

437 The Tower of Babylon

1. ¿Su solución es BU (Bottom-Up) o TP (Top-Down)?

La solución es Top-Down. El código utiliza una función recursiva calculate_max_height que resuelve el problema buscando soluciones parciales y almacenando los resultados de subproblemas ya resueltos (memoización) para evitar cálculos redundantes

2. ¿Cual es la complejidad espacial y temporal? con una breve explicación.

Complejidad temporal:

La complejidad temporal del algoritmo es $O(n^2)$, donde n es el número total de bloques (incluyendo todas las rotaciones). Esto se debe a que, para cada bloque, en el peor de los casos, se revisan todos los otros bloques para ver si pueden ser colocados encima o debajo.

Explicación:

La generación de dependencias entre los bloques implica comparar cada bloque con los demás, lo cual toma $O(n^2)$ tiempo.

Además, la función recursiva calculate_max_height tiene un costo $O(n)$ debido a que memoiza las soluciones y no recalcula la altura para bloques ya calculados.

El bucle principal que determina la altura máxima para cada bloque también requiere $O(n)$ lo que en combinación con la búsqueda de dependencias nos lleva a $O(n^2)$

Complejidad espacial:

La complejidad espacial es $O(n)$ debido a las siguientes estructuras:

La lista dp de tamaño n almacena las soluciones parciales.

La lista dependencies, también de tamaño n , contiene las dependencias entre los bloques.

3. Explique brevemente la estrategia de la solución y como hace uso de la estrategia de Memoización.

El problema se aborda desde la perspectiva de encontrar el bloque con la mayor altura que pueda ser apilado sobre otro. Para cada bloque, se generan todas sus rotaciones posibles (ya que un bloque puede ser apilado en tres posibles orientaciones). Se construye una lista de dependencias que indica qué bloques pueden ser apilados uno sobre otro (basado en las dimensiones de sus bases). Usando recursión, la función calculate_max_height calcula la altura máxima de la torre que se puede formar colocando un bloque en la parte superior. Si la altura de un bloque ya ha sido calculada, se utiliza el valor memoizado (almacenado en la lista dp). Esto evita la recalculación de la altura para bloques que ya han sido considerados, mejorando la eficiencia de la solución.