Assignment #2:

1. Consider scheduling problem with one machine and n jobs. We are given p_j = processing time of job j; d_j = due date for job j. We want to minimize the number tardy. Here is a proposed algorithm: [This is what was described in class]

   1. Sort the jobs and renumber so that $d_1 \leq d_2 \leq ... \leq d_n$
   2. Set $J \leftarrow \phi; \bar{J} \leftarrow \{1, 2, ..., n\}; J^d \leftarrow \phi; k \leftarrow 1 // J$ is the set of currently included jobs; J^d is the set of deleted jobs; $\bar{J}$ is the set yet to be processed.
   3. $J \leftarrow J \cup \{k\}; \bar{J} \leftarrow \bar{J} - \{k\}$
   4. **if** $\sum_{j \in J} p_j > d_k$
   5.   **then let** $p_l = \max_{j \in J} p_j$
   6. $J \leftarrow J - \{l\}; J^d \leftarrow J^d \cup \{l\}$
   7.   **if** $k < n$ **then** $k \leftarrow k + 1$ and go to Step 4.
   8.   **else return** J.
   9. **else** $k \leftarrow k + 1$ and go to Step 4.

   Show that this algorithm is correct by proving the invariant property that at any step we have the maximum number of jobs meeting deadlines from among jobs $\{1, 2, ..., k\}$.

2. Consider a three machine flowshop with the following data

$\frac{Jobs \rightarrow}{Machines \downarrow}$	1	2	3
1	1	10	3
2	14	4	1
3	2	1	12

   Does permitting preemption reduce $C_{\max}$ when $r_j = 0 \forall j$? Find an optimal schedule.

3. Show in as much generality as possible the following statement: "Given that preemptions are permitted, there exists an optimal schedule without inserted idle time." You may assume regular performance measures.

4. If preemptions are not permitted, give an example to show that inserted idle time may improve the schedule.