



TOLENTINO ROMERO GERARDO



Analisis y Diseño de Algoritmos

Andres Garcia Floriano

Practica 5

Instrucciones

Recibes un arreglo de k listas vinculadas, cada una está ordenada de forma ascendente

Fusiona todas las listas vinculadas en una sola lista y retórnala.

Nota: puedes representar las listas como arreglos.

Solución

```
def mergeKLists(lists):  
    heap = []  
  
    for linked_list in lists:  
        for val in linked_list:  
            heapq.heappush(heap, val)  
  
    merged_list = []  
  
    while heap:  
        merged_list.append(heapq.heappop(heap))  
  
    return merged_list  
  
lists = [[1, 4, 5], [1, 3, 4], [2, 6]]  
print(mergeKLists(lists))
```

Introducción

En esta práctica, el objetivo fue fusionar varias listas ordenadas en una sola lista usando Python. Se nos permitió representar las listas como arreglos en vez de listas vinculadas.

Usamos un **min-heap** para juntar los elementos de todas las listas y luego extraerlos en orden. Con la librería `heapq` de Python, logramos manejar los datos de manera eficiente y generar la lista fusionada de forma ordenada.

El código funcionó correctamente. Por ejemplo, al fusionar las listas `[1, 4, 5]`, `[1, 3, 4]`, y `[2, 6]`, obtuvimos `[1, 1, 2, 3, 4, 4, 5, 6]`. También, en el caso de listas vacías, el resultado fue una lista vacía.

Conclusión

La técnica del min-heap fue útil y eficiente para este problema. Sin embargo, para conjuntos pequeños, una solución más sencilla podría ser igual de válida. Esta práctica nos permitió entender mejor el uso de estructuras de datos en problemas comunes.