19.7
Greedy algorithms – an epilogue

Greedy proof techniques: Overview

1. **Greedy's first step leads to an optimum solution.** Show that optimal solution can be modified to agree with greedy after first step. Then use induction. Example, Interval Scheduling.

2. **Greedy algorithm stays ahead.** Show that after each step the solution of the greedy algorithm is at least as good as the solution of any other algorithm. Example, Interval scheduling.

3. **Structural property of solution.** Observe some structural bound of every solution to the problem, and show that greedy algorithm achieves this bound. Example, Interval Partitioning (see Kleinberg-Tardos book).

4. **Exchange argument.** Gradually transform any optimal solution to the one produced by the greedy algorithm, without hurting its optimality.
   Example: Minimizing lateness, and Interval scheduling

Takeaway Points

1. Greedy algorithms come naturally but often are incorrect.
   A proof of correctness is an absolute necessity.

2. Exchange arguments are often the key proof ingredient. Focus on why the first step of the algorithm is correct: need to show that there is an optimum/correct solution with the first step of the algorithm.

3. Thinking about correctness is also a good way to figure out which of the many greedy strategies is likely to work.