Quiz Revisions
Tutoring 5-7 Mondays
WICS++: Wednesday, 3/9 in Monroe 311 from 9 pm - 10 pm

Seems like could apply proof to any ordering
How to find best f? Work backwards/Use computers

Goals:
1. Improve our understanding of exchange proof
2. Analyze runtime
3. Dealing with ties
4. Describe Max Weight Independent Set Problem

Q: What is the runtime of the greedy scheduling alg.?

A) $O(1)$ B) $O(n)$ **C) $O(n \log n)$**   D) $O(n^2)$

Dropping w_i/t_i are unique assumption?
where did it show up?

Here → $w_j/t_j \geq w_k/t_k \Rightarrow w_j t_k \geq w_k t_j \Rightarrow w_j t_k - w_k t_j \geq 0$ ←

No contradiction
A might not have improved

New Idea: Keep exchanging

Pf Sketch: Choose some labeling

$w_1/t_1 \geq w_2/t_2 \geq w_3/t_3 \dots$

T	f
1	10
2	10

(1,2), (2,1)

Let $\sigma = (1, 2, 3, \dots, n)$ (Greedy Ordering)
Let σ^* = any other ordering

σ^*: $(\dots k, j \dots)$, $w_j/t_j \geq w_k/t_k$ —exchange k,j→ $\sigma^{*\prime}$: $(\dots k', j' \dots)$, $w_{j'}/t_{j'} \geq w_{k'}/t_{k'}$ → $\sigma^{*\prime\prime}$ → … → σ: $(1, 2, 3, \dots n)$

Bubble Sort

$A(\sigma^*) \geq A(\sigma^{*\prime}) \geq A(\sigma^{*\prime\prime}) \geq \dots \geq A(\sigma)$

What is the runtime of this greedy alg. (no assumption of unique f-values)?

A) $O(1)$ B) $O(n)$ **C) $O(n \log n)$**   ~~D) $O(n^2)$~~

Idea: Boss says "We must optimize $A(\sigma) = \sum_i w_i C_i(\sigma)$"

Thm: Greedy alg with $f = w^2/t$ is optimal for obj. fun. $A(\sigma) = \sum_{i=1}^{n} w_i C_i(\sigma)$.

Pf: Exchange argument (contradiction)

job	time	weight	f	
(2) ~~1~~	5	2	2/5	next
(3) ~~2~~	3	1	1/3	smallest
(1) 3	9	4	4/9	largest

Assume w_i^2/t_i are distinct $\forall\, i \in \{1, 2, \dots, n\}$

WLOG, relable so $w_1^2/t_1 > w_2^2/t_2 > w_3^2/t_3 > \dots w_n^2/t_n$

So let $\sigma = (1, 2, 3, \dots, n)$. This is the greedy ordering. For contradiction, assume σ is not optimal. Let σ^* be the optimal ordering. $\sigma \neq \sigma^*$

Then $\exists$ jobs j, k that are not in numeric order in σ^*

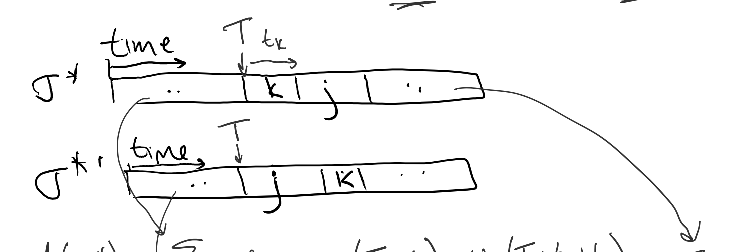
$\sigma^* = (\dots, k, j \dots)$ $j < k$

Let $\sigma^{*\prime}$ be the same as σ^* but with j, k exchanged:

$\sigma^{*\prime} = (\dots, j, k, \dots)$ (same / same)

$A(\sigma) = \sum_{i=1}^{n} w_i C_i(\sigma)$

Q: What is $A(\sigma^*) - A(\sigma^{*\prime})$



$A(\sigma^*) = \sum_i w_i C_i + w_k(T + t_k) + w_j(T + t_k + t_j) + \sum_\ell w_\ell C_\ell$

$A(\sigma^{*\prime}) = \sum w_i C_i + w_j(T + t_j) + w_k(T + t_k + t_j) + \sum_\ell w_\ell C_\ell$

→ $A(\sigma^*) - A(\sigma^{*\prime}) = w_j t_k - w_k t_j$

But $w_j^2/t_j > w_k^2/t_k$ $\overset{\text{mult.}}{\Rightarrow}$ $w_j^2 t_k > w_k^2 t_j \Rightarrow w_j^2 t_k - w_k^2 t_j > 0$

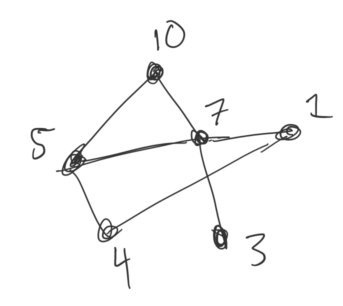
Thus $A(\sigma^*) - A(\sigma^{*\prime}) \overset{?}{>} 0$, a ~~contradiction~~ because this means σ^* is not optimal.
So σ is optimal.

4s. MWIS (Dynamic Programming)

Friday, March 12, 2021
11:44 AM

Max Weight Independent Set Problem (MWIS)

Input: Graph: $G = (V, E)$ (V: set vertices, E: set of edges)
Weights: $w: V \to \mathbb{Z}^+$ ($\mathbb{N}$)



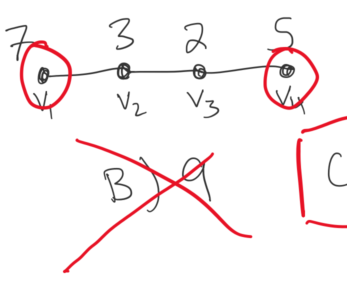
Output: $S \subseteq V$ s.t.
Independent Set ⇒ • If $\{u, v\} \in E$, $u \notin S$ or $v \notin S$
Max weight ⇒ • $W(S) = \sum_{v \in S} w(v)$ is maximized
(objective function)

Applications
- Cell Tower Transmission
- Choose Franchise Location
- Party Invite
- Scheduling

☆

☆

What is $W(S)$ for MWIS S of



$S = \{v_1, v_4\}$

~~A) 8~~   ~~B) 9~~   **C) 12**   ~~D) 17~~

Ind Set	Weight
$\emptyset$	0
$\{v_1\}$	7
$\{v_1, v_3\}$	9
~~$\{v_2, v_3\}$~~	⋮
⋮	⋮