Greedy algorithms (Chapter 4)

A possibility:

In the last lecture we tried to implement the algorithm on interval scheduling to run in time proportional to the size of the output. I discussed some ideas towards this objective. If you are interested in this problem, please see me. We together can look into this issue a bit more.

A correction:

In the last lecture I described an algorithm for the interval partitioning problem:

Given: A set of lecture intervals $[s_i, f_i], i = 1, 2, ..., n$.

Output: Schedule these lectures in minimum number of classrooms so that no two lectures occur at the same time in the same room.

Unfortunately the algorithm when applied to the following example will not produce the optimal result. Clearly, only two classrooms are needed. The example was generated by Lvyu Ye (a student in the class).

```
---------------
  ----       ---------------------------------
                  -----    --------   ---
```

The following algorithm is described in the text. I am using a simpler language.

```
Sort the intervals by their start time.
\*Let s(1), s(2), ..., s(n) be the sorted sequence.
\* We use (i) to indicate the interval with starting time s(i).}

m = 0 \* number of classrooms allocated

for i = 1 to n do
   if (there is a classroom j such that (i) is compatible
                               with the lectures in classroom j)
```

```
        schedule (i) in classroom j
    else {
      assign (i) in classroom m + 1
      m = m+1
    {
endfor
```

If, for each classroom j, we maintain the maximum of the finish times (given by the last job assigned to j) of the lectures scheduled in classroom j, the above algorithm can be implemented in $O(nlogn) + O(mn)$ where the first term is for the sorting and the second part is for the for-loop. If we store the classrooms in a priority queue (a heap structure) with keys being the maximum of the finish times, the above algorithm takes $O(nlogn)$ time in the worst case.(Why? You should figure this out.)

Read the text for the proof of optimality.

Problems: Try to work out the following problems. I will not provide solutions to these questions. You have to try yourself and for help, you can see me, the TA or send us e-mails with specific questions. You can also ask me in the class.

1. (Selecting break points) There are refuelling stations at certain points (random) along the Trans Canada Highway from Vancouver to Montreal. Suppose you are driving a car with fuel efficiency of C kms per tankful. Design a strategy such that the car makes minimum refuelling stops for the entire trip. The car starts from Vancouver with a tankful of gas. You need to show that your strategy is optimal.
2. Let X be a set of intervals on the real line. We say that a set P of points *pierce* X if every interval interval in X contains at least one point of P. We would like to select a minimum size set of points that pierce X. If you use a greedy algorithm, show that it is correct.
3. Problems 3, 5 of Chapter 4 of the text.