Analisis de algoritmos 2022-1

Tarea 2

Fecha de entrega: viernes 29 de octubre de 2021

1 Ejercicios

- Peso: 10 puntos. Considera el problema de la *mochila fraccional.* Entrada: una capacidad de mochila C y una secuencia de n parejas $item_i = (v_i, w_i)$, de tal forma que el elemento iésimo tiene valor total v_i, y su peso total es w_i. Salida: un vector $q_1, ..., q_n$, con entradas en $[0, 1]$, que indican las fracciones del total de los respectivos elementos, de tal forma que:

  - La suma de los pesos de las fracciones de los elementos es compatible con la capacidad de la mochila ($\sum_{i=1}^{n} q_i \cdot w_i \leq C$), y
  - La suma de las fracciones de los valores de los elementos ($\sum_{i=1}^{n} q_i \cdot v_i \leq V$) donde V sea el mayor valor total posible.

  Por simplicidad, y para que la solución sea única, supón que el valor unitario (v_i/w_i) de cada elemento es único.

  Para este problema:

  1. Enuncia y demuestra un lema de intercambio, que te permita transformar toda solución no óptima A en una solución de mejor valor A'.
  2. Demuestra con el lema anterior, que la solución es única (usando la hipótesis de que los valores unitarios no se repiten).
  3. Enuncia y demuestra un lema que caracterice la estructura de la solución única (que es la única a la que ya no se le puede aplicar un intercambio que la mejore).
  4. Propón un algoritmo que, apoyandose en el lema anterior, dada una instancia del problema, construya la solución óptima, en tiempo $O(n \log n)$.
     - Demuestra que el algoritmo es correcto.
     - Demuestra que su complejidad es la requerida.

- Ejercicio Opcional. Peso 5 puntos extra. Máximo número de intervalos compatibles. Entrada: un conjunto de n intevalos abiertos $S = \{(b_i, e_i), i \in 1, \ldots, n\}$. Salida: un subconjunto de tamaño máximo posible de intervalos compatibles. Dos intervalos son compatibles si sus interiores no se intersecan. Ejemplo: $S = \{(1,3),(2,5),(4,6)\}$. Salida: $\{(1,3),(4,6)\}$. Observa que en general, la solución óptima no es única.

  Para este problema:

  1. Enuncia y demuestra un lema de intercambio, que te permita transformar toda solución óptima, en una que tenga estructura más controlada, sin empeorar su optimalidad.
  2. Enuncia y demuestra un lema que caracterice la estructura de la solución final a la que eventualmente se transforma cualquier solución óptima (que es la única a la que ya no se le puede aplicar el intercambio del paso anterior).
  3. Propón un algoritmo que, apoyandose en el lema anterior, dada una instancia del problema, construya la solución óptima, en tiempo $O(n \log n)$.
     - Demuestra que el algoritmo es correcto.
     - Demuestra que su complejidad es la requerida.