

*Facultad de Ingeniería y Ciencias Exactas*

**Segundo Cuatrimestre 2024**

Clase 13 - Diseño y Análisis de Algoritmos

Alumna: Giorgetta Agustina

**Consignas**

Actividad 1

Copiar el código de muestra que está en el repo de la materia y observar en python tutor.

1. ¿Cómo funciona el backtracking en este problema?

En este caso nos encontramos con el problema de las N-reinas. En este buscamos colocar ‘n’ reinas en un tablero ‘n×n’ y para ello el algoritmo recorre fila por fila analizando que la posición de la reina sea segura (según los criterios esto depende de que no haya otras reinas en la misma fila, columna o diagonales).

Entonces, cuando la posición es segura se coloca una reina y se avanza recursivamente a colocar la siguiente hasta que la cantidad colocada sea igual a ‘n’ o hasta q se recorra todo y no se encuentre solución alguna. Durante el proceso de búsqueda, si en una fila no se logró colocar ninguna reina se hace el “backtrack”, retrocediendo a quitar la ultima reina puesta para recolocarla porque suponemos que no estaba en una posición optima.

1. ¿Qué pasa cuando el algoritmo encuentra una solución? ¿Qué ocurre cuando no puede colocar más reinas?

Como explicaba, el algoritmo funciona recursivamente, por lo que en cada llamada devuelve un valor: sea ‘false’ si no logró colocar más reinas y precisa continuar buscando una posición para la anterior o ‘true’ en caso de hallar la solución completa y haber colocado la ultima reina en la última fila. El valor ‘true’ podemos decir que *se propaga* en cada llamada recursiva hasta salir de la recursión. En ese punto se encontró o no una solución que se verá reflejada cuando se imprima por pantalla el tablero resultante.

1. ¿Qué sucede en el código cuando el algoritmo "retrocede"? ¿Cómo se visualiza en Python Tutor?

Cuando el algoritmo retrocede porque el análisis devolvió false, lo que hace es eliminar la reina colocada en dicha fila (volviendo el valor fila/columna = 0) y analizar la posición siguiente para moverla de lugar si es que es seguro (es decir, analiza en la misma fila la siguiente columna).

En Python Tutor podemos ver claramente como analiza paso a paso, nos muestra como se ubica en la fila 0 columna 0 y como llama al método isSafe para verificar los valores de los alrededores (diagonales y verticales), si este devuelve ‘true’ se coloca la reina y se avanza a la fila 1 columna 0 para seguir con el mismo análisis, caso contrario retrocede.

1. ¿Qué modificaciones harías para aumentar N a 8? ¿Cómo crees que cambiaría el tiempo de ejecución?

Para aumentar N a 8 solo deberíamos cambiar el 4 por el 8 en la línea de asignación: **private static final int N = 9;**

El tiempo de ejecución al crecer N aumenta exponencialmente dado que en cada iteración tenemos N filas y N columnas posibles de colocación, eso sumado a los retrocesos y reanálisis que conlleva el algoritmo).

1. ¿Por qué el método isSafe es crucial en este algoritmo?

Este método es crucial porque es el encargado de analizar que las reinas no puedan atacar o ser atacadas por otras. Además, descarta las posiciones invalidas (ósea que evita tener q hacer un retroceso por movimiento, backtrackings innecesarios) mejorando la eficiencia.