

Miniproyecto FADA - Universidad del Valle

David Santiago Cortés, Alejandro Orozco, Brayan Rincones

1 Soluciones Planteadas

Idea General de la solución:

Almacenar los animales y sus grandezas en una estructura de datos tipo lista, arreglo o diccionario, se ordena esta estructura de acuerdo a los valores de las grandezas y se guarda utiliza para armar las escenas de todas las partes del evento.

1.1 $O(n^2)$

Idea de la solución:

Estructuras de datos utilizadas:

Algoritmo de ordenamiento:

Lenguaje en el que se implementó:

1.2 $O(n * \log(n))$

Idea de la solución:

Estructuras de datos utilizadas:

Algoritmo de ordenamiento:

Lenguaje en el que se implementó:

1.3 $O(n)$

Idea de la solución: Después de recolectar los datos de entrada se construye un diccionario que mapea animales a sus respectivas grandezas a partir de la lista de animales y grandezas que se pasaron como parametro de entrada. Este diccionario se usará más adelante para apoyar distintas operaciones.

Para atacar el problema se utiliza una estrategia *bottom-up*

1. Ordenar las escenas localmente: para que todas las escenas del evento queden ordenadas localmente solo basta con ordenar las escenas de la apertura. El ordenamiento se hace con una implementación de Counting Sort que ordena una lista de tuplas (animal,grandezas) en orden ascendente de acuerdo a las grandezas.
2. Ordenar las escenas de las demás partes (localmente): con el resultado del paso anterior se construye un diccionario/tabla hash que relaciona escenas desordenadas con escenas ordenadas, se usa un ciclo que itera sobre cada escena de las $m-1$ partes y se reemplaza cada escena en cada parte haciendo la consulta en la tabla hash.
3. Ordenar todas las escenas: teniendo todas las escenas ordenadas localmente en cada parte, lo que resta es ordenar las escenas dentro de sus partes de acuerdo a sus grandezas. El proceso se divide en dos, la primera parte se encarga de constuir un diccionario/tabla hash que relacione escenas con su grandezza, para ello se itera sobre apertura y se calcula la grandezza de cada escena, con esta información ordenamos apertura utilizando Counting Sort.
En la segunda parte se itera sobre las siguientes $m - 1$ partes y sobre las k escenas de tales partes, se consulta cada escena en el diccionario para hallar su grandezza y antes de cambiar a la siguiente parte,

se ordenan las escenas de la parte actual con Counting Sort. Al finalizar el ciclo externo las escenas dentro de cada parte están ordenadas ascendentemente.

(a) Empates: se buscan y se resuelven cada que se termina de ordenar las escenas de una parte.

4. Ordenar las partes:

Estructuras de datos utilizadas: Listas, tablas hash, tuplas

Algoritmo de ordenamiento: Counting Sort para listas de tuplas (animal, grandeza)

Lenguaje en el que se implementó: Python

2 Análisis de Resultados

2.1 $O(n^2)$

2.2 $O(n * \log(n))$

2.3 $O(n)$

3 Instrucciones para la ejecución

4 Sets de prueba

5 Conclusiones del proyecto