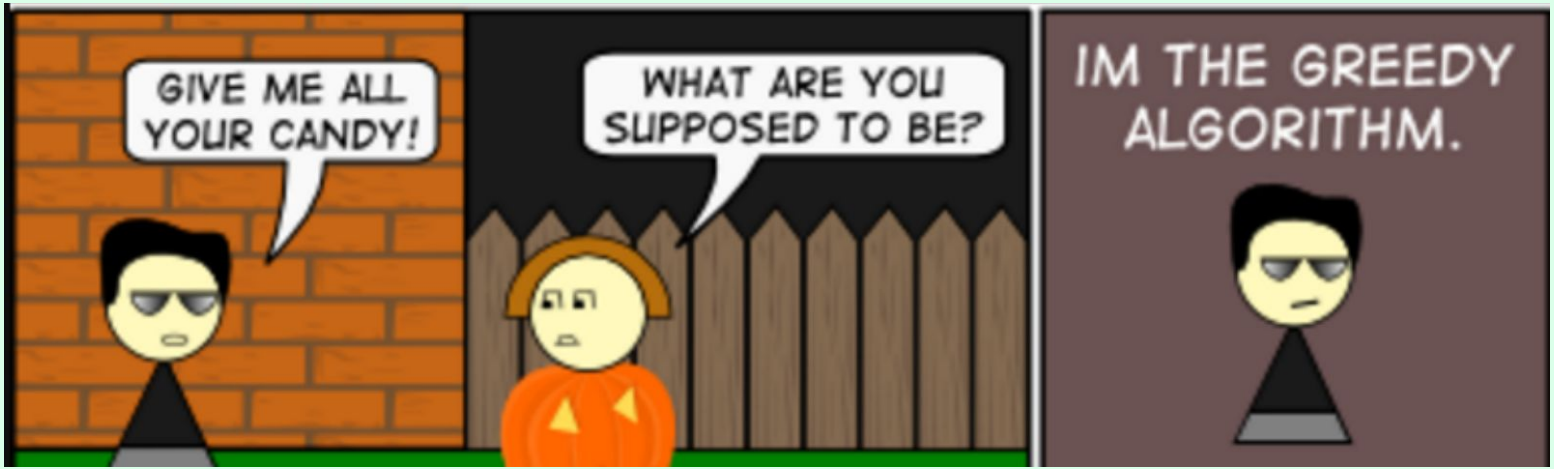
CS 310

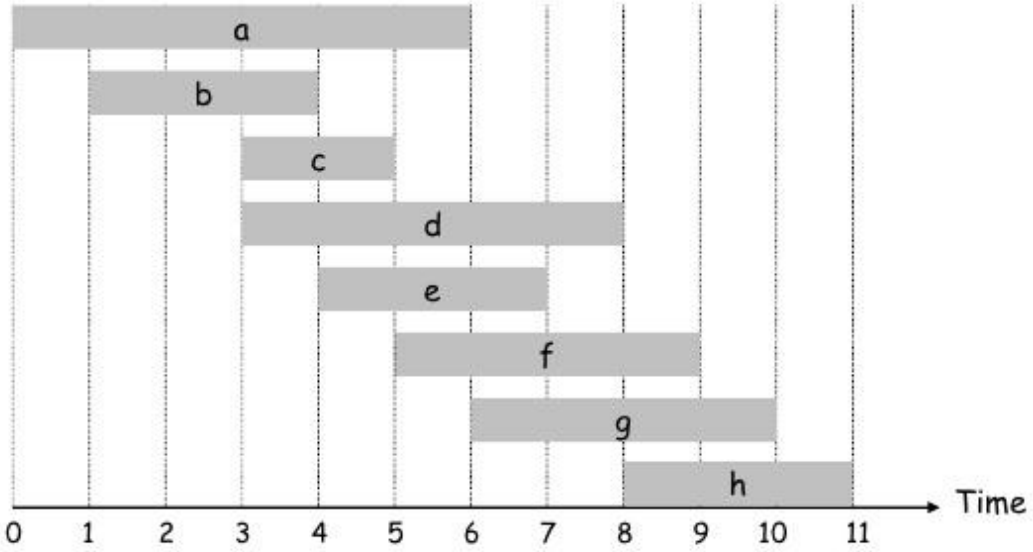

GIVE ME ALL YOUR CANDY!
WHAT ARE YOU SUPPOSED TO BE?
IM THE GREEDY ALGORITHM.

Interval Scheduling Problem

Interval Scheduling

Interval scheduling.

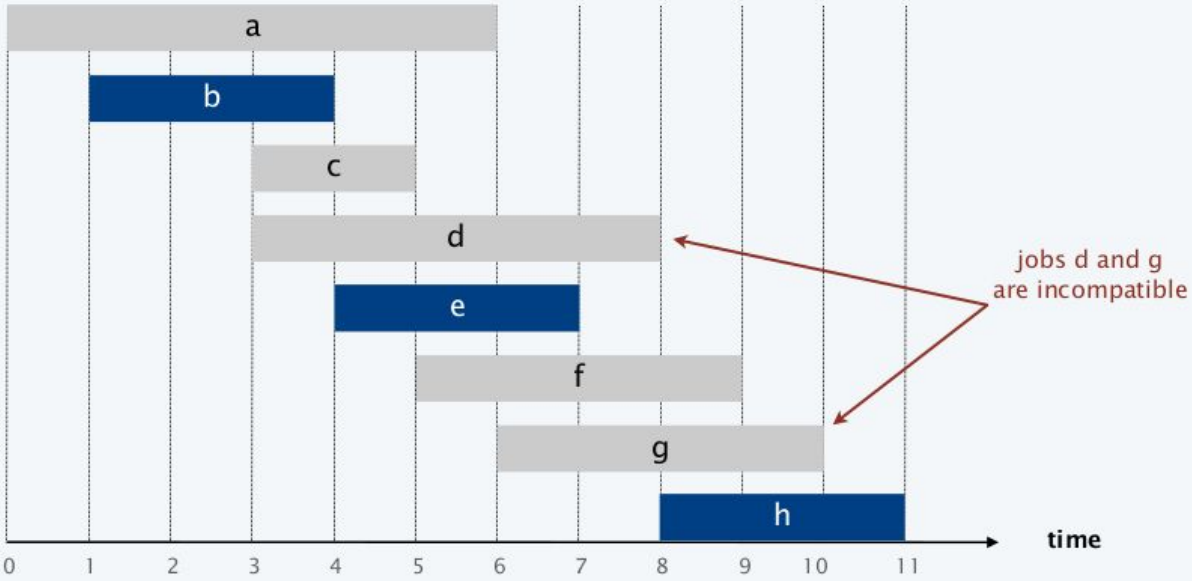
- Job j starts at s_j and finishes at f_j.
- Two jobs compatible if they don't overlap.
- Goal: find maximum subset of mutually compatible jobs.



One classroom analogy

Interval scheduling

- Job j starts at s_j and finishes at f_j.
- Two jobs compatible if they don't overlap.
- Goal: find maximum subset of mutually compatible jobs.



Greedy algorithms

Short-sighted greed in the design of algorithms

Greedy algorithms

Short-sighted greed in the design of algorithms

An algorithm is greedy if it builds up a solution in small steps, shortsightedly choosing a decision at each step to **optimize some underlying criterion**.

Greedy algorithms

Short-sighted greed in the design of algorithms

An algorithm is greedy if it builds up a solution in small steps, shortsightedly choosing a decision at each step to **optimize some underlying criterion**.

A greedy algorithm always **makes the choice that looks best at the moment**. That is, it **makes a locally optimal choice in the hope that this choice will lead to a globally optimal solution**.

Steps in a greedy algorithm for interval scheduling

- Use a simple rule to select a first request.
  - Once a request is accepted, we reject all requests that are not compatible with it.
- Select the next request to be accepted.
  - Again reject all requests that are not compatible with it.
- Continue until we run out of requests.

The challenge is in deciding **which simple rule to use** for the selection

counterexample for earliest start time

counterexample for shortest interval

counterexample for fewest conflicts

Interval scheduling: earliest-finish-time-first algorithm

EARLIEST-FINISH-TIME-FIRST $(n, s_1, s_2, \ldots, s_n, f_1, f_2, \ldots, f_n)$

SORT jobs by finish time so that $f_1 \leq f_2 \leq \ldots \leq f_n$

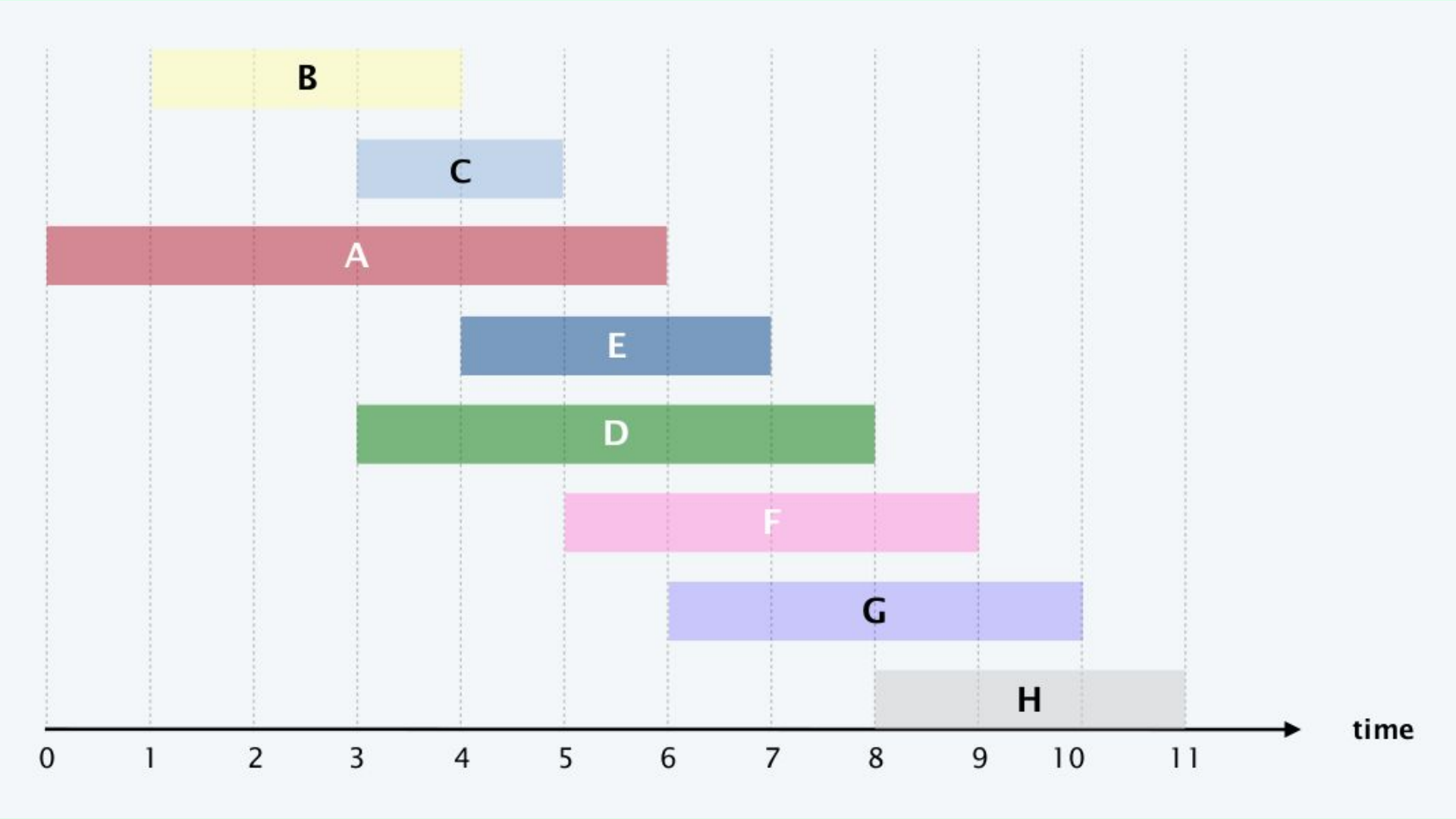
$A \leftarrow \phi$ ⟵ set of jobs selected

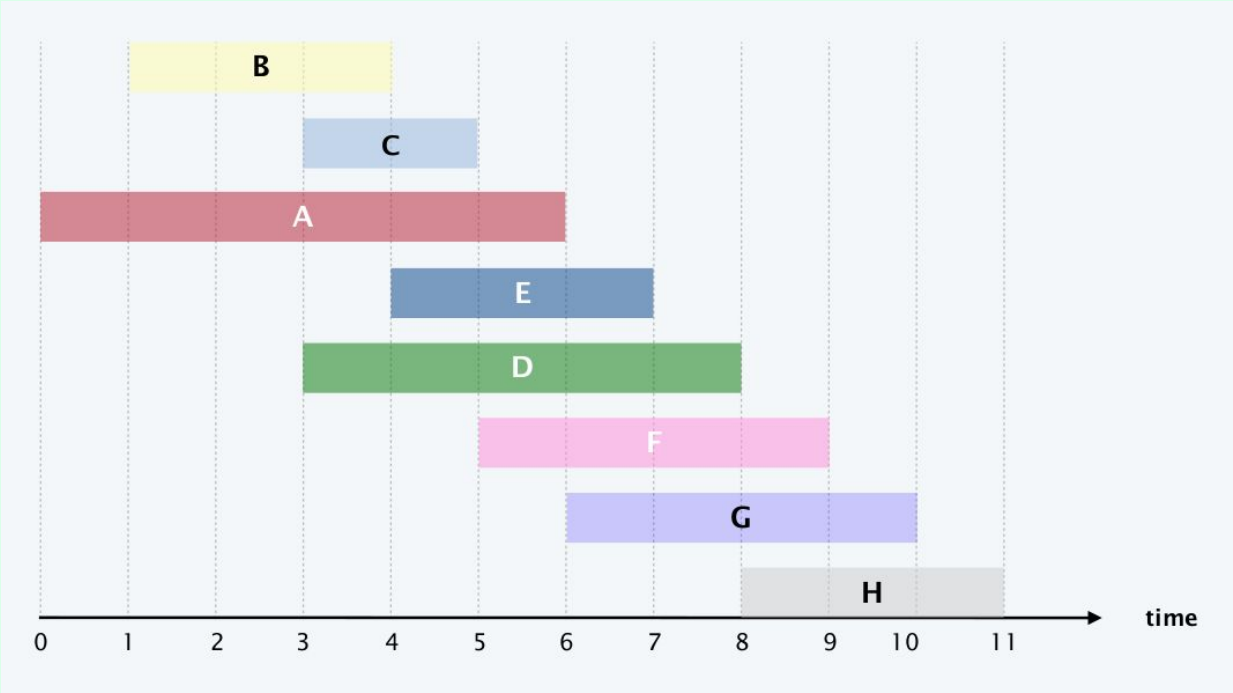
FOR $j = 1$ TO n

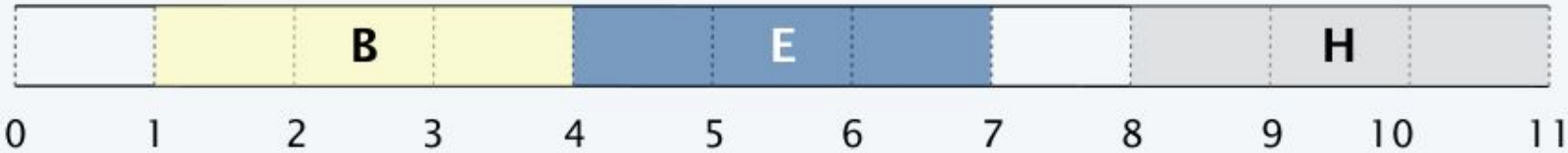
  IF job j is compatible with A

    $A \leftarrow A \cup \{ j \}$

RETURN A


B
C
A
E
D
F
G
H
time
0
1
2
3
4
5
6
7
8
9
10
11


B
C
A
E
D
F
G
H
time
0
1
2
3
4
5
6
7
8
9
10
11


B
E
H
0
1
2
3
4
5
6
7
8
9
10
11

Earliest-finish-time-first - **Complexity?**

EARLIEST-FINISH-TIME-FIRST $(n, s_1, s_2, \ldots, s_n, f_1, f_2, \ldots, f_n)$

SORT jobs by finish time so that $f_1 \leq f_2 \leq \ldots \leq f_n$

$A \leftarrow \phi$ ⟵ set of jobs selected

FOR $j = 1$ TO n

  IF job j is compatible with A

    $A \leftarrow A \cup \{ j \}$

RETURN A

Interval Scheduling - Earliest-finish-time-first

Is this algorithm optimal?