

Algoritmos y Estructuras de Datos II, Segundo del Grado de Ingeniería Informática, Test de Programación Dinámica, mayo 2014

Para cada uno de los siguientes dos problemas hay que explicar cómo se pueden resolver por Programación Dinámica, indicando:

- El número de pasos para obtener la solución final, y los subproblemas que se consideran para llegar a esa solución.
- La ecuación de recurrencia que relaciona la solución para un tamaño y número de pasos en función de tamaños y número de pasos menores, y cuáles son y los valores de los casos base.
- Las estructuras de datos (tablas, vectores...) a utilizar para guardar las subsoluciones y cómo se rellenan estas estructuras.
- Dónde se encuentra la solución final o cómo se recompone a partir de la información de las tablas.
- Una estimación (cuanto más precisa mejor) del tiempo de ejecución del método de resolución del problema por Programación Dinámica.
- Explicar cómo funciona sobre un ejemplo de tamaño reducido el método de resolución ideado.

1) Dada una cadena de números naturales, se trata de encontrar la subcadena creciente más larga de números pares o impares. Si hay varias de la misma longitud sólo hay que devolver una de ellas. Una subcadena no tiene obligatoriamente que estar formada por números en posiciones consecutivas de la cadena original, y no se consideran repeticiones en la subcadena. Por ejemplo, dada la cadena 4,7,3,5,8,6,9,2,5,3,6, las siguientes son dos subcadenas válidas 4,7,3,5,8,6,9,2,5,3,6 y 4,7,3,5,8,6,9,2,5,3,6.

2) Tenemos una mochila con capacidad C , y n objetos con beneficios b_i y pesos p_i , y algunos de los objetos se pueden dividir en 2 o 4 partes. El número de partes en que se puede dividir cada objeto viene dado por d_i , con valor 1 (no se puede partir), 2 o 4. Al tomar una parte de un objeto se obtiene el beneficio y el peso proporcional a la parte tomada. Por ejemplo, si un objeto se puede particionar en cuatro partes, se puede no tomar nada del objeto, se puede tomar $1/4$, $2/4$, $3/4$ o 1, y si se toma una cantidad c el beneficio que se obtiene es $c * b_i$ y el peso $c * p_i$. Se trata de obtener el beneficio máximo que se puede obtener incluyendo objetos o partes de ellos en la mochila sin exceder su capacidad. Hay que calcular el beneficio máximo y la parte de cada objeto que se introduce para obtenerlo.