

Práctica 4 - Backtracking

NOTAS PRELIMINARES

Los objetivos de esta práctica son:

- Introducir la técnica de *backtracking*.
- Identificar los pasos requeridos para alcanzar una solución por *backtracking*.
- Distinguir la diferencia entre problemas donde se busca una solución, un conjunto de soluciones o resolver un problema de optimización.

Ejercicio 1 *

A la orilla de un río se encuentra un granjero junto a un lobo, una oveja y un repollo. El granjero desea cruzar los animales a la orilla opuesta con su bote. En el bote solo puede cargarse uno de los tres y si el lobo se queda solo con la oveja la come, así como ésta lo hace con el repollo al no estar el granjero. Programar un algoritmo por *backtracking* que permita resolver el problema como una secuencia de acciones a ser realizadas por el granjero. Para eso:

- modele un estado del problema (situaciones antes o después de las acciones del granjero),
- identifique los problemas hijos (o soluciones extendidas) y las condiciones de corte (o poda),
- programe la función recursiva que realiza el *backtracking* y
- realice la primera llamada a esta función para resolver el problema.

Ejercicio 2 *

Programar una solución para el problema de Sudoku. Su entrada es una tabla con las casillas que ya están completas y su valor asociado (suponemos que la entrada es válida en el sentido de que las casillas llenas no presentan incompatibilidades).

Ejercicio 3 *

- a. Generar las permutaciones de una lista por *backtracking*.
- b. En el problema del viajante de comercio, un viajante debe visitar n ciudades y volver al origen recorriendo la menor distancia posible. Por lo tanto, la entrada del problema es el conjunto de distancias entre todo par de ciudades y la salida es un circuito que incluye a todas las ciudades. Resolver el problema del viajante de comercio usando un *backtracking*.

Ejercicio 4 *

- a. Generar el conjunto de partes de un conjunto por *backtracking*.
- b. Cómo podemos hacer para representar un subconjunto de un conjunto de N elementos utilizando un número de N bits?
Generar el conjunto de partes de un conjunto a través de los números que representan los posibles subconjuntos.
- c. Resolver el problema de la mochila usando *backtracking*.