1. (a) Design a greedy algorithm for the m-machine interval scheduling algorithm.

[10 points]

SOLUTION:
Sort so that $f_1 \leq f_2 \ldots \leq f_n$
For $j = 1..m$
  $F_j := 0$ %initialize the current finishing time for each machine
EndFor
For $i = 1..n$
  IF there exists $F_j \leq s_i$
   then $\sigma(i) = argmax[F_j : F_j \leq s_i]$;
     $F_{\sigma(i)} := f_i$ % assign interval i to machine causing the least gap
   else $\sigma(i) = 0$
  EndIF
EndFor

(b) Use an inductive argument to show that after the i^{th} iteration of the algorithm ($i \geq 0$), the partial solution $\sigma_i : \{1, \ldots, i\} \rightarrow \{0, 1, \ldots, m\}$ is promising and hence proving that the algorithm is optimal.
Hint: In one of the cases for the induction step, the exchange will involve more than one interval.

[10 points]

SOLUTION: Let σ_i denote the mapping of jobs to intervals after the i^{th} iteration (for $0 \leq i \leq n$). Need to show by induction that σ_i can be extended to an optimal mapping OPT_i. The induction is just as in the one machine case but now there is an additional case. Namely, suppose the greedy algorithm places interval $i+1$ on (say) machine 1 while OPT_i has interval $i+1$ on (say) the k^{th} machine. Now we obtain OPT_{i+1} by swapping all the intervals in OPT_i after interval i on machines 1 and k.

(c) Bonus question: Can you prove the optimality of your algorithm using a "charging argument". Hint: I don't know the answer to this question.

[15 points]

2. (a) Let EFT-colour denote the greedy colouring algorithm that first sorts so that $f_1 \leq f_2 ... \leq f_n$ and then greedily colours using the lowest non-conflicting colour. Give an input instance for which EFT-colour is not optimal. [10 points]

SOLUTION:

```
aaaaaaa       aaaaaa      aaaaaaaa      aaaaaaaaaa
   bbbbbb                             bbbbbbb
      ccccccccccccccccc
                     dddddddddddddddddddddddddddddddd
```

A simpler example

```
aaaaaaa             aaaaaa
   bbbbbbbb
      cccccc
         ddddddddddddddddddddddd
```

(b) Bonus question: The student who obtains the best lower bound on the approximation ratio for EFT-colour (ie exhibits an input instance with the worst ratio) will receive an additional 10 points. If (essentially) the same input instance is obtained by more than one person that these bonus marks will be split equally.

(c) Bonus question: Can you derive an upper bound on the approximation ratio for EFT-colour; that is, an upper bound on a constant c such that EFT-colour$[\mathcal{I}] \leq c * OPT[\mathcal{I}]$ for every input set $\mathcal{I}$. [15 points]

(d) Exhibit a graph which is not a perfect graph. [5 points]
SOLUTION:
A cycle on 5 nodes has clique number 2 and chromatic number 3.

3. This question concerns the computation of a MST. Let $G = (V, E)$ be a connected graph with edge weights $\{c(e) | e \in E\}$.

(a) Suppose all the edge weights are distinct and let $v \in V$ be an arbitrary vertex. Prove or disprove the following statement: The MST must contain at least one minimum cost edge adjacent to v. [10 points]
SOLUITION: The statement is true. If all edge weights are distinct then the MST is unique and Kruskal's algorithm is computing this unique MST. In Kruskal's algorithm, when an edge adjacent to v is first taken, it must be a minimum weight edge.

(b) Suppose $c(e_1) < c(e_2) < c(e_3) \ldots < c_{m-1} = c_m$. That is, all edge weights are distinct except for the largest two edge weights which are equal. Prove or disprove the following statement: The MST is unique. [10 points]
SOLUTION: The statement is false. Consider a cyle where the edges (u, v) and (v, w) adjacent to some v in the cycle have the largest equal weight. Kruskal's algorithm will form a path between u and w and then have a choice of adding (u, v) or (v, w) to complete the MST.
What would happen if the two smallest edge weights were equal and all other edge weights were distinct?

4. We wish to prove the following result: Consider a set of symbols $a, b, c, d, \ldots$ with probabilities $f_a, f_b, \ldots.$ summing to 1 where at least one probability is strictly great than $\frac{2}{5}$. Show that the optimal Huffman prefix tree contains a leaf at distance 1 from the root.

   (a) Show that this result is true when there are exatly four symbols. Note that the result is trivial when there are two or three symbols. [10 points]
   SOLUTION: If the Huffman tree did not contain a leaf at distance 1 from the root then the tree would be a balanced tree of depth 2.

```
     G
    * *
   E   F
  * * * *
  a b c d
```

   Let the leaves have probabilities $f_a \leq f_b \leq f_c \leq f_d$ with $f_d > 2/5$. Then $f_a + f_b + f_c < 3/5$, $f_a + f_b = f_E \geq f_d > 2/5$ (or else c would have been matched with node E). This implies that $f_c < 1/5$. But f_a and f_b are both $\leq f_c$ so that $2/5 < f_a + f_b \leq 2 \cdot f_c$ which implies $f_c > 1/5$ and hence a contradiction.

   (b) Argue that this result holds when there are more than four symbols using the fact that it is true for four symbols. [10 points]