# CSCI3160: Regular Exercise Set 3

Prepared by Yufei Tao

**Problem 1.** Let $S$ be a set of $n$ intervals $\{[s_i, f_i] \mid 1 \le i \le n\}$, satisfying $f_1 \le f_2 \le ... \le f_n$. Denote by $S'$ the set of intervals in $S$ that are disjoint with $[s_1, f_1]$. Prove: if $T' \subseteq S'$ is an optimal solution to the activity selection problem on $S'$, then $T' \cup \{[s_1, f_1]\}$ is an optimal solution to the activity selection problem on $S$.
(Note: This completes the induction step of the correctness proof discussed in the class.)

**Solution.** We will prove the claim by contradiction. Suppose that $T' \cup \{[s_1, f_1]\}$ is not an optimal solution to the activity selection problem on $S$. As proved in the class, there exists an optimal solution $T$ (to the activity selection problem on $S$) which includes $[s_1, f_1]$. Because all the intervals in $T' \cup \{[s_1, f_1]\}$ are disjoint, we know $|T' \cup \{[s_1, f_1]\}| < |T|$ (otherwise, $T' \cup \{[s_1, f_1]\}$ would be an optimal solution to the activity selection problem on $S$).

Since every interval in $T \setminus \{[s_1, f_1]\}$ is disjoint with $[s_1, f_1]$, we know that all the intervals in $T \setminus \{[s_1, f_1]\}$ must come from $S'$. As $T'$ is an *optimal* solution the activity selection problem on $S'$, we know:

$$\begin{aligned} |T'| &\ge |T \setminus \{[s_1, f_1]\}| \\ \Rightarrow |T' \cup \{[s_1, f_1]\}| &\ge |T| \end{aligned}$$

thus causing a contradiction.

**Problem 2.** Describe how to implement the activity selection algorithm discussed in the lecture in $O(n \log n)$ time, where $n$ is the number of input intervals.

**Solution.** Let $S$ be the set of $n$ intervals given, where each interval has the form $[s, f]$. Sort the intervals in ascending order the $f$-value. Denote the sorted order as $[s_1, f_1], [s_2, f_2], ..., [s_n, f_n]$ where $f_1 \le f_2 \le ... \le f_n$. Proceed as follows:

1. $T = \{[s_1, f_1]\}$; $last = 1$
2. **for** $i = 2$ to $n$
3.     **if** $s_i > f_{last}$ **then**
4.         add $[s_i, f_i]$ into $T$; $last = i$

After sorting, the above algorithm runs in $O(n)$ time.

**Problem 3.** Prof. Goofy proposes the following greedy algorithm to "solve" the activity selection problem. Let $S$ be the input set of intervals. Initialize an empty $T$, and then repeat the following steps until $S$ is empty:

- (Step 1) Add to $T$ the interval $I = [s, f]$ in $S$ that has the smallest $s$-value.
- (Step 2) Remove from $S$ (i) the interval $I$, and (ii) all the intervals that overlap with $I$.

Finally, return $T$ as the answer.

Prove: the above algorithm does not guarantee an optimal solution.

**Solution.** Here is a counterexample: $S = \{[1, 10], [2, 3], [4, 5]\}$. Prof. Goofy's algorithm returns $\{[1, 10]\}$, while the optimal solution is $S = \{[2, 3], [4, 5]\}$.

**Problem 4**.** Prof. Goofy is giving another try! This time he proposes a more sophisticated greedy algorithm. Again, let $S$ be the input set of intervals. Initialize an empty $T$, and then repeat the following steps until $S$ is empty:

- (Step 1) Add to $T$ the interval $I \in S$ that overlaps with the *fewest* other intervals in $S$.
- (Step 2) Remove from $S$ the interval $I$ as well as all the intervals that overlap with $I$.

Finally, return $T$ as the answer.

Prove: the above algorithm does not guarantee an optimal solution.

**Solution.** The following nice counterexample is by courtesy of the site *http://mypathtothe4.blogspot.com/2013/03/greedy-algorithms-activity-selection.html.*

$$S = \{[1, 10], [2, 22], [3, 23], [20, 30], [25, 45], [40, 50], [47, 62], [48, 63], [60, 70]\}$$

Prof. Goofy's algorithm returns 3 intervals (one of them must be $[25, 45]$), while the optimal solution consists of 4 intervals.