

2023-05-13 - Handout – Priority Queues / Heaps

Q1. Merge k Sorted Lists

You are given an array of k linked-lists `lists`, each linked-list is sorted in ascending order.

Merge all the linked-lists into one sorted linked-list and return it.

Example 1:

Input: `lists = [[1,4,5],[1,3,4],[2,6]]`

Output: `[1,1,2,3,4,4,5,6]`

Constraints:

- $k == \text{lists.length}$
- $0 \leq k \leq 10^4$
- $0 \leq \text{lists}[i].\text{length} \leq 500$
- $-10^4 \leq \text{lists}[i][j] \leq 10^4$
- `lists[i]` is sorted in **ascending order**.
- The sum of `lists[i].length` will not exceed 10^4 .

Q2. Meeting Rooms II

Given an array of meeting time intervals `intervals` where `intervals[i] = [starti, endi]`, return *the minimum number of conference rooms required*.

Example 1:

Input: `intervals = [[0,30],[5,10],[15,20]]`

Output: 2

Constraints:

- $1 \leq \text{intervals.length} \leq 10^4$
- $0 \leq \text{start}_i < \text{end}_i \leq 10^6$

Q3. Reorganize String

Given a string `s`, rearrange the characters of `s` so that any two adjacent characters are not the same.

Return any possible rearrangement of `s` or return "" if not possible.

1) Input: `s = "aab"`, Output: `"aba"`

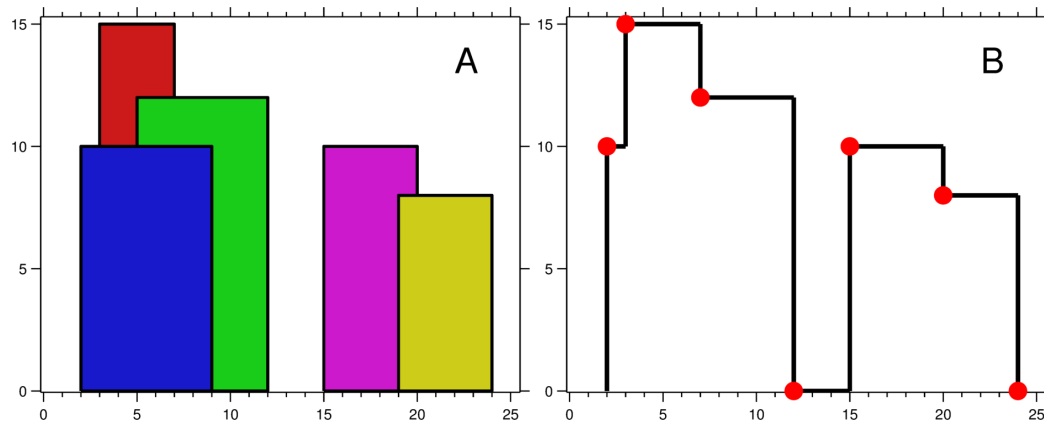
2) Input: `s = "aaab"`, Output: `""`

Constraints:

- $1 \leq s.\text{length} \leq 500$ (`s` consists of lowercase English letters.)

Q4. The Skyline Problem

A city's skyline is the outer contour of the silhouette formed by all the buildings in that city when viewed from a distance. Now suppose you are given the locations and height of all the buildings as shown on a cityscape photo (Figure A), write a program to output the skyline formed by these buildings collectively (Figure B).



The geometric information of each building is represented by a triplet of integers $[Li, Ri, Hi]$, where Li and Ri are the x coordinates of the left and right edge of the i th building, respectively, and Hi is its height. It is guaranteed that $0 \leq Li, Ri \leq \text{INT_MAX}$, $0 < Hi \leq \text{INT_MAX}$, and $Ri - Li > 0$. You may assume all buildings are perfect rectangles grounded on an absolutely flat surface at height 0.

For instance, the dimensions of all buildings in Figure A are recorded as: $[[2, 9, 10], [3, 7, 15], [5, 12, 12], [15, 20, 10], [19, 24, 8]]$.

The output is a list of "key points" (red dots in Figure B) in the format of $[[x1, y1], [x2, y2], [x3, y3], \dots]$ that uniquely defines a skyline. A key point is the left endpoint of a horizontal line segment. Note that the last key point, where the rightmost building ends, is merely used to mark the termination of the skyline, and always has zero height. Also, the ground in between any two adjacent buildings should be considered part of the skyline contour.

For instance, the skyline in Figure B should be represented as: $[[2, 0], [3, 15], [7, 12], [12, 0], [15, 10], [20, 8], [24, 0]]$.