

PROGRAMACIÓN Y ESTRUCTURAS DE DATOS AVANZADAS

Febrero 2016 (Segunda semana)

Normas de valoración del examen:

- La nota del examen representa el 80% de la valoración final de la asignatura (el 20% restante corresponde a las prácticas).
- Cada cuestión contestada correctamente vale 1 punto.
- Cada cuestión contestada incorrectamente baja la nota en 0.3 puntos.
- Debe obtenerse un mínimo de 3 puntos en las cuestiones para que el problema sea valorado (con 3 cuestiones correctas y alguna incorrecta el examen está suspenso).
- La nota total del examen debe ser al menos de 4.5 para aprobar.
- Las cuestiones se responden en una hoja de lectura óptica.

SOLUCIONES:

Test:

Tipo A: 1C 2D 3C 4D 5C 6D

Tipo B: 1D 2C 3C 4D 5C 6D

Problema (4 puntos).

Se dispone de n cubos identificados con un número de 1 al n . Cada cubo tiene impresa en una de sus caras una letra distinta. Se indica además una palabra de n letras. Se trata de colocar los n cubos uno a continuación otro, de forma que con esa disposición se pueda formar la palabra dada. Como en diferentes cubos puede haber letras repetidas, la solución puede no ser única o no existir.

La resolución del problema debe incluir, por este orden:

1. Elección del esquema más apropiado, el esquema general y explicación de su aplicación al problema (0,5 puntos)
2. Descripción de las estructuras de datos necesarias (0,5 puntos solo si el punto 1 es correcto)
3. Algoritmo completo a partir del refinamiento del esquema general (2,5 puntos solo si el punto 1 es correcto). Si se trata del esquema voraz, debe realizarse la demostración de optimalidad. Si se trata del esquema de programación dinámica, deben proporcionarse las ecuaciones de recurrencia.
4. Estudio del coste del algoritmo desarrollado (0,5 puntos solo si el punto 1 es correcto)

Solución:

1. El esquema más apropiado es el de vuelta atrás. El esquema general se encuentra formulado en el libro de texto de la asignatura, en la página 161.

2. Los cubos vienen dados por un vector $C[1..n]$, donde el cubo $C[i]$ se representa como un vector $C[i][1..6]$ que denota las letras en las 6 caras del cubo. La palabra viene dada por un vector $P[1..n]$ de letras.

Podemos representar la solución como una tupla (x_1, \dots, x_n) de pares (i, j) indicando que se considera la letra de la cara j del cubo $C[i]$. Se tiene que cumplir que en cada posición se utiliza una cara válida de alguno de los cubos, que los cubos no se repitan y que se haya formado la palabra correcta. Utilizamos un vector *usado* $[1..n]$ para saber que cubos ya han sido utilizados

3. La solución completa al problema se encuentra desarrollada en:

Estructuras de datos y métodos algorítmicos
N. Martí Oliet, Y Ortega Mallén y J.A. Verdejo López
Prentice Hall
Página 463

4. El árbol de exploración tiene n niveles y el grado del árbol es $6n$, por lo que el tamaño del espacio de búsqueda tiene como cota superior $O(6n)^n$. Una cota más ajustada sería $O(6^n(n!))$.