

Práctica Backtracking

AED-Brais López Yáñez

Objetivo: implementación en C de la estrategia de vuelta atrás o *Backtracking* para resolución del problema de asignación de tareas a personas, explicando las conclusiones en cuanto a la complejidad de la tarea.

Definición: *Backtracking* o búsqueda de vuelta atrás es una técnica de programación para hacer una búsqueda exhaustiva y sistemática a través de todas las configuraciones posibles de búsqueda. Es decir, siempre encuentra la solución, ya que si no la encuentra vuelve al punto de inicio y trata de buscar el camino correcto hasta encontrarla, si no hay más posibilidades de elección la búsqueda termina. Para la implementación de *Backtracking* se usan llamadas a funciones recursivas, probando con todos los valores posibles y manteniendo aquella que ha tenido éxito en las siguientes llamadas recursivas.

El algoritmo de backtracking está muy relacionado con la búsqueda combinatoria, de estos los problemas a considerar son: problemas de decisión y problemas de optimización.

Complejidad: en el primer nivel tenemos n posibilidades, en el segundo $n-1$, y así hasta llegar al nivel n , que tendría 1 posibilidad. Por lo tanto en nuestro problema de asignación de tareas, cuando $n=3$. Tenemos 3 posibilidades para la primera asignación, 2 para la segunda asignación, y 1 para la tercera. Por lo tanto podemos afirmar y decir que la complejidad del algoritmo de *Backtracking* para asignación de personas es $n!$.

Conclusiones: el algoritmo sigue siendo muy ineficiente. Aunque garantiza la solución óptima, para mejorar el tiempo habría que aplicar una poda según el criterio de optimización.