8/8

CS 1510
Algorithm Design

Greedy Algorithms

Problems 9 and 10

Due September 5, 2014

Buck Young and Rob Brown

19 /22

Problem 9

(a)
Input: A number n of skiers with heights $p_1, ..., p_n$, and n skies with heights $s_1, ..., s_n$.
Output: The minimal average difference between skier height p_i and ski height $s_{\alpha(i)}$: $\frac{1}{n}\sum_{i=1}^{n}|p_i - s_{\alpha(i)}|$
Theorem: The given "minimized height difference first" algorithm MF is correct.

Proof: Consider a "counter-example" algorithm CE to prove MF is incorrect.

Let T be the total height difference (and $\frac{T}{n}$ be the average total height difference), & H_i be the height difference between a person p_i and a chosen ski $s_{\alpha(i)}$.

Input: Skies = {10, 12, 18, 20}
Skiers = {16, 20, 22, 25}

MF

$H_1 = |P_2 - S_4| = |20-20| = \emptyset$
$H_2 = |P_1 - S_3| = |16-18| = 2$
$H_3 = |P_3 - S_2| = |22-12| = 10$
$H_4 = |P_4 - S_1| = |25-10| = 15$
$\frac{T_{MF}}{n} = 27 \div 4 = 6.75$

$\frac{T_{MF}}{n} = 6.75$

CE

$H_1 = |P_1 - S_1| = |16-10| = 6$
$H_2 = |P_2 - S_2| = |20-12| = 8$
$H_3 = |P_3 - S_3| = |22-18| = 4$
$H_4 = |P_4 - S_4| = |25-20| = 5$
$\frac{T_{CE}}{n} = 23 \div 4 = 5.75$

$\frac{T_{CE}}{n} = 5.75$

Clearly $\frac{T_{MF}}{n} > \frac{T_{CE}}{n}$ and the problem wants us to minimize $\frac{T}{n}$.

Also – let OPT be the optimal solution to this problem – we see that MF < CE ≤ OPT.

∴ The given algorithm MF is sub-optimal and not correct by way of this counter-example.

(b)

Input: A number n of skiers with heights $p_1, ..., p_n$, and n skies with heights $s_1, ..., s_n$.

Output: The minimal average difference between skier height p_i and ski height $s_{\alpha(i)}$: $\frac{1}{n}\sum_{i=1}^{n} |p_i - s_{\alpha(i)}|$

Theorem: The given "shortest-skier to shortest-ski" algorithm SS is correct.

Proof: Assume to reach a contradiction that there exists an input on which SS produces an unacceptable output.

Let SS(I) be the output of SS on input I
& OPT(I) be the optimal solution which agrees with SS(I) the most number of steps

$$SS\ (p_1, s_1) \ldots (p_k, s_\alpha) \ldots (p_\alpha, s_\beta)$$
$$OPT\ (p_1, s_1) \ldots \underbrace{(p_k, s_\beta)}_{k} \ldots (p_\beta, s_\alpha)$$

Let k be the first point in time where the pairing of skier to ski in SS(I) differs from the pairing in OPT(I)

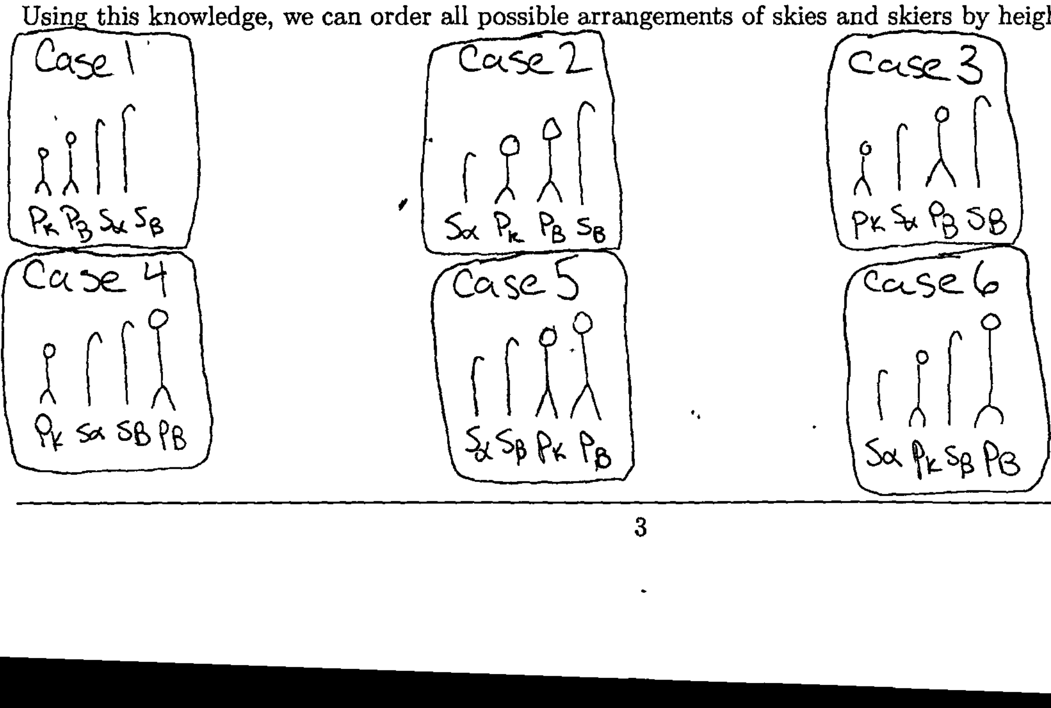
Let OPT'(I) = OPT with s_β and s_α swapped.
Note: p_α could be the same person as p_β but they do not have to be

$$SS\ (p_1, s_1) \ldots (p_k, s_\alpha) \ldots (p_\alpha, s_\beta)$$
$$OPT\ (p_1, s_1) \ldots (p_k, s_\beta) \ldots (p_\beta, s_\alpha)$$
$$OPT'\ (p_1, s_1) \ldots \underbrace{(p_k, s_\alpha)}_{k} \ldots (p_\beta, s_\beta)$$

Clearly OPT' is more like SS at time step k
Ultimately, we want to show that the average height difference between OPT and OPT' has not gotten worse (in order to prove that OPT' is still an acceptable solution).

First we make a few simple observations: by definition of SS, we know that $p_k < p_\beta$ and that $s_\alpha < s_\beta$

Using this knowledge, we can order all possible arrangements of skies and skiers by height:



Thus we have 6 clear cases and for each we must show that the average height difference between OPT and OPT' has not gotten worse. We can do this by using an inequality: $|p_k - s_\alpha| + |p_\beta - s_\beta| \le |p_k - s_\beta| + |p_\beta - s_\alpha|$ (the height difference of OPT' is $\le$ the height difference of OPT)

If we can prove this inequality valid for each of our six cases, we will have proven that OPT' is no worse than OPT. We may remove the absolute values in the inequality by simply reordering the terms to always produce a positive value (based on which of the terms is larger on a case-by-case basis)...

Case 1 : $p_k \le p_\beta \le s_\alpha \le s_\beta$

$$s_\alpha - p_k + s_\beta - p_\beta \le s_\beta - p_k + s_\alpha - p_\beta$$
$$-s_\alpha + p_k - s_\beta + p_\beta \qquad -s_\beta + p_k - s_\alpha + p_\beta$$
$$0 \le 0$$

Check.

Case 2 : $s_\alpha \le p_k \le p_\beta \le s_\beta$

$$p_k - s_\alpha + s_\beta - p_\beta \le s_\beta - p_k + p_\beta - s_\alpha$$
$$+s_\alpha - s_\beta \qquad -s_\beta + s_\alpha$$
$$p_k - p_\beta \le -p_k + p_\beta$$
$$2p_k \le 2p_\beta$$
$$p_k \le p_\beta$$

Check.

Case 3 : $p_k \le s_\alpha \le p_\beta \le s_\beta$

$$s_\alpha - p_k + s_\beta - p_\beta \le s_\beta - p_k + p_\beta - s_\alpha$$
$$+p_k - s_\beta \qquad -s_\beta + p_k$$
$$s_\alpha - p_\beta \le p_\beta - s_\alpha$$
$$2s_\alpha \le 2p_\beta$$
$$s_\alpha \le p_\beta$$

check.

Case 4 : $p_k \le s_\alpha \le s_\beta \le p_\beta$

$$s_\alpha - p_k + p_\beta - s_\beta \le s_\beta - p_k + p_\beta - s_\alpha$$
$$+p_k - p_\beta \qquad +p_k - p_\beta$$
$$s_\alpha - s_\beta \le s_\beta - s_\alpha$$
$$2s_\alpha \le 2s_\beta$$
$$s_\alpha \le s_\beta$$

Check.

Case 5 : $s_\alpha \le s_\beta \le p_k \le p_\beta$

$$p_k - s_\alpha + p_\beta - s_\beta \le p_k - s_\beta + p_\beta - s_\alpha$$
$$-p_k + s_\alpha - p_\beta + s_\beta \qquad -p_k + s_\beta - p_\beta + s_\alpha$$
$$0 \le 0$$

check.

Case 6 : $s_\alpha \leq p_k \leq s_\beta \leq p_\beta$

$$p_k - s_\alpha + p_\beta - s_\beta \leq s_\beta - p_k + p_\beta - s_\alpha$$
$$+ s_\alpha \quad - p_\beta \qquad\qquad - p_\beta \quad + s_\alpha$$
$$p_k - s_\beta \leq s_\beta - p_k$$
$$2p_k \leq 2s_\beta$$
$$p_k \leq s_\beta \quad \text{check,}$$

Our inequality is valid for every case, thus we have shown that the height difference between OPT and OPT' has not gotten any worse.

∴ We have reached a contradiction because OPT' is still a valid solution and it agrees with SS for one additional step despite OPT being defined as agreeing for the most number of steps.

Problem 10

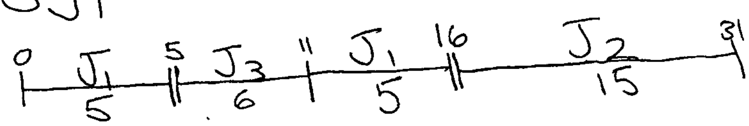
(SJF)
Input: A collection of jobs $J_1, ..., J_n$, where the ith job is a tuple (r_i, x_i) of non-negative integers specifying the release time r and size of the job x.
Output: A preemptive feasible schedule on one processor that minimizes the total completion time $\sum_{i=1}^{n} C_i$
Theorem: That the given "shortest job first" algorithm SJF is correct.

Proof: Consider a "counter-example" algorithm CE to prove SJF is incorrect.

Let C be the total completion time and C_i be the completion time for job i.

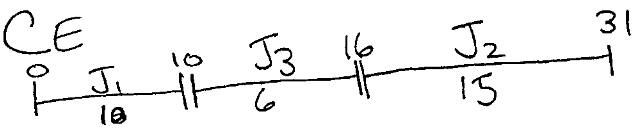
Input: $\{J_1 = (0, 10), J_2 = (3, 15), J_3 = (5, 6)\}$

SJF



$C_1 = 16$
$C_2 = 31$
$C_3 = 11$
$C_{SJF} = 58$

CE



$C_1 = 10$
$C_2 = 31$
$C_3 = 16$
$C_{CE} = 57$

Clearly $C_{SJF} > C_{CE}$ and the problem requires us to minimize C.

Also – let OPT be the optimal solution to this problem – we see that SJF < CE ≤ OPT.

∴ The given algorithm SJF is sub-optimal and not correct by way of this counter-example.

(SRPT)

Input: A collection of jobs $J_1, ..., J_n$, where the ith job is a tuple (r_i, x_i) of non-negative integers specifying the release time r and size of the job x.

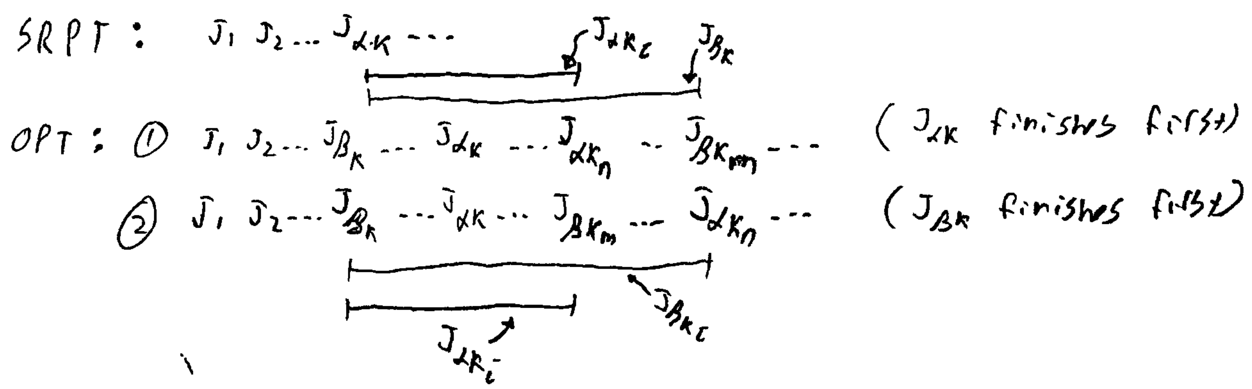
Output: A preemptive feasible schedule on one processor that minimizes the total completion time $\sum_{i=1}^{n} C_i$.

Theorem: The given "shortest remaining processing time" algorithm SRPT is correct.

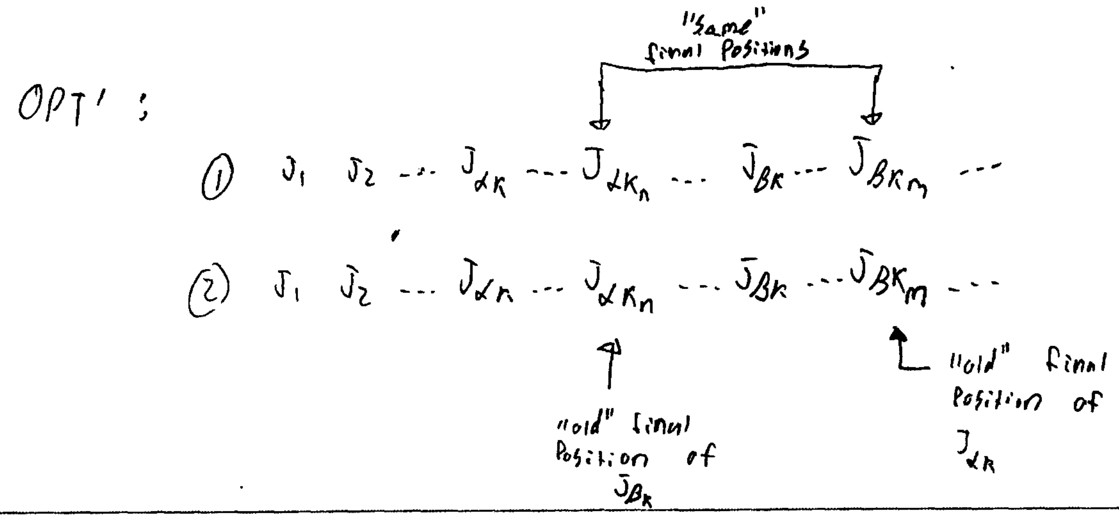
Proof:

Suppose, for the sake of reaching a contradiction, SRPT produces unacceptable output for some input I. Let $SRPT(I) = \{J_{\alpha 1} J_{\alpha 2}, ..., J_{\alpha n}\}$ be the output of SRPT, and and $OPT(I) = \{J_{\alpha 1} J_{\alpha 2}, ... J_{\alpha n}\}$ be the output of some optimal algorithm which agrees with SRPT for the most number of steps. For each $J_i \in$ OUTPUT, J_i identifies a job that executes for a single unit of time and r_i is the remaining number of time units required for job completion. ~~[illegible]~~.

Consider the output of these two algorithms. Let k be the first point of disagreement, where SRPT schedules $J_{\alpha k}$ which completes at $J_{\alpha k_n}$ and OPT schedules $J_{\beta k}$ which completes at $J_{\beta k_m}$. This disagreement can take the two following forms.



Let OPT' be OPT with each $S_{\alpha k_i} \in \{S_{\alpha k_1}, ..., S_{\alpha k_n}\}$ swapped with $S_{\beta i} \in \{S_{\beta k_1}, ..., S_{\beta k_m}\}$ until every $S_{\beta i}$ appears AFTER every $S_{\alpha i}$ in the schedule. Note that $r_{\alpha k} = |\{S_{\alpha k}, ..., S_{\alpha k C}\}|$ and $r_{\beta k} = |\{S_{\beta k}, ..., S_{\beta k C}\}|$ and $r_{\alpha k} \leq r_{\beta k}$. Consider OPT' for the two general forms above



CASE 1

$C'_\alpha \leq C_\alpha$ since $J_{\alpha k_C}$ no longer needs to wait for any $J_{\beta i}$, and $C'_\beta = C_\beta$ since $r_\alpha \leq r_\beta$ (and $J_{\beta k_C}$ is therefore never swapped). Thus $C'_\alpha + C'_\beta \leq C_\alpha + C_\beta$. Since no intervals were inserted/created (only swapped) OPT' has not affected any other completion times, and this translates into $\sum_{i=1}^{n} C_i$ being minimized for OPT'.

CASE 2

$C'_\alpha \leq C_\beta$ (in words, $J_{\alpha k} \epsilon OPT'$ completes at or before $J_{\alpha k} \epsilon OPT$) since all $J_{\alpha k_i}$ are swapped until they appear before all $J_{\beta k_i}$; and $C'_\beta = C_\alpha$ since the position of $J_{\alpha k_n}$ in OPT is swapped with some $J_{\beta k_i}$ to become the new completion time of $J_{\beta k}$ in OPT'). Thus $C'_\alpha + C'_\beta \leq C_\alpha + C_\beta$, and as in CASE 1 since no intervals were created/inserted, this translates into $\sum_{i=1}^{n} C_i$ being minimized for OPT'.

Clearly, OPT' is more like SRPT than OPT. We have also demonstrated the correctness of OPT' in the two feasible scenarios of disagreement between OPT and SRPT.

∴ We observe a contradiction implying the correctness of SRPT.