Name ____________ Student No. ____________ TA or Tutorial room ____________

AIDS ALLOWED: One page (two sides) of handwritten notes
Answer ALL questions on test paper. Use backs of sheets for additional space. For every question, briefly justify your answer. You can use knowledge from the problem set as part of any answer.

REMINDER: You get 20% for any question or subquestion if you state "I do not know how to answer this question". You get 10% for any question which you just leave blank.

1. (20 points)

   Consider the following maximization problem:
   Input: An exact 3CNF formula $F = C_1 \wedge C_2 \ldots \wedge C_m$. The goal is to maximize the number of clauses satisfied by at least 2 literals.

   (a) (15 points)
   Sketch a randomized algorithm such that the expected number of clauses returned (i.e. satisfied by at least 2 variables) is $m/2$. Briefly explain why the expected number of clauses is $m/2$.

   (b) (5 points)
   By what method can you convert your randomized algorithm into a deterministic algorithm that always returns at least $m/2$ clauses.

2. (20 points)

   Consider the following *2 from 4 frequency set cover problem*: We are given a collection of sets $\mathcal{S} = \{S_1, \ldots, S_m\}$ for $S_i \subseteq U$ with the property that every $u \in U$ occurs in exactly four different sets S_i in $\mathcal{S}$. There is also a (positive valued) cost function $c : \mathcal{S} \to \mathbb{R}^+$ and we let c_i denote the cost of set S_i. A feasible solution is a sub-collection $\mathcal{S}' \subseteq \mathcal{S}$ such that every $u \in U$ occurs in at least two different sets S_i in $\mathcal{S}'$. The goal is to find a feasible solution $\mathcal{S}'$ so as to minimize the cost $c(\mathcal{S}') = \sum_{i:S_i \in \mathcal{S}'} c_i$.

   (a) (10 points) Formulate the *2 from 4 frequency set cover problem* as a $\{0, 1\}$ IP

   (b) (10 points) Show how to use LP relaxation + rounding to obtain a c-approximation algorithm for some constant c. What constant c can you obtain? Explain why your rounded solution is a feasible solution to the IP and why it provides a c-approximation.

3. (20 points) Consider following knapsack decision problem:
An input item $I_j = (s_j, v_j)$ where s_j is the size of the item and v_j is its value. All parameters are positive integers say represented in binary. Let
$L = \{(I_1, \ldots, I_n, W, V)$: $\exists\ \mathcal{I}$ such that $\sum_{I_j \in \mathcal{I}} s_j \leq W$ and $\sum_{I_j \in \mathcal{I}} v_j \geq V\ \}$. Note that L is obviously in NP.

   (a) (10 points)
   Show how to transform the SUBSET-SUM problem to the knapsack decision problem thereby showing that the knapsack decision problem is NP-hard (and therefore also NP complete).

   (b) (10 points)
   The knapsack optimization problem is to find an optimal subset $\mathcal{I}$. That is, given input $\{I_1, \ldots, I_n, W\}$, the goal is to find a subset $\mathcal{I} \subseteq \{I_1, \ldots, I_n\}$ such that $\sum_{I_j \in \mathcal{I}} s_j \leq W$ so as to maximize $\sum_{I_j \in \mathcal{I}} v_j$. Show how to polynomial time reduce the knapsack optimization problem to the knapsack decision problem. Hint: first find the optimum value of a feasible subset.