Pontificia Universidad Católica de Chile
Departamento de Ciencia de la Computación
IIC2133 - Estructura de Datos y Algoritmos

Informe Tarea 3

21 de junio de 2024
Monserrat Díaz
21626545

Motivación

Esta tarea involucra la aplicación de diversas técnicas algorítmicas para resolver problemas complejos relacionados con grafos y optimización. La capacidad de modelar y resolver estos problemas es fundamental para desarrollar soluciones eficientes y efectivas en el ámbito de la ciencia de la computación. A continuación, se detallan las estrategias y algoritmos implementados para abordar cada una de las partes de la tarea, destacando la relevancia de estos métodos y su aplicabilidad práctica.

Informe

1. Parte 1: Cerebros Codicioso

Este problema consistía en cubrir el mayor número de planetas con el menor número de naves posibles dentro de un rango R, para lo cual se entrega una lista de posiciones de planetas y se debe determinar la cantidad mínima de naves necesarias para cubrir todos los planetas.

La estrategia utilizada es un algoritmo greedy, basado en el pseudo-código de la ayudantía 11 del curso, que ordena las posiciones de los planetas y luego coloca una nave en la posición que maximiza la cobertura de planetas no cubiertos.

La complejidad del algoritmo es $O(n \log n)$ debido a la ordenación de las posiciones de los planetas, seguida de un recorrido lineal $O(n)$. La complejidad espacial es $O(1)$, es decir, inplace.

2. Parte 2: Acumulando Basura

Esta parte de la tarea consistía en encontrar la cantidad mínima de basurales necesarios para obtener exactamente una masa dada M a partir de una lista de tipos de basurales con diferentes masas.

La estrategia utilizada es la programación dinámica, similar al problema de la mochila. Se crea un array dp donde $dp[i]$ representa la cantidad mínima de basurales necesarios para obtener una masa i. Esta estrategia es relevante porque permite considerar todas las combinaciones posibles de basurales de manera eficiente.

Respecto a la complejidad temporal del algoritmo es $O(n \times M)$, donde n es el número de tipos de basurales y M es la masa objetivo. La complejidad espacial es $O(M)$ debido al array dp.

3. Parte 3: Identificación de Regiones

Aquí se debía identificar cuántas componentes conectadas existen en un grafo no dirigido representado por una lista de aristas.

La estrategia utilizada es búsqueda en profundidad (DFS) para contar el número de componentes conectadas, que es distinto a las CFC (sino usaríamos kosaraju). Se utiliza una lista de adyacencia para representar el grafo y un array para marcar los nodos visitados. Esta técnica es relevante porque permite explorar completamente cada componente conectada de manera eficiente.

La complejidad temporal del algoritmo es $O(V + E)$, donde V es el número de nodos y E es el número de aristas. La complejidad espacial es $O(V + E)$ debido a la lista de adyacencia y el array de nodos visitados.

Conclusión

Se implementaron y analizaron diversos algoritmos para resolver los problemas propuestos en la tarea, considerando tanto la complejidad temporal como espacial según lo solicitado en el enunciado. Las estrategias adoptadas se demostraron eficientes para los tamaños de entrada esperados, proporcionando soluciones prácticas a problemas de optimización y el recorrido de grafos mediante dfs.

Referencias

- https://github.com/IIC2133-PUC/2024-1/blob/main/Clases/Secci%C3%B3n%201/
  class21%20-%20DFS%20y%20aplicaciones.pdf
- https://medium.com/@fabiola.barajas20/soluci%C3%B3n-de-un-problema-tipo-mochila-uti
  C3%B3n-din%C3%A1mica-52e0a637bff0
- https://github.com/IIC2133-PUC/2024-1/blob/main/Ayudant%C3%ADas/10%2B1-Ayudantia-Co
  20Estrategias%20Algor%C3%ADtmicas.pdf
- https://github.com/IIC2133-PUC/2024-1/blob/main/Ayudant%C3%ADas/10-Ayudantia-Greedy
  pdf
- Documentación del curso IIC2133 – Estructuras de Datos y Algoritmos, Pontificia Universidad Católica de Chile.