CC40A - Guía 2

3 de Abril de 1997

1 Búsqueda Exhaustiva

1. Resuelva el problema del laberinto. Se tiene una matriz de celdas (i, j), con $1 \leq i, j \leq n$. Se tiene ademas información acerca de los movimientos permitidos, es decir funciones booleanas *arriba, abajo, derecha, izquierda*, por ejemplo $arriba(i, j)$ dice si es posible moverse hacia arriba desde la celda (i, j). Se parte de la celda $(1, 1)$ y la meta es la (n, n), debiéndose imprimir el camino exitoso. Tenga cuidado de no caer en ciclos. Analice su algoritmo.

2. Resuelva el problema de las reinas. Se tiene un tablero de ajedrez de $n \times n$, donde se debe ubicar el máximo número posible de reinas sin que éstas se ataquen entre sí (una reina ataca toda su fila, toda su columna, y las dos diagonales que la cruzan). Imprima las posiciones. Analice su algoritmo.

3. Diseñe un algoritmo que dado un número natural C imprima todas las formas de obtener C mediante sumar una secuencia ascendente de otros naturales mayores que 0. Por ejemplo, para 6 las soluciones son $1 + 2 + 3$, $1 + 5$, $2 + 4$, 6.

2 Heurísticas

1. Proponga heurísticas para reducir el tiempo del algoritmo de las reinas, resolviéndolo correctamente. A cuánto logra reducirlo?

2. Considere la siguiente variación del problema de la mochila. Tiene los elementos $x_1, ..., x_n$ y un número ilimitado de mochilas de tamaño K. Proponga heurísticas para utilizar el menor número posible de mochilas. Puede encontrar alguna cota a la optimalidad de su algoritmo?

3. Se tienen n elementos en un espacio desconocido, donde lo único que se conoce es la distancia $d(i, j)$ entre cada par de elementos. Cuánto cuesta encontrar el par más alejado? Proponga heurísticas más baratas, aunque no siempre hallen el óptimo.

3 Algoritmos Avaros

1. Resuelva el problema de las actividades. Hay n actividades, y cada una de ellas tiene un tiempo de comienzo s_i y un tiempo de terminación f_i. Todas compiten por un recurso, de modo que no se pueden realizar dos al mismo tiempo. Obtenga el conjunto maximal de actividades que pueden ser realizadas sin colisionar. Demuestre que su algoritmo funciona y analícelo.

2. Se tiene un conjunto de n tareas a realizar, con sus duraciones d_i. Seleccione un orden en el cual ejecutarlas de modo de reducir el tiempo total de espera de los usuarios (hay un usuario esperando por cada tarea).