CS3000: Algorithms & Data — Summer '25 — Laney Strange

Scheduling - Greedy

This is an optimization problem. We have a list of activities denoted by their start times s and finish times f. Two activites a_i, a_j are *compatible* if $f[i] \le s[j]$ or $f[j] \le s[i]$. We want to create a set S of compatible activities, and we want that set S to be optimal. In this case, we are optimizing for number of activities – we want $|S|$ to be as large as possible.

The algorithm below takes in two arrays s and f – the start and finish times of our activities. The arrays are the same length, and correspond with each other: Activity a_i has start time $s[i]$ and finish time $f[i]$. It returns a set of indices.

```
SCHEDULING($s, f, n$)
sort $s$ and $f$ by finish time
$S = \{1\}$
$k = 1$
for $m = 2$ to $n$
    if $s[m] \ge f[k]$
        $S = S \cup \{m\}$
        $k = m$
return $S$
```

We typeset the pseudocode above with the following LaTeX:

```
    \begin{codebox}
\Procname{$\proc{Scheduling}(s, f, n)$}
\li sort $s$ and $f$ by finish time
\li $S \gets \{1\}$
\li $k \gets 1$
\li \For $m \gets 2 \To n$
\Do
\li \If $s[m] \ge f[k]$
\Then
\li $S \gets S \cup \{m\}$
\li $k \gets m$
\End
\End
\li \Return $S$
\end{codebox}
```