Lab 5

CSC 225 Summer 2023

May 30, 2023

Problem 1: Merge overlapping intervals. Given a set of integer intervals, merge overlapping ones and print a final set of non-overlapping intervals. Use a stack.

More specifically, write `Algorithm MergeOverlappingIntevals(intervals, N)` that takes as input

- `intervals`: an array of intervals
- `N`: the number of intervals in the `intervals` array

and returns

- `interval list`: a list of the merged intervals

Each interval `i` has two attributes:

- `i.start`: the lower bound of interval `i`
- `i.end`: the upper bound of interval `i`

Example: given `intervals` $\leftarrow$ `[(1,3)(2,6),(2,4),(7,10)]`, you should return `[(1,6), (7,10)]`. Intervals `(1,3)`, `(2,6)`, and `(2,4)` are merged as they contain at least one integer in common.

Hint: Sort the intervals.