

CS311 Homework 3

200 points

Problem 1

For this assignment you will implement three solutions to the Interval Scheduling problem, :

Problem: Interval Scheduling

Input: A set of jobs J , each described by an integer start and end time.

Output: The largest possible set $S \subset J$ of jobs such that $\forall x, y \in S, x \neq y \rightarrow x$ and y do not conflict. Two jobs A and B conflict iff A starts strictly before B ends and B starts strictly before A ends, that is, if A 's start time is smaller than B 's end time and B 's start time is smaller than A 's end time.

1. Write a class called "BruteForceScheduler" that implements the interface IScheduler defined below. BruteForceScheduler shall solve the Interval Scheduling problem by iterating over every subset of the input set of jobs (which is, of course, precisely the set of possible solutions to the input), keeping the largest set with no conflicts.
2. Write a class called "SmartBruteForceScheduler" that implements the interface IScheduler defined below. SmartBruteForceScheduler shall solve the Interval Scheduling problem by iterating over the possible solutions to its input from largest to smallest, in such a fashion that the first solution discovered is guaranteed to be a largest solution. The algorithm shall then immediately halt and return the discovered solution.
3. Write a class called "EarliestDeadlineScheduler" that implements the interface IScheduler defined below. EarliestDeadlineScheduler shall contain an implementation of the linear-time Earliest Deadline algorithm discussed in lecture.

Run BruteForceScheduler, SmartBruteForceScheduler, and EarliestDeadlineScheduler on some example inputs of varying sizes. Measure the execution time required for each algorithm to solve each example according to a methodology of your choice. In a .txt file titled results.txt, describe your methodology, your results, and compare the results to your expectations based on a Big-Oh running time analysis of your implementations.

Submit BruteForceScheduler.java, SmartBruteForceScheduler.java, and EarliestDeadlineScheduler.java, and results.txt on Blackboard by the posted due date. You will be graded on the correctness of your implementations, the accuracy of your experimental methodology, and the correctness of your Big-Oh running time analyses.

```
import java.util.Set;

public interface IScheduler
{
    public interface IInterval
    {
        public int getStartTime();
        public int getEndTime();
    }
    public Set<IInterval> optimalSchedule(Set<IInterval> s);
}
```