1. **(SOLO)** Read very carefully the course information page:

   [http://www.cs.cmu.edu/~15251/course-info.html](http://www.cs.cmu.edu/~15251/course-info.html)

   This page contains links to 2 important documents:
   

   Read these carefully as well. All of this reading is a required component of the course. We will ask you questions on these to make sure that you have indeed read and understood them.

2. **(SOLO)** Read Chapters 1 and 2 of the course notes.

   There are various definitions in Chapter 1 that we did not mention in class. Make sure you know all of them.

   Chapter 2 is on countable and uncountable sets. This semester, instead of lecturing on this topic that you have already seen in Concepts, we are asking you to read the corresponding chapter in order to remind yourself of the important concepts.

   If you have any questions about anything, let us know.

3. **(SOLO)** Below you are given a list of claims and corresponding “proofs”. For each one, do the following.

   (i) Identify whether the proof is correct. Don’t just explain why the claim is wrong; rather, explain how the argument violates the notion of a correct proof. In particular, you have to point out specifically which step of the argument is wrong. If you think the proof is correct, you do not need to elaborate.

   (ii) Go through the proof checklist at


   and tick the boxes that you think the proof satisfies. For each box you do not tick, explain very briefly why the proof violates the corresponding point.

Here is the list.

(a) **Claim:** Every natural number is either prime or a perfect square.

   **Proof:** As the inductive hypothesis, use $P_n =$ “every number less than or equal to $n$ is a prime or a perfect square.”

   $P_1$ is certainly true, since 1 is a perfect square. Now consider $n$. If $n$ is prime we are done. Otherwise, $n$ can be factored as $n = rs$ with $r$ and $s$ less than or equal to $n - 1$. By the inductive hypothesis, $r$ and $s$ are perfect squares, $r = u^2$ and $s = v^2$. Then $n = rs = u^2v^2 = (uv)^2$.

(b) We’ll show that 251 is a magic number.

   **Claim:** For all positive integers $n$, we have $251^{n-1} = 1$.

   **Proof:** If $n = 1$, $251^{n-1} = 251^{1-1} = 251^0 = 1$. And by induction, assuming that the theorem is true for $1, 2, \ldots, n$, we have

   $$251^{(n+1)-1} = 251^n = \frac{251^{n-1} \times 251^{n-1}}{251^{n-2}} = \frac{1 \times 1}{1} = 1;$$
so the theorem is true for \( n + 1 \) as well.

(c) **Claim:** For all negative integers \( n \), \(-1 - 3 - \cdots + (2n + 1) = -n^2\).

**Proof:** The proof will be by induction on \( n \).

**Base Case:** \(-1 = -(-1)^2\). It is true for \( n = -1 \).

**Inductive Hypothesis:** Assume that \(-1 - 3 - \cdots + (2n + 1) = -n^2\).

**Inductive Step:** We need to prove that the statement is also true for \( n - 1 \) if it is true for \( n \), that is, \(-1 - 3 - \cdots + (2(n - 1) + 1) = -(n - 1)^2\). Starting from the left hand side,

\[
-1 - 3 - \cdots + (2(n - 1) + 1) = (-1 - 3 - \cdots (2n + 1)) + (2(n - 1) + 1)
\]
\[
= -n^2 + (2(n - 1) + 1) \quad (\text{Inductive Hypothesis})
\]
\[
= -n^2 + 2n - 1
\]
\[
= -(n - 1)^2.
\]

Therefore, the statement is true.

(d) **Claim:** Every positive integer \( n \geq 2 \) has a unique prime factorization.

**Proof:** We will prove by strong induction on \( n \).

**Base Case:** \( 2 \) is a prime itself. It is true for \( n = 2 \).

**Inductive Hypothesis:** Assume that the statement is true for all \( 2 \leq k \leq n \).

**Inductive Step:** We must prove that the statement is true for \( n + 1 \). If \( n + 1 \) is prime, then it itself is a unique prime factorization. Otherwise, \( n + 1 \) can be written as \( x \times y \) where \( 2 \leq x, y \leq n \). From the inductive hypothesis, both \( x \) and \( y \) have unique prime factorizations. The product of unique prime factorizations is unique, therefore, \( n + 1 \) has a unique prime factorization.

4. (SOLO) This question is about the idea that many computational problems have a natural decision version such that the computational problem is efficiently solvable if and only if the corresponding decision problem is efficiently solvable. This is why in theoretical computer science, we often restrict our attention to decision problems, i.e. languages.

Consider the following problem. The input is a natural number \( N \). The output is the prime factorization of \( N \) (which can be represented as a tuple of prime numbers). Consider also the following decision version of this problem: given as input natural numbers \( N \) and \( K \), output True if \( N \) has a factor between 2 and \( K \) (inclusive), and output False otherwise.

As seen in lecture and the notes, we encode the input and output to get the corresponding computational problem and the decision problem. So we will assume that the numbers are given to us as binary strings (corresponding to the usual binary representation of natural numbers).

Hopefully it is clear that if someone gives you a function \( g \) that solves the computational problem, then you can come up with an algorithm that solves the decision problem: just call \( g \) to get the prime factorization of \( N \), and compare \( K \) to the smallest prime factor of \( N \).

Now suppose someone gives you function \( f \) that solves the decision problem. Give an algorithm that solves the computational problem. Give an algorithm that solves the computational problem. Your algorithm should be efficient in the following sense: it can call \( f O(\log^k N) \) times, and do additional work with running time \( O(\log^k N) \), for some constant \( k \) of your choosing (for instance \( k \) could be 2 or 3). You can assume that basic arithmetic operations like plus, minus, multiplication and division between numbers of value at most \( N \) take at most \( O(\log^2 N) \) time.
Make sure you have a clear description of how your algorithm works. A formal proof of correctness is not required.

5. (SOLO) Let $L \subseteq \{0,1\}^*$ be the set of all strings $s$ with the property that $s = s^R$.
   
   (a) Give a recursive definition for $L$.
   
   (b) Prove that your definition is correct. That is, if $K$ is the set of all words that can be constructed inductively by applying your recursive rules from part (a), then prove that $L = K$.

6. (SOLO) Fix an alphabet $\Sigma$. We define the notion of an awesome language recursively as follows.
   
   • $\emptyset$ is awesome;
   
   • $\{a\}$ for each $a \in \Sigma$ is awesome;
   
   • if $L_1$ and $L_2$ are awesome, then $L_1 \cup L_2$ is awesome;
   
   • if $L_1$ and $L_2$ are awesome, then $L_1L_2$ is awesome;

   So an awesome language is any language that can be constructed by starting from the base cases and applying the recursive rules a finite number of times (hopefully it is clear which ones above correspond to the base cases and which ones correspond to the recursive rules).

   We define the notion of a super awesome language recursively as follows.

   • $\emptyset$ is super awesome;
   
   • $\{a\}$ for each $a \in \Sigma$ is super awesome;
   
   • if $L_1$ and $L_2$ are super awesome, then $L_1 \cup L_2$ is super awesome;
   
   • if $L_1$ and $L_2$ are super awesome, then $L_1L_2$ is super awesome;
   
   • if $L$ is super awesome, then $L^*$ is super awesome.

   (a) Give a simple characterization for the set of all awesome languages. Briefly justify your answer. A detailed argument is not needed.

   (b) Is the set of all awesome languages the same as the set of all super awesome languages? Briefly justify your answer using part (a).

   (c) For a function $f : \Sigma \rightarrow \Sigma^*$ and a language $L \subseteq \Sigma^*$, define the language

   $$L^f = \{f(a_1)f(a_2)\cdots f(a_k) : \text{ for all } i, a_i \in \Sigma, \text{ and } a_1a_2\ldots a_k \in L\}.$$ 

   Prove that for any $f : \Sigma \rightarrow \Sigma^*$, if $L$ is super awesome, then so is $L^f$.

7. (SOLO) Determine (with proof) whether the following sets are countable or uncountable.

   (a) The set of all pretty cool languages over $\Sigma = \{0,1\}$, where a language is pretty cool if every word in the language contains an equal number of 0’s and 1’s.

   (b) The set of all finite languages over the alphabet $\Sigma = \{0,1,\ldots,9\}$.

8. (OPEN) Let $\Sigma$ be an alphabet. Let $x$ and $y$ be nonempty strings over $\Sigma$. Consider the following three statements about $x$ and $y$:
(i) \( xy = yx \);
(ii) there is a nonempty string \( z \) and numbers \( m, n \in \mathbb{N}^+ \) such that \( x = z^m \) and \( y = z^n \);
(iii) there are numbers \( k, \ell \in \mathbb{N}^+ \) such that \( x^k = y^\ell \).

Show that statements (i), (ii), (iii) are equivalent; if one of them holds, so do the other two.

9. **(SOLO - BONUS)** Solve the Infected Chessboard Problem seen in class. What is the smallest number of initially infected squares required to infect an \( n \times n \) board? Does it depend on \( n \)? If so, how?

Note: Solutions to bonus problems should be sent to the instructors by email. Bonus problems are not worth any points. They are for your spiritual growth!