Optimización - 2013 - Práctica 3

Golosos

1. (Fracciones Egipcias.) Encontrar una representación de un racional positivo como suma de racionales de numerador 1 y denominadores positivos distintos. Por ejemplo $\frac{7}{15} = \frac{1}{3} + \frac{1}{8} + \frac{1}{120}$, pero para $\frac{2}{3}$ no sirve poner $\frac{1}{3} + \frac{1}{3}$. Implementar una función en Python que resuelva el problema. Demostrar que con el abordaje goloso se encuentra una representación y observar que no es necesariamente la mínima en cantidad de sumandos. Exhibir un contraejemplo.

2. Dado un grafo dirigido $G = (V, E)$ con costos $c(u, v) \geq 0$ en su aristas y un nodo s en V (una fuente), escribir un algoritmo que use la técnica golosa para determinar las distancias mínimas desde s a todos los v en V (puede buscar entre los algoritmos conocidos). Compare con su solución del ejercicio 7 de la práctica de Programación Dinámica.

3. Dados n archivos de longitudes $l_1, ..., l_n$ se los quiere guardar en una cinta para después poder leerlos. Encontrar un algoritmo que los escriba en el orden óptimo en una cinta bajo las siguientes condiciones:

   - cada lectura comienza con la cinta completamente rebobinada,
   - la lectura de cada archivo tarda una cantidad de operaciones elementales igual a la longitud de todo lo que esté escrito antes, más la longitud del archivo leído y
   - todos los archivos van a ser leídos una vez.

   Pruebe que la solución de su algoritmo es óptima (en caso de que lo sea) y calcule la complejidad de todo el procedimiento.

4. El problema continuo de la mochila. Tenemos n sustancias, cada una de ellas con peso w_i y valor v_i por unidad. De la sustancia i tenemos un total a_i (no necesariamente entero) de unidades. A diferencia del problema discreto de la mochila, ahora podemos transportar fracciones de unidades. Encontrar un algoritmo que use la técnica golosa para maximizar el valor total de la carga sujeto a que el peso sea no mayor que un valor P dado.

5. Dado un grafo $G = (V, W)$ conexo con pesos en las aristas, un árbol generador de G es un árbol $T = (V, W')$ tal que $W' \subset W$. Su peso es la suma de los pesos de sus aristas. Un árbol generador mínimo es un árbol generador tal que su peso es mínimo entre todos los árboles generadores del grafo. Consideramos la siguiente estrategia golosa (algoritmo de Prim):

   1. Inicializar $V_{nuevos} = \{x\}$ donde x es un nodo arbitrario, $E_{nuevos} = \{\}$.
   2. Repetir hasta que $V_{nuevos} = V$.
   3.   Tomar un eje (u, v) con peso mínimo tal que u esté en V_{nuevos} y v no.
   4.   Agregar v a V_{nuevos} y (u, v) a E_{nuevos}.
   5. Devolver E_{nuevos}, serán las aristas de un árbol generador mínimo.

   - Convencerse de que este algoritmo devuelve un árbol.
   - Demuestre que el árbol devuelto es un árbol generador mínimo (es decir, la estrategia es óptima).

6. Llamamos a un grafo una foresta si es acíclico (es decir, es una unión de árboles disjuntos). No necesariamente es conexo. Encuentre un algoritmo goloso que, dado un grafo con pesos no negativos sobre sus aristas, encuentre una foresta de máximo peso.
7. En una ciudad todas las calles son de tierra y asfaltar cada cuadra en cada calle tiene un costo. Como es impracticable asfaltar todo se decide hacerlo sólo en un conjunto de costo mínimo de calles que permita llegar desde cualquier esquina a cualquier otra esquina por un camino asfaltado. Resolver con un algoritmo goloso.