

Problem 1: Maximum Number of Non-Overlapping Tasks

Overview: Given N tasks, each with a start and end time, determine the maximum number of tasks a person can complete without any time overlap. This problem is a classic example of the activity selection problem, where the goal is to select the maximum number of activities that don't overlap in time.

Algorithm Steps:

1. Sort Tasks:

- First, sort the tasks based on their end times. This sorting is crucial as it allows selecting the maximum number of non-overlapping tasks.

2. Select Tasks:

- Start with the first task in the sorted list as it finishes the earliest.
- For each subsequent task, if its start time is greater than or equal to the end time of the previously selected task, select it.

3. Implementation Details:

- Define a class Task with start and end as attributes.
- Implement a comparator to sort the tasks based on end times.

4. Code

```
import java.util.Arrays;

public class TaskScheduler {
    static class Task {
        int start;
        int end;

        Task(int start, int end) {
            this.start = start;
            this.end = end;
        }
    }

    // Function to find the maximum number of non-overlapping tasks
    public static int maxTasks(Task[] tasks) {
        Arrays.sort(tasks, (a, b) -> a.end - b.end);

        int count = 0;
        int lastEndTime = Integer.MIN_VALUE;
        for (Task task : tasks) {
            if (task.start >= lastEndTime) {
                count++;
                lastEndTime = task.end;
            }
        }
        return count;
    }

    // Main method to test the maxTasks function
    public static void main(String[] args) {
        Task[] tasks = {
            new Task(1, 3),
            new Task(2, 5),
            new Task(4, 6),
            new Task(6, 7),
            new Task(5, 8),
            new Task(8, 9)
        };
    }
}
```

```
};

    System.out.println("Maximum number of tasks that can be completed: " + maxTasks(tasks));
}
}
```

➤ **Problem 2, 3**

↳ 2 cells hidden