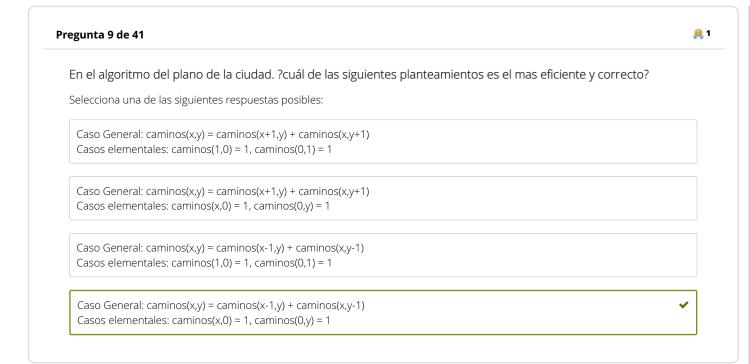


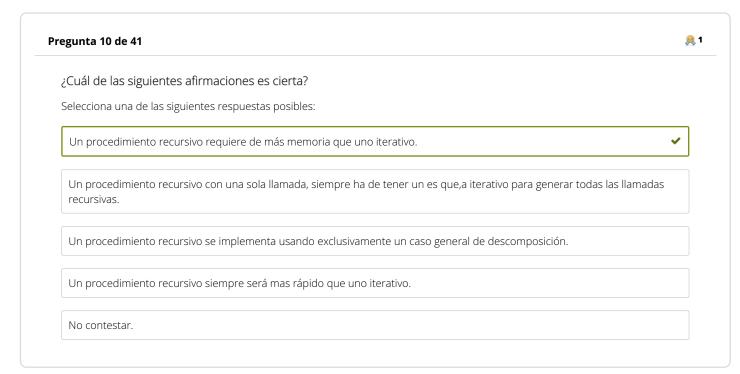


egunta 7 de 41	
¿Cuál de las siguientes afirmaciones es cierta?	
Selecciona una de las siguientes respuestas posibles:	
No contestar.	
Si tenemos un algoritmo recursivo, sólo se debe de traducir a iterativo en caso de que la versión iterativa sea m eficiente.	ucho mas 🗸
Un algoritmo recursivo que posea un solo esquema condicional con una cosa llamada recursiva puede ser de o exponencial.	rden
Todo algoritmo recursivo hace uso del montón (heap) para almacenar las llamadas recursivas.	
Siempre hay que buscar una versión iterativa a la hora de resolver cualquier tipo de problema, ya que ésta siem más simple de obtener.	pre será mucho











Selecciona una de las siguientes respuestas posibles:

Para comprobar si un conjunto de tareas es factible, basta comprobar si la permutación que contiene a las tareas en orden creciente de plazos es factible.

Para comprobar si un conjunto de tareas es factible, basta comprobar si una permutación aleatoria de ese conjunto es factible.

No contestar.

Para comprobar si un conjunto de tareas es factible, es necesario comprobar todas las permutaciones posibles de ese conjunto de tareas.

Para comprobar si un conjunto de tareas es factible, basta comprobar si la permutación que contiene a las tareas en orden decreciente de beneficios es factible.

Pregunta 12 de 41

<u>,</u> 1

¿Cuál de las siguientes condiciones no tiene porqué cumplirse a la hora de aplicar el método de divide y vencerás.? Selecciona una de las siguientes respuestas posibles:

Se ha de seleccionar de forma conveniente cuando se usa el algoritmo básico en vez de seguir descomponiendo el problema.

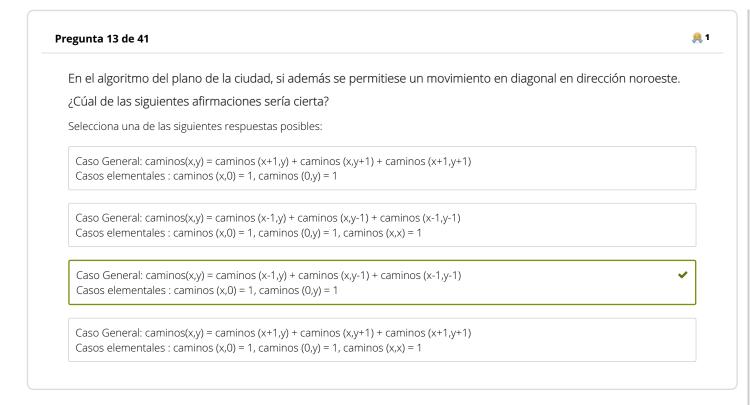
Se aplica en problemas de optimización.

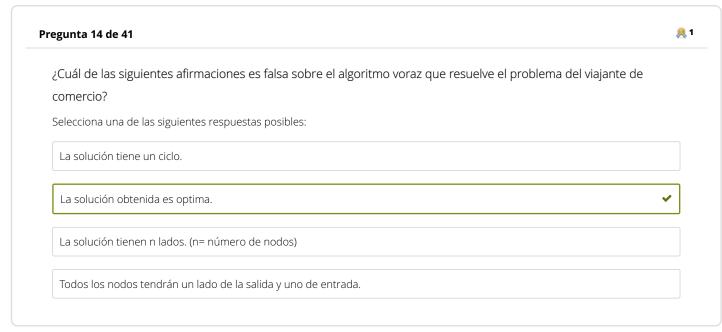
Ha de ser posible la descomposición del problema en subproblemas y recomponer las soluciones de estos subproblemas de manera eficiente.

Los subproblemas han de ser aproximadamente del mismo tamaño.

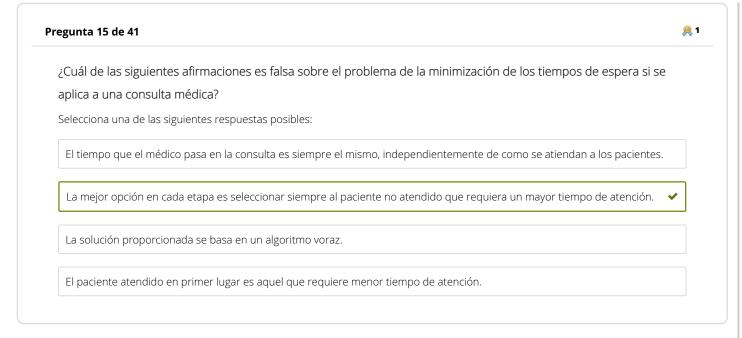


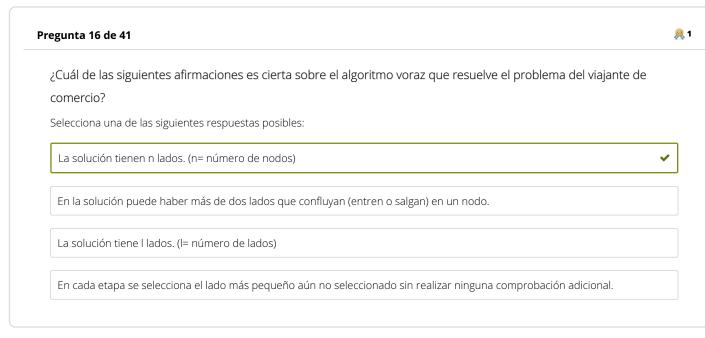
No contestar.



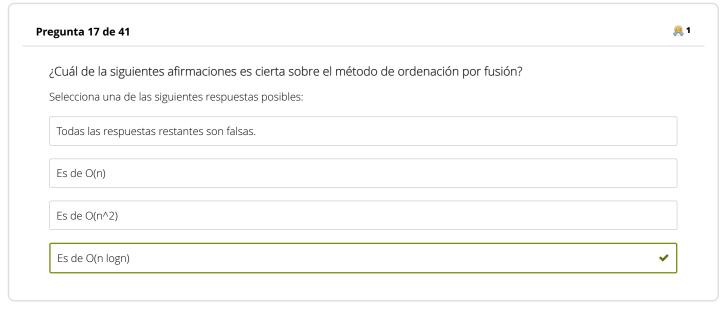












Pregunta 18 de 41

¿Cuál de las siguientes afirmaciones sobre el algoritmo voraz que resuelve el problema de la mochila es cierta?

Selecciona una de las siguientes respuestas posibles:

Aunque haya materiales suficientes, puede que la mochila no se llene totalmente.

Solo proporcionan una solución óptima cuando los materiales son particionables.

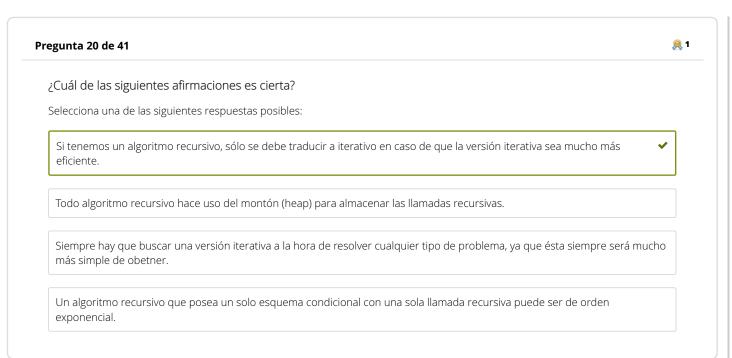
✓

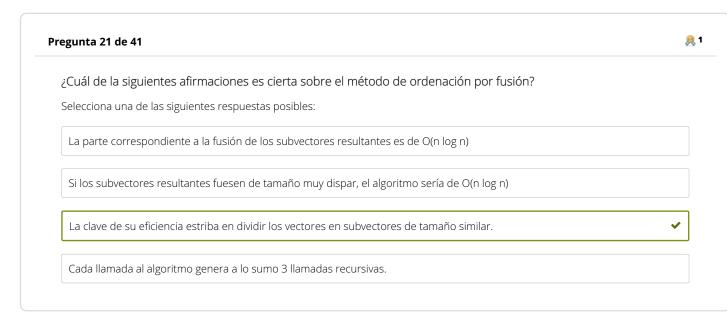
Todos los materiales seleccionados, se seleccionan en su totalidad.

Puede que haya más de un material seleccionado que no se seleccione en su totalidad.

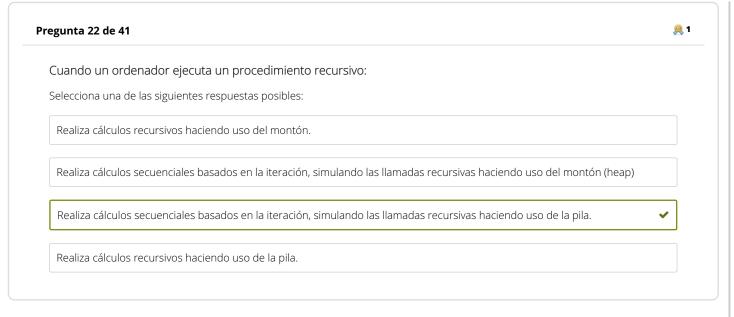


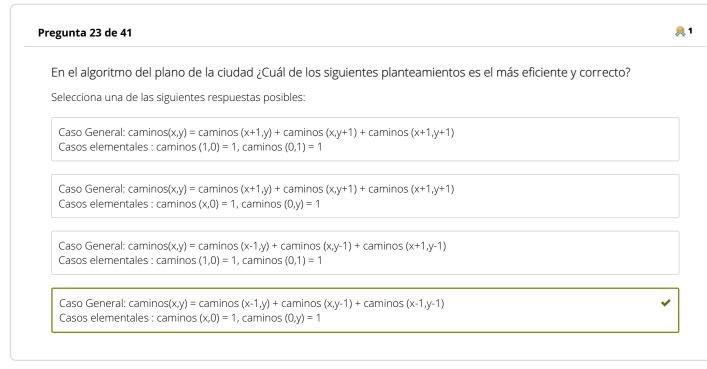






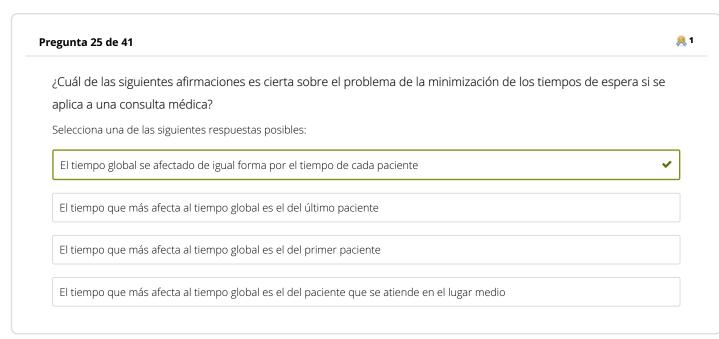




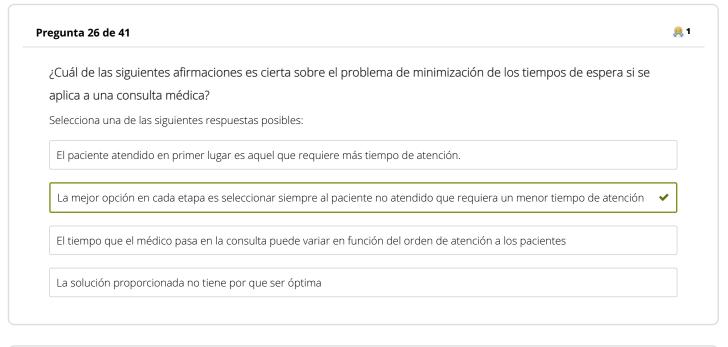


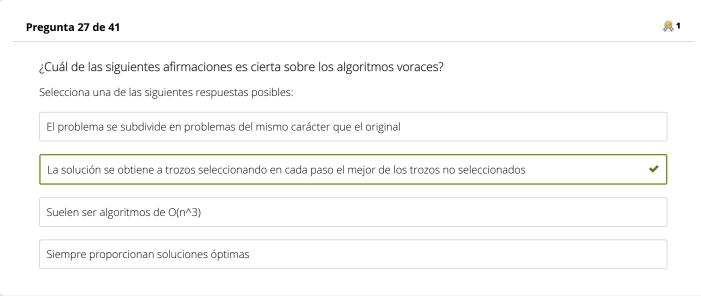


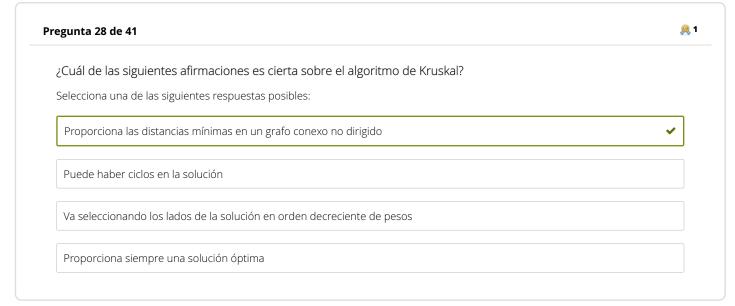




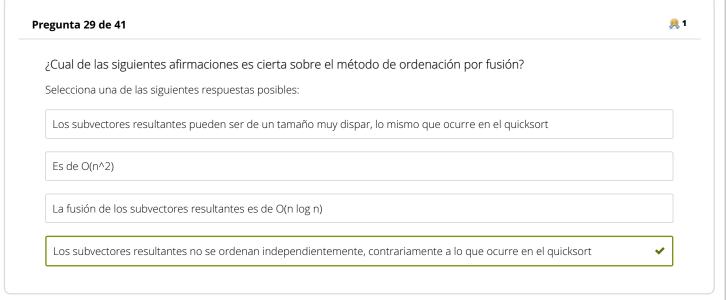


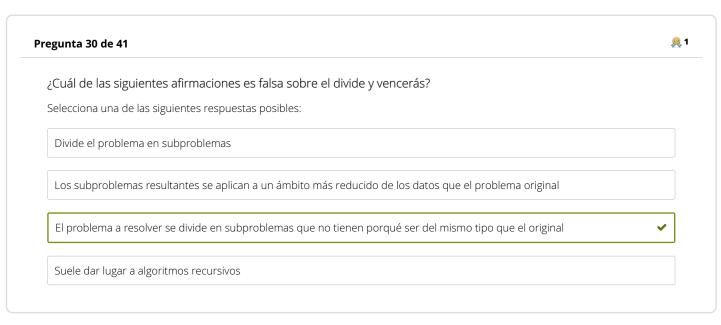


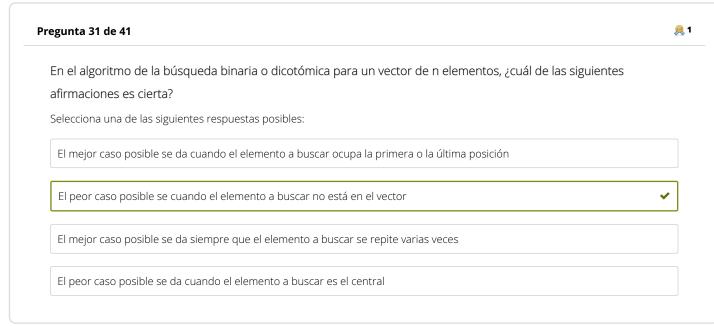




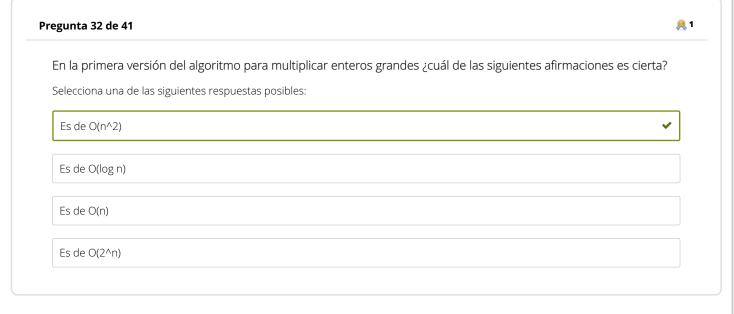


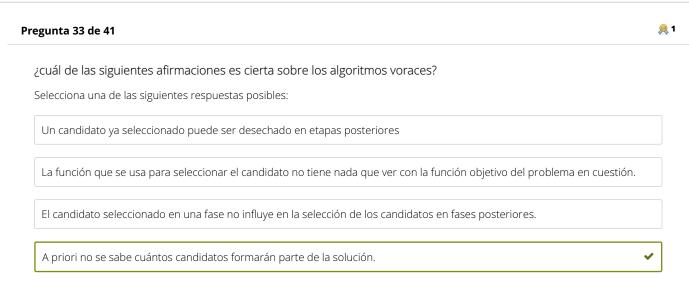


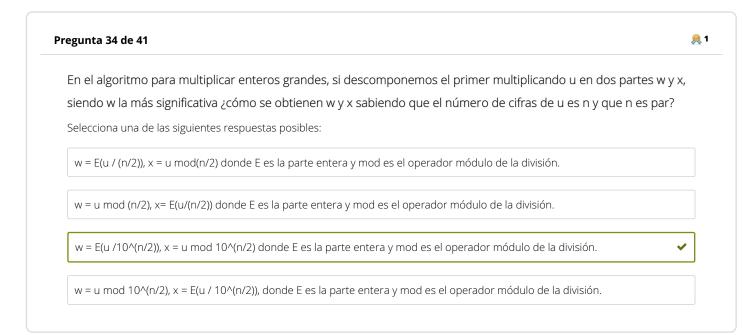




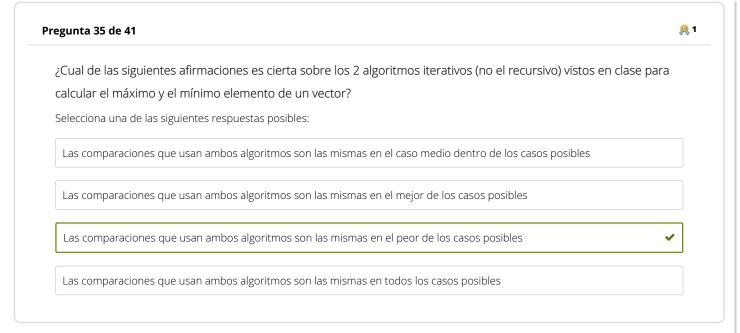


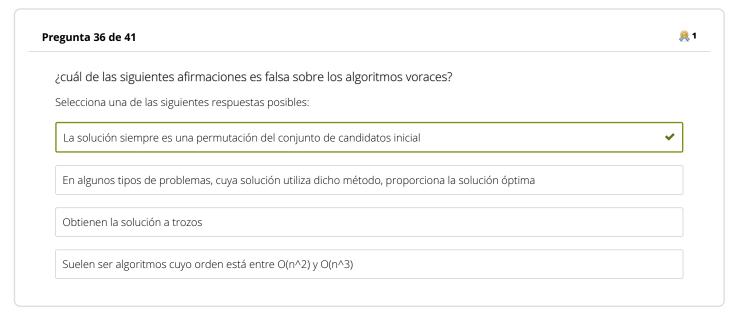














Pregunta 37 de 41

<u>,</u> 1

¿Cual de las siguientes afirmaciones es cierta sobre el algoritmo que resuelve el problema de la planificación de tareas a plazo fijo?

Selecciona una de las siguientes respuestas posibles:

para comprobar si un conjunto de tareas es factible, es necesario comprobar todas las permutaciones posibles de ese conjunto de tareas

para comprobar si un conjunto de tareas es factible, basta comprobar si una permutación aleatoria de ese conjunto es factible

para comprobar si un conjunto de tareas es factible, basta comprobar si la permutación que contiene a las tareas en orden creciente de plazos es factible

para comprobar si un conjunto de tareas es factible, basta comprobar si la permutación que contiene a las tareas en orden decreciente de beneficios es factible

Pregunta 38 de 41



En un algoritmo recursivo la clave de su diseño es:

Selecciona una de las siguientes respuestas posibles:

Descomponer el problema en subproblemas, que no tienen porqué ser similares al de partida, para los mismos datos del problema original

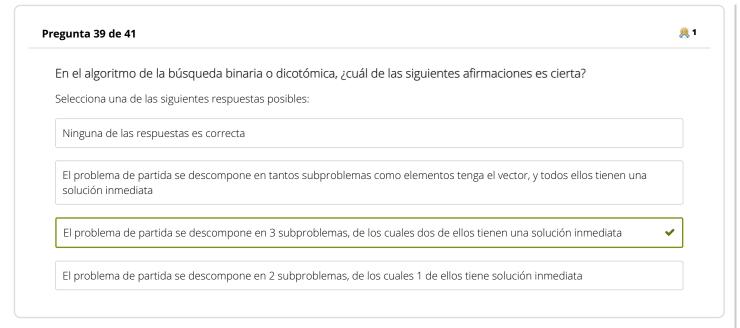
Descomponer el problema en subproblemas similares al de partida para los mismos datos del problema original

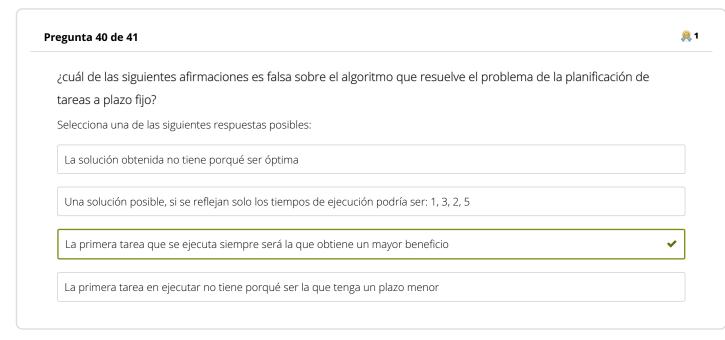
Descomponer el problema en subproblemas similares al de partida para datos más simples que los del problema original



Descomponer el problema en subproblemas, que no tienen porqué ser similares al de partida, para datos más simples que los del problema original









En todo a	lgoritmo recursivo que contenga dos llamadas recursivas, como por ejemplo el de la sucesión de Fibona
o el de la	torres de Hanoi:
Selecciona	una de las siguientes respuestas posibles:
Siempre	se producirán llamadas recursivas repetidas que se pueden evitar
Son de c	omplejidad O(N*N)
No siem	ore tienen porqué producirse llamadas recursivas repetidas