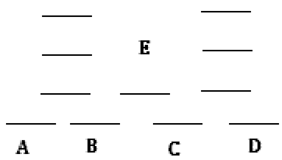
CS130B, Winter 2009

Solutions to the Greedy Practice Problems

1. This algorithm does not solve the problem of finding a maximum cardinality set of non-overlapping intervals. Consider the following intervals:



Obviously, the optimal solution is $\{A, B, C, D\}$. However, the interval that overlaps with the fewest others is E, and the algorithm will select E first, which precludes it from picking intervals B and C.

2. The greedy algorithm is not optimal for the problem of making change with the minimum number of coins when the denominations are $1, 5, 10, 20, 25$, and 50. In order to make 40 Shillings, the greedy algorithm would use three coins of 25, 10, and 5 shillings. The optimal solution is to use two 20- shilling coins.

2
(a) This algorithm is INCORRECT for the problem of minimizing the average difference between the heights of skiers and their skis. Consider an instance with the following values:
$p_1 = 5, p_2 = 10$, and $s_1 = 9, s_2 = 14$.
The algorithm would pair $p1$ with s_2 and p_2 with s_1 for a total cost of $1/2(1 + 9) = 5$. Pairing p_1 with s_1 and p_2 with s_2 yields a total cost of $1/2(4 + 4) = 4$.
(b) This algorithm is CORRECT. The proof is by contradiction. Assume the people and skis are numbered in increasing order by height. If the greedy algorithm is not optimal, then there is some input $p_1, ..., p_n, s_1, ..., s_n$ for which it does not produce an optimal solution. Let the optimal solution be $T = \{(p_1, s_{j(1)}), ..., (p_n, s_j(n))\}$, and let the output of the greedy algorithm be $G = \{(p_1, s_1), ..., (p_n, s_n)\}$. Beginning with p_1, compare T and G. Let p_i be the first person who is assigned different skis in G than in T . Let s_j be the pair of skis assigned to p_i in T . Create solution T' by switching the ski assignments of p_i and p_j . By the definition of the greedy algorithm, $s_i \leq s_j$. The total cost of T' is given by

$$Cost(T') = Cost(T) - 1/n(|p_i - s_j| + |p_j - s_i| - |p_i - s_i| - |p_j - s_j|).$$

There are six cases to be considered. For each case, one needs to show that

$$(|p_i - s_j| + |p_j - s_i| - |p_i - s_i| - |p_j - s_j|) \geq 0.$$

Case 1: $p_i <= p_j <= s_i <= s_j$.
$|p_i - s_j| + |p_j - s_i| - |p_i - s_i| - |p_j - s_j| = (s_j - p_i) + (s_i - p_j) - (s_i - p_i) - (s_j - p_j) = 0.$

Case 2: $p_i \leq s_i \leq p_j \leq s_j$.

$|p_i - s_j| + |p_j - s_i| - |p_i - s_i| - |p_j - s_j| = (s_j - p_i) + (p_j - s_i) - (s_i - p_i) - (s_j - p_j) = 2(p_j - s_i) >= 0.$

Case 3: $p_i <= s_i <= s_j <= p_j$.
$|p_i - s_j| + |p_j - s_i| - |p_i - s_i| - |p_j - s_j| = (s_j - p_i) + (p_j - s_i) - (s_i - p_i) - (p_j - s_j) = 2(s_j - s_i) >= 0$
Case 4: $s_i <= s_j <= p_i <= p_j$.
$|p_i - s_j| + |p_j - s_i| - |p_i - s_i| - |p_j - s_j| = (p_i - s_j) + (p_j - s_i) - (p_i - s_i) - (p_j - s_j) = 0.$

Case 5: $s_i <= p_i <= s_j <= p_j$.
$|p_i - s_j| + |p_j - s_i| - |p_i - s_i| - |p_j - s_j| = (s_j - p_i) + (p_j - s_i) - (p_i - s_i) - (p_j - s_j) = 2(s_j - p_i) >= 0$

Case 6: $s_i <= p_i <= p_j <= s_j$.
$|p_i - s_j| + |p_j - s_i| - |p_i - s_i| - |p_j - s_j| = (s_j - p_i) + (p_j - s_i) - (p_i - s_i) - (s_j - p_j) = 2(p_j - p_i) >= 0$

4. This algorithm is correct for the problem of minimizing the total sum of all line penalties. The proof is by contradiction. Assume there is an optimal solution T, and call the output of the greedy algorithm G. Let s_i be the penalty of the ith line of solution S. Let j be the number of the first line in T that is different from the jth line in G. By the definition of the algorithm, $g_i < t_i$. Create a new solution T' by moving the first word of line $i + 1$ in T to the end of line i. Let l be the length of this word. Note that $t'_{i+1} = t_{i+1} + l$ and $t'_i = t_i - l$.
Therefore, the sum of all line penalties in T' is the same as the sum of all line penalties of T. Contradiction.