This is a closed-book, closed-notes, open-brain exam. If you brought anything with you besides writing instruments and your handwritten $8\frac{1}{2}'' \times 11''$ cheat sheet, please leave it at the front of the classroom.

Print your name, netid, and alias in the boxes above. Print your name at the top of every page (in case the staple falls out!).

You should answer all the questions on the exam.

The last few pages of this booklet are blank. Use that for a scratch paper. Please let us know if you need more paper.

If your cheat sheet if not hand written by yourself, or it is photocopied, please do not use it and leave it in front of the classroom.

Please submit your cheat sheet together with your exam. An exam without your cheat sheet attached to it will not be checked.

If you are NOT using a cheat sheet you should indicate it in large friendly letters on this page.

Questions containing the expression: “I don’t know”, will get 25% of the points of the question. If you write anything else, it would be ignored.

The total number of points given for “I don’t know” answers, will not exceed 10.

Write short and concise answers. Long and tedious answers will not be graded and will get grade zero automatically.

Time limit: 175 minutes.

Relax.
1. HW 0 Practice Problems  
[20 Points]

A. [10 Points] We toss a fair coin \( n \) times. What is the expected number of “runs”? Runs are consecutive tosses with the same result. For example, the toss sequence \( \text{HHHTTHHT} \) has 5 runs, and \( \text{HHTTTHTTHHTT} \) has 6 runs.

B. [10 Points] Solve the recurrence

\[
D(n) = \max_{n/3 < k < 2n/3} (D(k) + D(n - k) + n),
\]

and prove the correctness of your solution. (Your solution should be asymptotically tight, although you need to provide and prove only an upper bound.)
2. **With a little help from a friend.**

   [20 Points]

   **A. [10 Points]**
   Problem: SAT-E

   **Instance:** Two CNF formulas $F$ and $G$ defined over $n$ variables.

   **Question:** Are $F$ and $G$ not equivalent? Namely, is there an assignment for the $n$ variables such that $F$ evaluates to true, while $G$ evaluates to false, or vice versa.

   **Prove** that SAT-E is NP-Complete.

   **B. [10 Points]** Given a black box that can answer SAT-E in polynomial time, show how to find a satisfying assignment to a given 3SAT formula $F$, if such an assignment exists. The algorithm should work in polynomial time.
3. Cuts
[20 Points]

A. **Approx Max Cut**
[10 Points]
Given a graph $G = (V, E)$ with $n$ vertices and $m$ edges, describe an algorithm, as fast possible, that outputs a cut $S \subseteq V$, such that the expected number of edges in the cut is $\geq M/2$, where $M$ is the number of edges in the maximum cut, where the number of edges in the cut is $|(S \times (V \setminus S)) \cap E|$.
How fast is your algorithm?

B. **Minimum Cut**
[10 Points]
Present a deterministic algorithm, such that given an undirected graph $G$, it computes the minimum cut in $G$. How fast is your algorithm?
4. **Matchings**  
   **[20 Points]**

**A. [10 Points]** Let $M$ be a matching in the graph $G$, such that $M$ is not a maximum matching in $G$. Prove that there exists an augmenting path $\pi$ to $M$ in $G$, such that $M \oplus \pi$ is a matching, and $|M \oplus \pi| > |M|$.

**B. [10 Points]** Let $G = (V \cup U, E)$ be a bipartite graph, such that $|U| = |V| = n$. Prove that there exists a matching in $G$ of size $n$, if and only if, for every set $S \subseteq V$, we have $|N(S)| \geq |S|$, where $N(S) \subseteq U$ is the set of vertices in $U$ connected to a vertex in $S$. Formally, $N(S) = \{ u \mid vu \in E, v \in S \}$. 
5. **Linear Programming**

   [20 Points]

   **A. [10 Points]** Given a linear program $L$ with $m$ inequalities defined over $n$ variables, and let $\sum_{i=1}^{n} d_i x_i = e$ be an additional linear equality, called $X$. We are interested in finding a value $(x_1, \ldots, x_n)$ that is feasible both for $L$ and for $X$. Show how we can find such a value by solving a single linear program with $n - 1$ variables. (You can assume that one of $d_1, \ldots, d_n$ is non-zero.)

   **B. [10 Points]** Prove that if a linear program $L$ with $m$ inequalities, defined over $n$ variables, is infeasible, then there exists a set of $n + 1$ inequalities in $L$ which are infeasible by themselves (i.e., the linear program that only those $n + 1$ inequalities form is infeasible).