Due: Wednesday, October 6, beginning of lecture

NOTE: Each problem set only counts 5% of your mark, but it is important to do your own work (but see below). These assignments will be followed by term tests, each worth 15% of your final grade. You may consult with others concerning the general approach for solving problems on assignments, but you must write up all solutions entirely on your own. As an experiment you may chose to work in pairs and then submit one assignment. Anything else is *plagiarism*, and is subject to the University's Code of Behavior. You will receive 1/5 points for any question/subquestion for which you say "I do not know how to answer this question". You will receive .5/5 points if you just leave the question blank.

1. Problem 4 of chapter 1 (page 19). You need only show how to modify the stable marriage algorithm given in section 1.1 (pages 5,6) but you should think about how to modify the proof showing the correctness of the algorithm so as to convince yourself that your algorithm will work.

2. Problem 8 of chapter 3 (pages 66,67).

3. Problem 7 of chapter 4 (pages 113,114). In addition to giving an algorithm, indicate how you would proceed to argue that your algorithm always provides an optimal schedule.

4. Problem 16 of chapter 4 (page 118).
   Hint: The "most obvious greedy algorithm" does not work.

5. Consider the unit profit job scheduling problem where the i^{th} job has release time r_i, processing time p_i and deadline d_i. Say for definitness that all parameters are positive integers. A feasible schedule is a mapping $\sigma : \{1, \ldots, n\} \rightarrow \{1, \ldots, max_j d_j\} \cup \{\infty\}$. That is, σ schedules job j to start at time $\sigma(j)$ if $\sigma(j) \neq \infty$ and does not schedule jobs with $\sigma(j) = \infty$. A feasible schedule σ satisfies the properties:
   1) $\sigma(j) \neq \infty$ and $\sigma(k) \neq \infty$ implies $[\sigma(j), \sigma(j) + p_j) \cap [\sigma(k), \sigma(k) + p_k) = \emptyset$
   2) $\sigma(j) \neq \infty$ implies $r_j \leq \sigma(j)$
   3)$\sigma(j) \neq \infty$ implies $\sigma(j) + p_j \leq d_j$.

   Define the ECF greedy algorithm for job scheduling on one processor as follows: In each iteration the algorithm gives highest priority to that job (if any) which has the earliest possible completion time consistent with jobs already scheduled. When a job can be scheduled, ECF schedules it as early as possible. (ECF for job scheduling is a generalization of EDF for interval scheduling.)

   The objective is to maximize the number of jobs scheduled in a feasible schedule σ.

   (a) Provide code (similar to that presented in the text) for the ECF job scheduling algorithm.

   (b) Show that there is some input set $\mathcal{I}$ for which $|OPT(\mathcal{I})| \geq 2 \cdot |ECF(\mathcal{I})|$ where $|A(\mathcal{I})|$ denotes the number of jobs scheduled by algorithm A on input set $\mathcal{I}$.

(c) Show that ECF always achieves a 2-approximation of the optimal; that is, $\forall \mathcal{I}\ |OPT(\mathcal{I})| \leq 2 \cdot |ECT(\mathcal{I})|$.

(d) Define the $static - ECF$ algorithm as follows: The algorithm first sorts jobs so that $r_1 + p_1 \leq r_2 + p_2 \leq ... r_n + p_n$. Then the algorithm tries to schedule each job as soon as poosible given the current schedule. Show that there is no constant c: $static - ECF$ is a c-approximation algorithm.