

Trabajo práctico N° 6 - Greedy

Ejercicio 1

Cambio de monedas: Dado un conjunto C de N tipos de monedas con un número ilimitado de ejemplares de cada tipo, se requiere formar, si se puede, una cantidad M empleando el mínimo número de ellas.

Por ejemplo, un cajero automático dispone de billetes de distintos valores: 100\$, 25\$, 10\$, 5\$ y 1\$, si se tiene que pagar 289\$, la mejor solución consiste en dar 10 billetes: 2 de 100\$, 3 de 25\$, 1 de 10\$ y 4 de 1\$.

Ejercicio 2

Problema de la mochila: Se tienen n objetos y una mochila. Para $i = 1, 2, \dots, n$, el objeto i tiene un peso positivo p_i y un valor positivo v_i . La mochila puede llevar un peso que no sobrepase P . El objetivo es llenar la mochila de tal manera que se maximice el valor de los objetos transportados, respetando la limitación de capacidad impuesta. Los objetos pueden ser fraccionados, si una fracción x_i ($0 \leq x_i \leq 1$) del objeto i es ubicada en la mochila contribuye en $x_i p_i$ al peso total de la mochila y en $x_i v_i$ al valor de la carga.

¿Qué estrategia Greedy seguiría para resolver el problema? Esquematice la resolución mediante un pseudocódigo en Java.

Ejercicio 3

Maximizar el número de actividades compatibles. Se tienen n actividades que necesitan utilizar un recurso, tal como una sala de conferencias, en exclusión mutua. Cada actividad i tiene asociado un tiempo de comienzo c_i y un tiempo de finalización f_i de utilización del recurso, con $c_i < f_i$. Si la actividad i es seleccionada se llevará a cabo durante el intervalo $[c_i, f_i)$. Las actividades i y j son compatibles si los intervalos $[c_i, f_i)$ y $[c_j, f_j)$ no se superponen (es decir, $c_i > f_j$ o $c_j > f_i$). El problema consiste en encontrar la cantidad máxima de actividades compatibles entre sí.

Ejercicio 4

Algoritmo de Dijkstra. Dado un grafo con pesos no negativos, implemente el algoritmo de Dijkstra para determinar el array de distancias y de predecesores en el camino más corto, desde un vértice origen dado como parámetro hacia el resto de los vértices. Una vez realizado, imprima el camino más corto que se debe seguir desde el origen hacia cada vértice.

Ejercicio 5

Desde un cierto conjunto grande de ciudades del interior de una provincia, se desean transportar cereales hasta alguno de los 3 puertos pertenecientes al litoral de la provincia. Se pretende efectuar el transporte total con mínimo costo sabiendo que el flete es más caro cuanto más distancia tiene que recorrer. Dé un algoritmo que resuelva este problema, devolviendo para cada ciudad el camino que debería recorrer hacia el puerto de menor costo.

Ejercicio 6

Problema del viajante. Dado un grafo ponderado de ciudades conectadas todas con todas, implemente un algoritmo que brinde una aproximación al problema del viajante.

Ejercicio 7

Armando CDs. Dado un conjunto de archivos de canciones, donde cada uno tiene la información de nombre, género, duración del tema, y tamaño en kilobytes, se desea grabar un disco CD (que tiene una capacidad máxima de M kilobytes) de modo tal de:

- Variante A: Maximizar la capacidad ocupada del disco CD.
- Variante B: Maximizar la cantidad de canciones que se pueden grabar en el CD.

Para ambas variantes se quiere, además, que el CD no contenga más de 3 canciones de un mismo género.

Ejercicio 8

Coloreo de un grafo. Dado un grafo se desea colorear cada uno de sus vértices utilizando la menor cantidad posible de colores totales, sabiendo que dos vértices adyacentes no podrán utilizar el mismo color

Ejercicio 9

Se posee una matriz cuadrada de tamaño $N \times N$ donde en cada celda de la matriz se aloja un número entero NO negativo (es decir, ≥ 0). Dada una celda de origen y una celda de destino, se desea encontrar, de ser posible, el camino de mayor costo entre el origen y el destino. El costo del camino será medido por la suma de los valores de las celdas que conforman dicho camino. Los movimientos válidos desde una celda son arriba, abajo, derecha e izquierda, un camino no puede pasar dos veces por una misma celda, y siempre que nos movemos de una celda C1 a una celda C2, el valor de la celda C2 debe ser mayor al valor de la celda C1. Por ejemplo, si estoy en una celda con valor 10, no puedo ir a una celda de valor 8, pero si puedo ir a una de valor 12.

Se pide plantear un algoritmo mediante estrategia Greedy

- a) ¿Cuál sería la estrategia Greedy que seguiría?.
- b) Escriba un pseudo-java que lo resuelva mediante la estrategia Greedy.