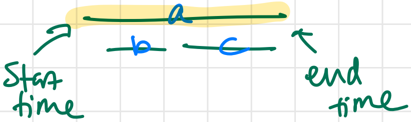
Interval Scheduling: given n jobs with start and finishing times, choose the largest compatible set of jobs.

```
Greedy-Schedule(n, array S of n start times,
                array f of n finishing times):
  sort jobs by ______
    and reorder f, S accordingly
  S = ∅
  for i in 1 to n:
      if job i is compatible w/ S:
          add i to S
```

best # compatible jobs
↓

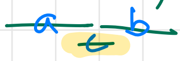
Earliest start time



OPT = 2 (choose jobs b, c)

EST chooses 1 (chooses job a)

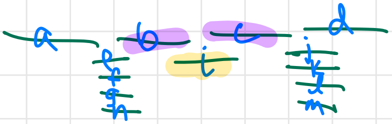
Shortest job length



OPT = 2
a, b

SJL chooses 1
c

Smallest # conflicts



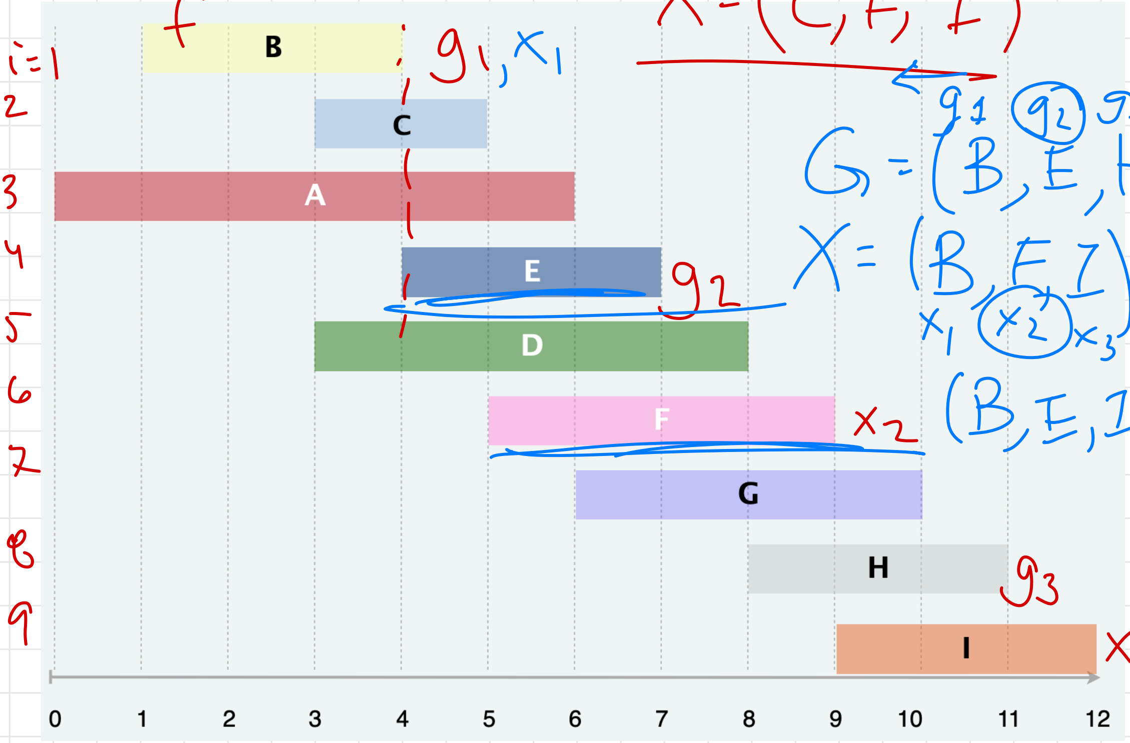
OPT = 4
a, b, c, d

SNC chooses 3
i,

Earliest Finish time

$S = (B, E, H)$

$X = (C, F, I)$



$g_1 \quad g_2 \quad g_3$

$G = (B, E, H)$

$X = (B, F, I)$

$x_1 \quad x_2 \quad x_3$

(B, E, I)

label jobs in order that EFT greedy schedule alg chooses them

$g_1, g_2, \dots, g_k$

Claim 1 For any instance of the interval scheduling problem, there is an optimal solution that uses the earliest finishing job.

in order of start time

Proof Let $X = (x_1, x_2, \dots, x_m)$ be an optimal solution to the ISP and let f be the earliest finishing job. If $f \neq x_1$

Can you swap f for x_1? why/why not?

Notice that f finishes before (or at the same time as) x_1 and x_1 has no conflicts with the rest of the jobs in X, so f also has no conflicts with the rest of the jobs in X.

So swapping f for x_1 yields a compatible schedule of the same size.

Claim 2 Every job in a greedy earliest finish time output can be swapped into an optimal solution maintaining compatibility.

Proof let $G = (g_1, g_2, \ldots, g_k)$ be the jobs chosen by greedy and let

$$X = (g_1, g_2, \ldots, g_{j-1}, \overset{g_j}{x_j}, x_{j+1}, \ldots, x_m)$$

be an optimal solution where x_j is the first difference from G.

IH: Assume for jobs before x_j, they could have been swapped into X maintaining compatibility.

Base case: claim 1. (suppose $j=1$ ...)

Inductive case:

By IH, $g_1, g_2, \ldots, g_{j-1}$ were swapped into X without creating conflicts. g_j does not conflict with these jobs. But what about $x_{j+1}, \ldots, x_m$? Since g_j has the earliest finish time amongst jobs that don't conflict with $g_1, g_2, \ldots, g_{j-1}$, g_j must finish earlier than x_j.

So swapping g_j for x_j maintains compatibility.