Problem 1 (5 points):
Prove that, for any constants $c$ and $d$, both greater than 1, $n^d = o(c^n)$, i.e., exponential grows faster than polynomial.

Problem 2 (9 points):
For each of the three lists of four functions, give an ordering $f_1, f_2, f_3, f_4$ such that $f_1(n) = O(f_2(n)) = O(f_3(n)) = O(f_4(n))$. You need not provide proofs, just correct orderings.

In some cases, stronger statements can be made, i.e., there are pairs $f_i, f_{i+1}$ in one or more of the correct orderings such that $f_i(n) = o(f_{i+1}(n))$. You may earn one point of extra credit for each such pair that you identify only if you provide a proof that this little-oh relationship holds.

1. $\log n$, $\sqrt{n}$, $(\log n)^2$, $\log \log n$.
2. $n^{4/3}$, $n \log n$, $n^2$, $\log(n!)$.
3. $n!$, $n^n$, $e^n$, $2^{\log n \log \log n}$.

Problem 3-a (9 points):
Specify a Turing Machine that, on input $\langle p \rangle$, the binary representation of a nonnegative integer $p$, outputs $\langle p + 1 \rangle$. You may use any number of states, any alphabet, and any number of work tapes.

Example input tape:
\[
\ldots 1001011111 \ldots
\]

Corresponding output tape:
\[
\ldots 1001100000 \ldots
\]

Your solution should have running time $O(n)$, where $n$ is the length of the input.

Problem 3-b (7 points):
Provide an oblivious Turing Machine that computes the same function as the machine that you provided in part (a). Your solution should have running time $O(n \log n)$.

Problem 4 (14 points):
Prove that 2SAT $\in$ P.

Problem 5 (10 points):
A “system of quadratic equations modulo 2 in $n$ variables” is a set of equations over $\mathbb{Z}_2$ in which each term has degree at most 2. (For example, $x_1 \cdot x_2 + x_3 \equiv 1 \pmod{2}$ is a quadratic equation modulo 2.) Such a system is said to be “solvable” if there is an assignment of values in $\mathbb{Z}_2$ to the variables $x_1, \ldots, x_n$ that satisfies all of the equations.
Prove that the language of solvable systems of quadratic equations modulo 2 is NP-complete.

**Problem 6 (13 points):**
A graph $G(V, E)$ is $k$-colorable if there is an assignment $f$ of $k$ colors to the vertices (i.e., $f : V \rightarrow \{1, 2, \ldots, k\}$), such that, if $(a, b) \in E$, then $f(a) \neq f(b)$. That is, the endpoints of each edge are assigned different colors. Let $3\text{Colorable}$ denote the set of all graphs that are 3-colorable.

Provide a many-to-one, polynomial-time reduction from $3\text{Colorable}$ to $\text{SAT}$. Prove that the reduction is correct and runs in polynomial time.

**Problem 7 (10 points):**
Prove that, if every unary NP language is in P, then $\text{EXP} = \text{NEXP}$.

**Problem 8 (13 points):**
Let $\Delta = \text{NP} \cap \text{coNP}$. Prove that $\Delta$ equals the class of decision problems that are polynomial-time Cook-reducible to $\Delta$, i.e., $\Delta = P^\Delta$.

**Problem 9 (10 points):**
Let $x_1 \neq x_2$ be distinct elements of $\{0, 1\}^\ast$. Then $E_{x_1, x_2}$ is defined as the set of all Turing Machines $M$ such that $M$ does not distinguish between the input $x_1$ and the input $x_2$. (More precisely, the elements of $E_{x_1, x_2}$ are encodings of Turing Machines as binary strings, but for brevity we will refer to them as machines.) That is, $M$ is in $E_{x_1, x_2}$ if $M$ either halts on both inputs $x_1$ and $x_2$ or halts on neither, and, if it halts on both, $M(x_1) = M(x_2)$.

Prove that membership in the set $E_{x_1, x_2}$ is not decidable.