CISC-365*
Test #3
October 31, 2008

Student Number (Required) ____________________

Name (Optional) Solutions

This is a closed book test. You may not refer to any resources.

This is a 50 minute test.

Please write your answers in ink. Pencil answers will be marked, but may not be reconsidered after the test papers have been returned.

The test will be marked out of 50.

Question 1	/20
Question 2	/30
TOTAL	/50

Question 1 (20 marks)

You have won the contract to install Wi-fi nodes along a very straight and sparsely populated stretch of road which runs due east and west. There are N houses along the road – each house is identified by its distance from the east end of the road. Your assignment is to install nodes so that each house is no more than 1 kilometer from a node. You want to install **as few nodes as possible**.

a) **(10 marks)** Give a Greedy Algorithm to find an optimal (minimal) set of locations for the wi-fi nodes.

1. Sort the houses by distance from the East end - closest first
2. Repeat
3. Place a node 1 km west of the first house in the list or at the West end of the road, whichever is reached first
4. Skip over all houses that are within 1 km of the node just placed
5. Until there are no houses left in the list

b) **(10 marks)** Prove that the first choice your algorithm makes for a node location is correct (i.e. that there is an optimal solution that contains this location as its first location).

Let A be the alg's solution, and let X be the first location chosen by the alg. If $x \in S$ we are done, so assume Let S be an optimal solution, that S does not contain X. Let V be the east-most location in S. Note that $V < X$, since if $V > X$ then the east-most house is not covered. Consider $S' = S - \{V\} + \{X\}$. Since all houses covered by V are also covered by X, S' is a solution. Since $|S'| = |S| - 1 + 1 = |S|$, S' is optimal.

$\therefore$ there exists an optimal solution containing the alg's first choice.

Question 1 (20 marks)

You have won the contract to install Wi-fi nodes along a very straight and sparsely populated stretch of road which runs due east and west. There are N houses along the road – each house is identified by its distance from the east end of the road. Your assignment is to install nodes so that each house is no more than 1 kilometer from a node. You want to install **as few nodes as possible**.

a) **(10 marks)** Give a Greedy Algorithm to find an optimal (minimal) set of locations for the wi-fi nodes.

-2 if they forget to sort

-1 if they forget to check for the end of the road

~5/10 if they have a vague idea of how to do a Greedy Algorithm

- some students apparently assumed the wi-fi nodes had to go in the houses – that's ok, but the answer changes a bit

b) **(10 marks)** Prove that the first choice your algorithm makes for a node location is correct (i.e. that there is an optimal solution that contains this location as its first location).

key points in the proof

3 - start with an optimal solution that doesn't contain the alg's 1st choice

3 - remove the 1st node from that solution, + replace it with the alg's 1st choice

3 - show that the resulting set is a solution + that it has the same size as the optimal solution, so it is optimal too

1 - conclude that ∃ an optimal solution that contains the alg's 1st choice

a) **(18 marks)** Here is a Greedy Algorithm:

1. Sort the performances according to END time, so that the performance that ends earliest is the first in the list.
2. Select the first performance in the list
3. Repeat
4. Select the next performance in the list that does not overlap with any performance selected so far.
5. Until there are no more performances to be considered

Does this algorithm find an optimal solution for all instances of this problem? Either find a counterexample (a set of performance times for which this algorithm chooses a non-optimal solution) or give a proof that the algorithm is correct. If you are providing a counterexample, show the solution the algorithm would choose and explain why, and show that there is a better solution. If you are proving correctness, you may focus on proving that the algorithm's first choice (Line 2) is appropriate.

Yes, it finds an optimal solution.

To prove that the first choice the algorithm makes is in an optimal solution, let A be the alg's solution and let X be the first performance in A. Let S be an optimal solution - if S contains ~~X~~ we are done, so suppose S does not contain X. Let V be the earliest-ending performance in S. Consider $S' = S - \{V\}$. All performances in S' start at times that are $\geq$ the end time of V, and therefore their start times are all $\geq$ the end time of X (X has the earliest end time). So $S'' = S' + \{X\}$ is a valid solution, and $|S''| = |S|$ so S'' is optimal.

$\therefore$ $\exists$ an optimal solution containing the algorithm's first choice.

a) **(18 marks)** Here is a Greedy Algorithm:

1. Sort the performances according to END time, so that the performance that ends earliest is the first in the list.
2. Select the first performance in the list
3. Repeat
4. Select the next performance in the list that does not overlap with any performance selected so far.
5. Until there are no more performances to be considered

Does this algorithm find an optimal solution for all instances of this problem? Either find a counterexample (a set of performance times for which this algorithm chooses a non-optimal solution) or give a proof that the algorithm is correct. If you are providing a counterexample show the solution the algorithm would choose and explain why, and show that there is a better solution. If you are proving correctness, you may focus on proving that the algorithm' first choice (Line 2) is appropriate.

Same as for 1b, but with point values

~~[illegible]~~ 5, 5, 5, 3

b) **(12 Marks)** Here is a different Greedy Algorithm:

1. Sort the performances according to LENGTH, so that the shortest performance is the first in the list
2. Select the first performance in the list
3. Repeat
4. Select the next performance in the list that does not overlap with any performance selected so far
5. Until there are no more performances to be considered

Does this algorithm find an optimal solution for all instances of this problem? Either find a counterexample (a set of performance times for which this algorithm chooses a non-optimal solution) or give a proof that the algorithm is correct. If you are providing a counterexample, show the solution the algorithm would choose and explain why, and show that there is a better solution. If you are proving correctness, you may focus on proving that the algorithm's first choice (Line 2) is appropriate.

No it doesn't. As a counter-example, consider the example given in the question. The algorithm given here will start by choosing the shortest performance: #5, and then #2, the second shortest. After that no more performances can be chosen.

However the optimal solution in this case is to choose performances 2, 4 & 6.

b) **(12 Marks)** Here is a different Greedy Algorithm:

1. Sort the performances according to LENGTH, so that the shortest performance is the first in the list
2. Select the first performance in the list
3. Repeat
4. Select the next performance in the list that does not overlap with any performance selected so far
5. Until there are no more performances to be considered

Does this algorithm find an optimal solution for all instances of this problem? Either find a counterexample (a set of performance times for which this algorithm chooses a non-optimal solution) or give a proof that the algorithm is correct. If you are providing a counterexample, show the solution the algorithm would choose and explain why, and show that there is a better solution. If you are proving correctness, you may focus on proving that the algorithm's first choice (Line 2) is appropriate.

- giving a set that is a counter-example - 5

- showing what the alg would pick + why - 5

- showing a better solution - 2