CS314: Greedy Algorithms

Note Title 2/8/2010

Announcements

- HW 3 out, due on Friday (written)
- No office hours Thursday 1-2; instead, will be available 9-10 am on Thursday

(Also will be in Wed. morning.)

Greedy Algorithms

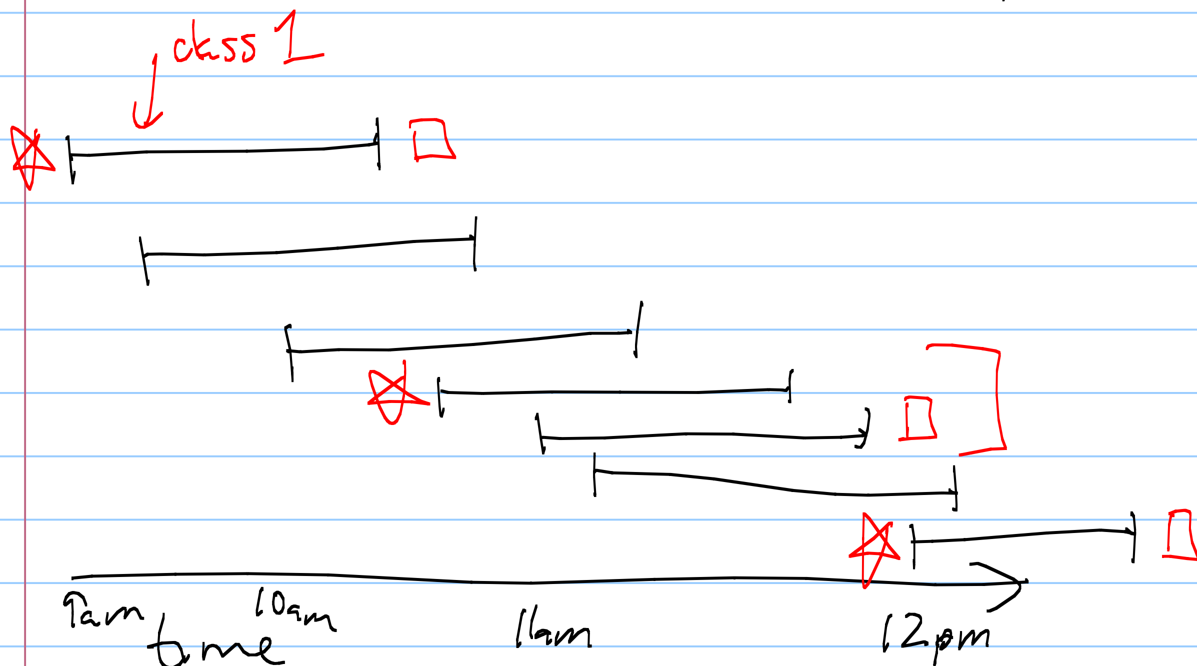
Main idea: Make a choice that is as good as possible *in the short term*.

Problem: Often doesn't work!

Proofs of correctness are very important here!

Ex: Set of requests for a classroom.

Goal: Schedule as many classes as possible.



How many?

3

Interval Scheduling

Input: Set of requests $\{1, .., n\}$
i^{th} request starts at time $s(i)$ and finishes at time $f(i)$

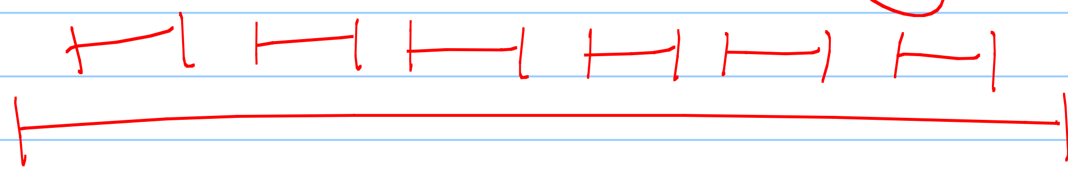
A subset of requests is <u>compatable</u> if no 2 overlap.

Goal: Find compatable set of maximum size.

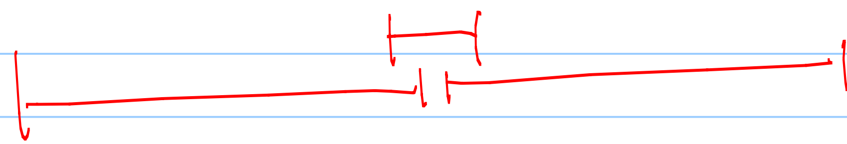
What are some "greedy" strategies?

← start earliest

① Take first job + keep going



② Take shortest job

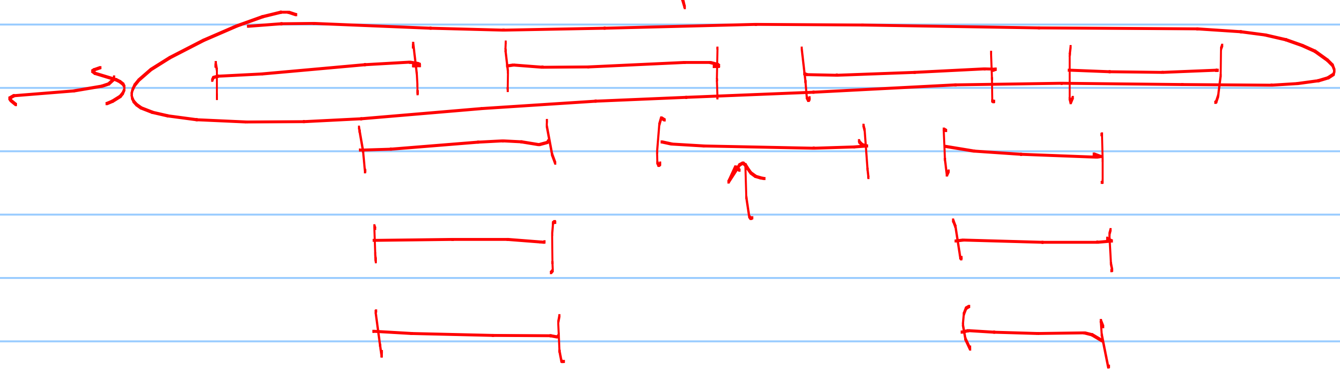


③ Earliest ending time ←

Other ideas:

(4) Take interval with fewest overlaps

counterexample



Idea: Take interval that finishes first.

```
GreedySchedule(S[1..n], F[1..n]):
    sort F and permute S to match
    count ← 1
    X[count] ← 1
    for i ← 2 to n
        if S[i] > F[X[count]]
            count ← count + 1
            X[count] ← i
    return X[1..count]
```

Runtime:

$O(n \log n)$

$O(n)$

Correctness:

Why should this work?

(Need to prove it is competable & as large as possible.)

Compatable: X is valid since no interval is accepted if it starts before previous interval ends.

(nothing overlaps)

Notation: Let O be optimal solution, $\{j_1, \ldots, j_m\}$

Let our solution be $\{i_1, \ldots, i_k\}$

$\rightarrow$ Goal: $m = k$ $\leftarrow$

Our Alg: $\{i_1, \dots, i_k\}$

Θ: $\{j_1, \dots, j_m\}$

Proof of correctness:

Lemma 1: For all indices $r \le k$, $f(i_r) \le f(j_r)$.

Induction on r:

Base Case: $r = 1$. Why is $f(i_1) \le f(j_1)$?

Our alg chose job that ended earliest.

IH: Assume $f(i_{r-1}) \le f(j_{r-1})$.

IS: Consider $f(i_r)$ & $f(j_r)$.

Know $f(j_{r-1}) \le s(j_r)$ since Θ is compatable.

$\Rightarrow f(i_{r-1}) \le s(j_r)$ by our IH.

So our alg could have chosen j_r.

Since our alg selected earliest available finish, $f(i_r) \le f(j_r)$ ∎

Claim: The greedy algorithm is optimal.

Proof: If it isn't then $m > k$.

Using lemma, know $f(i_k) \leq f(j_k)$

Since $m > k$, there is a j_{k+1}.

$s(j_{k+1}) \geq f(j_k) \geq f(i_k)$

We could take j_{k+1} also!

So we have $> k$ elements