This is a list of (mostly true/false) questions to test your understanding of several important concepts and definitions that will often appear in this class. Most (but not all) of them should be straightforward assuming you have taken an undergraduate course in Discrete Math, Algorithms, and Automata theory. You are encouraged to go through your course notes or the relevant chapters provided in the references.

Feel free to discuss the questions on Perusall. We will go over the questions in our first class on Jan 9, and revisit some of these concepts in the first few lectures. This homework is optional and does not count towards your grade.

Problem 1 (Logic and Counting)**.** (References: [Sipser] Chapter 0.2)
Are the following statements True or False?

1. The set of integers between 1 to n can be represented using $\lceil \log n \rceil$ bits.
2. There is a 1-1 correspondence between the collection of subsets of $[n] := \{1, \ldots, n\}$ and the set of n-bit strings $\{0,1\}^n$.
3. There is a 1-1 correspondence between the set of functions of the form $f\colon [n] \to \{0,1\}$ and the set of n-bit strings.
4. There are 2^n many distinct functions of the form $f\colon \{0,1\}^n \to \{0,1\}$.
5. The complement of the set $\{\psi : \psi$ is a CNF formula that has a satisfying assignment$\}$ is $\{\psi : \psi$ is a CNF formula that has a non-satisfying assignment$\}$.
6. The set of real numbers are countable.
7. The set of finite binary strings is countable.
8. The set $\{0,1\}^*$, defined as $\cup_{i \in \mathbb{N}} \{0,1\}^i$, is the same as $\{0,1\}^{\mathbb{N}}$.

Problem 2 (Asymptotics and algorithms)**.** (References: [Sipser] Chapter 7.1, and Kleinberg and Tardos)

1. For each pair of the following functions $f(n)$ and $g(n)$, determine whether (i) $f(n) = O(g(n))$, (ii) $f(n) = \Omega(g(n))$; $f(n) = \Theta(g(n))$, or none of above.
   (a) $\log(n^{n/10})$ (b) $n!$ (c) $2^{\log n + \log\log n}$ (d) $2^{(\log n)^{1/2}}$ (e) $n/\log n$.
2. An algorithm A runs in time $3 \cdot 2^n$ on inputs of size $n \le 100$, and runs in time $3 \cdot n^2$ on inputs of size $n > 100$. What is the asymptotic time complexity of A?
3. Let $G = (V, E)$ be a directed graph with no multiple edges. Is it true that $O(\log|E|) = O(\log|V|)$?
4. Is it true that the Depth-First Search (DFS) algorithm runs in linear time in the number of vertices of the input graph?

Problem 3 (Turing Machines)**.** (References: [Sipser] Chapter 4) Are the following statements True or False?

1. The size of a Turing machine depends on the input length.
2. There are finitely many Turing machines.

3. The set of Turing machines is countable.

4. If a Turing machine runs on an input and does not accept, then it must reject.

5. The language $L := \{\langle M, x, t\rangle : M \text{ is a Turing machine that halts on input } x \text{ in } t \text{ steps}\}$ is decidable.

Problem 4 (Complexity classes)**.** (References: [Sipser] Chapter 7)

Which of following problems are known to be in the classes of P, NP, NP-hard, NP-complete, or neither?

1. INPUT: an undirected graph $G = (V, E)$ with n vertices
   OUTPUT: the size of the maximum clique in G

2. INPUT: A unary input 1^n
   OUTPUT: 1 if n is prime and 0 otherwise

3. INPUT: an undirected graph $G = (V, E)$ with n vertices
   OUTPUT: 1 if G contains k vertices that are not adjacent to each other and 0 otherwise

4. INPUT: A directed graph $G = (V, E)$ with n vertices and two vertices s and t
   OUTPUT: 1 if s and t are connected by a path and 0 otherwise

5. INPUT: a CNF formula ψ on n variables and an assignment $x \in \{0, 1\}^n$
   OUTPUT: 1 if x satisfies ψ and 0 otherwise

6. INPUT: a CNF formula ϕ on n variables
   OUTPUT: 1 if ψ has a satisfying assignment and 0 otherwise

Problem 5 (Reductions)**.** (References: [Sipser] Chapter 7.4)

Let L_1, L_2 be two languages. Are the following statements True or False?

1. L_1 is polynomial-time reducible to L_2 if there is a polynomial-time algorithm ϕ such that $x \in L_1$ if and only if $\phi(x) \in L_2$.

2. Suppose L_1 is NP-complete. If L_1 is polynomial-time reducible to L_2 and L_2 is in P, then $\mathsf{P} = \mathsf{NP}$.

3. Checking whether an undirected graph is connected is polynomial-time reducible to checking whether a CNF formula has a satisfying assignment.

4. Deciding whether a CNF formula has a satisfying assignment requires exponential time.