Andrew Rutherford

CSCI 3104

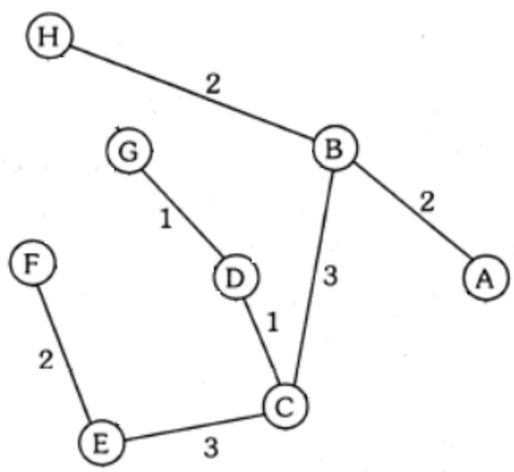
CPU: 2.8 GHz Intel Core i7

Ram: 16 GB 1600 MHz DDR3

OSX Yosemite

Homework #6

On my honor, as a University of Colorado at Boulder student, I have neither given nor received any unauthorized help.



Kruskal	Prim	
CD (1)	A	Visited = {A}
DG (1)	AB (2)	Visited = {A,B}
AB (2)	BH (2)	Visited = {A,B,H}
BH (2)	BC (3)	Visited = {A,B,C,H}
EF (2)	CD (1)	Visited = {A,B,C,D,H}
BC (3)	DG (1)	Visited = {A,B,C,D,G,H}
CE (3)	CE (3)	Visited = {A,B,C,D,E,G,H}
	EF (2)	Visited = {A,B,C,D,E,F,G,H}

Algorithm:

Sort A in order such that $a_1 \geq a_2 \geq \dots \geq a_n$

Sort B in order such that $b_1 \geq b_2 \geq \dots \geq b_n$

Return $\prod_{i=1}^{n} a_i^{b_i}$

Proof: Suppose optimal solution is not produced by this algorithm:

- Let S = optimal solution where a_1 is paired with b_p and a_g is paired with b_1
- S' where a_1 is paired with b_1, a_g is paired with b_p, and all other pairs are the same as S

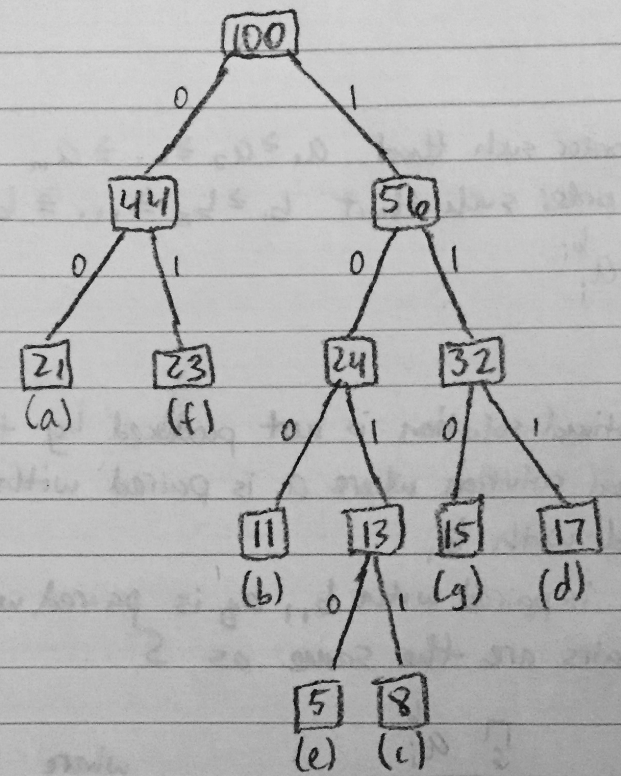
$$\frac{\text{Payoff}(S)}{\text{Payoff}(S')} = \frac{\prod_S a_i^{b_i}}{\prod_{S'} a_i^{b_i}}$$

where $a_1 \geq a_g$ and $b_1 \geq b_p$

$$= \frac{(a_1)^{b_p}(a_g)^{b_1}}{(a_1)^{b_1}(a_g)^{b_p}}$$

$$= \left(\frac{a_1}{a_g}\right)^{b_p - b_1} \rightarrow \frac{\text{Payoff}(S)}{\text{Payoff}(S')} \leq 1$$

This contradicts the assumption that S is the optimal solution.

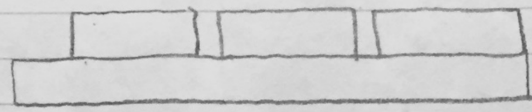


A = 00
B = 100
C = 1011
D = 111
E = 1010
F = 01
G = 110

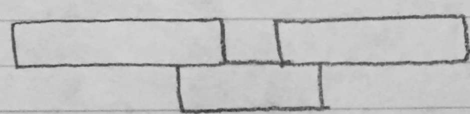
A = $2 \cdot 10^6 \cdot .21 = 420000$
B = $3 \cdot 10^6 \cdot .11 = 330000$
C = $4 \cdot 10^6 \cdot .08 = 320000$
D = $3 \cdot 10^6 \cdot .17 = 510000$
E = $4 \cdot 10^6 \cdot .05 = 200000$
F = $2 \cdot 10^6 \cdot .23 = 460000$
G = $3 \cdot 10^6 \cdot .15 = 450{,}000$

Total = 2,690,000 bits

Earliest Start time - counterexample

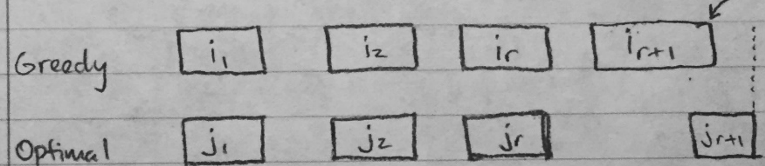


Shortest Interval - counterexample



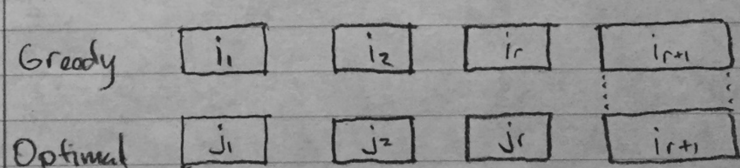
Proof (by contradiction) that greedy algorithm of earliest finish time is optimal:

Outcome 1:



job i_{r+1} finishes before j_{r+1}

Outcome 2:



job i_{r+1} finishe[s] before j_{r+1}

Let $i_1, i_2, \ldots i_k$ = set of jobs using greedy alg.

$j_1, j_2 \ldots j_k$ = set of jobs in optimal solution