CS340 Analysis of Algorithms

Handout:	**1**	**Professor:**	**Dianna Xu**
Title:	**Greedy Interval Scheduling**	**E-mail:**	dxu@brynmawr.edu
Date:		**URL:**	**http://cs.brynmawr.edu/cs340**

Sample description, pseudo code and proof of correctness for the greedy interval scheduling algorithm. Please refer to lecture notes for details of the algorithm design and time analysis. Recall that given a set $R = \{r_1, r_2, ..., r_n\}$ of n activities, with associated start and finish times $[s_i, f_i]$, $\forall 1 \le i \le n$, we want to determine the largest subset of R that are non-conflicting.

1 Description

We begin by sorting the activities by earliest finish times. We schedule the activity r with the earliest finish time, skip over all activities that conflict with r from our sorted list, and repeat.

2 Pseudocode

```
presort R by finish times and store in sorted array
Function greedyInterval(R)
    A = ∅
    pi = -1 // index of last picked activity
    for each r_i ∈ R do
        //if no last picked or current activity has no conflict with last picked
        if (pi == -1 || s_i > f_pi) then
            append r_i to A
            pi = i
        end
    end
    return A
```

3 Correctness Proof

Termination is easily argued because the set R is finite and we delete at least one activity in each iteration of the `while` loop.
We prove the optimality of `greedyInterval` by showing that the greedy criteria of earlist finish first (EFF) is optimal.
Let $O = \{[o_{s_1}, o_{f_1}], ..., [o_{s_i}, o_{f_i}], ...\}$ be the set of activities in some optimal schedule and $G = \{[g_{s_1}, g_{f_1}], ..., [g_{s_i}, g_{f_i}], ...\}$ be the set of activities picked by the EFF schedule. Both sets are sorted by earliest finish times.
Lemma: Greedy finishes first. Specifically, any greedy activity has earlier or the same finish time as its counter part in any optimal schedule, or $g_{f_i} \le o_{f_i}, \forall i$
Proof: by induction.

- Base case: $i = 1$, by greedy construction

- IH: assume that `EFF` works up till $n-1$, i.e. $g_{f_i} \leq o_{f_i}, \forall 1 \leq i \leq n-1$.
- Want to show: $g_{f_n} \leq o_{f_n}$.
  By IH, g_{n-1} finishes earlier than o_{n-1}. o_n must start after o_{n-1} (O is optimal and activities in O can not overlap themselves), therefore also after g_{n-1}. Thus $o_{s_n} > g_{f_{n-1}}$. Notice that o_n has no conflict with g_{n-1} and therefore must be in the pool of candidate activities when EFF was picking g_n, along with g_n. If o_n and g_n differ, then by EFF construction, g_n must have earlier finish times than o_n, and thus $g_{f_n} \leq o_{f_n}$.

Finally, we argue that $|G| = |O|$. We just proved that EFF activities always finish earlier or the same as any optimal schedule picks. Recall that the goal of the original problem was to find the largest subset, and therefore we must show that it is impossible for O to pick activities with later finish times yet somehow end up with a larger set. Suppose that to the contrary, it was so and $|O| > |G|$. We consider the first activity in O that starts after EFF activities ended, i.e. some o_{n+1}, where $|G| = n$. Note that o_{n+1} has no conflict with g_n and therefore would not be deleted from the list of activities when g_n was scheduled. Thus EFF would not have terminated and would have scheduled o_{n+1} into G next. Contradiction. ■