This exam contains this cover (page 1), four pages of problems (pages 2-5), and one blank page (page 6).

You have 60 minutes.

No notes, calculator, computer, etc. are allowed.

I will not be available during the exam to answer questions. If you think that a problem is incorrectly or ambiguously stated, then make a good-faith effort to interpret the intent of the problem, and explain your interpretation in your solution.

Except on TRUE/FALSE/PUNT questions, always show your work or otherwise explain your answer. A correct answer with no supporting argument may not earn much credit.

Good luck.

1. Give a proposition in disjunctive normal form (the $\vee$ of a bunch of clauses, each of which is a $\wedge$ of variables and their negations) that is logically equivalent to $(p \vee (q \Rightarrow r)) \wedge (q \vee r)$.

2. For each question, answer TRUE, FALSE, or PUNT (not just T, F, or P). No justification is needed. PUNT earns half credit, the correct answer earns full credit, and the incorrect answer earns no credit.
A. The product of two rational numbers is always rational.

B. Using the standard algorithm, addition of $n \times n$ matrices takes time $\Theta(n)$.

C. Any algorithm to compute the nth Fibonacci number uses time $\Omega((3/2)^n)$.

D. About people: Let $m(x, y)$ be "x is the mother of y". Then $\exists y \, \forall x \, m(x, y)$.

E. $\begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{bmatrix} \begin{bmatrix} 6 & 5 \\ 4 & 3 \\ 2 & 1 \end{bmatrix} = \begin{bmatrix} 20 & 56 \\ 14 & 41 \end{bmatrix}$.

3. Prove that for all $n \geq 0$, $\displaystyle\sum_{k=0}^{n} k k! = (n+1)! - 1$.

4. The *median* of a list of numbers is the number m in the list such that half of the numbers are less than or equal to m, and half are greater than or equal to m. There exists an algorithm to compute the median of a list of length n in time $\Theta(n)$. The *mystery sort* algorithm sorts a list of n distinct numbers as follows. If $n \leq 2$, then it sorts the list by bubble sort. If $n > 2$, then it finds the median m of the list using the median algorithm. It scans through the list, splitting it into a list L of numbers less than m and a list G of numbers greater than m. It recursively sorts each of these lists. Finally it joins the sorted L, m, and the sorted G into a list, and outputs that list.

A. Write a recurrence relation for the running time $T(n)$ of mystery sort on a list of length n.

B. Use the master theorem to compute the running time.

5. For any set X, the *power set* is defined to be the set of all subsets of X. For example, if $X = \{a, b, c\}$, then the power set is $\{\{\}, \{a\}, \{b\}, \{c\}, \{a, b\}, \{a, c\}, \{b, c\}, \{a, b, c\}\}$. It might help you to know that if X has $n \geq 0$ elements then the power set has 2^n elements. Write a `powerSet` function, that takes a set X as input and outputs its power set. (Use the language or pseudocode of your choice — as long as your algorithm is utterly clear and unambiguous. My answer is seven lines of Python.)