Name: Harshit Jain
User ID: hmj5262

Problem 1

Proof by reduction: SAT $\leq_p$ DOUBLE-SAT

We will define the function $f(\phi) = \psi$ such that: $\psi = \phi \wedge (x \vee \bar{x})$ where x is a variable that does not appear in ϕ.

If n is the length of ϕ, it will take $O(n)$ time to find a variable name that is not in ϕ and constant time to append "$\wedge(x \vee \bar{x})$" to ϕ. Thus this reduction can be computed in polynomial time.

If $\phi \in$ SAT, then $\psi \in$ DOUBLE-SAT. If $\phi \in$ SAT, then we know that the left side of ψ is satisfiable. We can then set our new variable x to True, which will satisfy ψ. Alternatively, we can then set our new variable x to False, which will satisfy ψ. Thus there are at least 2 different satisfying assignments and $\psi \in$ DOUBLE-SAT.

If $\phi \notin$ SAT, then $\psi \notin$ DOUBLE-SAT. If $\phi \notin$ SAT, then there is no way to satisfy the left side of ψ. Because of the $\wedge$ operator, this leaves us with no way to satisfy ψ overall. If ψ cannot be satisfied then it certainly cannot have 2 satisfying assignments and $\psi \notin$ DOUBLE-SAT.

Thus, DOUBLE-SAT is in NP-HARD. Since it is in NP as well, **DOUBLE-SAT is NP-Complete**.

Problem 2

First, we show that the set-partition problem belongs to NP. Given the set S, our certificate is a set A which is a solution to the problem. The verification algorithm checks that $A \subseteq S$ and that $\sum_{x \in A} x = \sum_{x \in S \setminus A} x$. Clearly, this can be done in polynomial time.

To show that the problem is NP-complete, we reduce from SUBSET-SUM. Let (S,t) be an instance of SUBSET-SUM. The problem is to determine whether there is a subset $A \subseteq S$ such that $t = \sum_{x \in A} x$. We construct an instance S_0 of the set-partition problem by setting $S_0 = S \cup \{r\}$ where $r = 2t - \sum_{x \in S} x$. Clearly, this reduction can be done in polynomial time.

Now it remains to show that there is a subset of S whose sum is t if and only if S_0 can be partitioned into two distinct subsets of equal weight. First, suppose that $A \subseteq S$ and the sum of elements in A is t. But now we have $\sum_{x \in S_0 \setminus A} x = \sum_{x \in S \setminus A} x + r = 2t - \sum_{x \in A} x = 2t - t = t = \sum_{x \in A} x$. Thus, the partition into A and $S_0 \setminus A$ is a solution to the set-partitioning problem.

Now, suppose that there exists $A \in S_0$ such that $\sum_{x \in A} x = \sum_{x \in S \setminus A} x$. Without loss of generality, assume that $r \notin A$. But now we have $2t = \sum_{x \in S} x + r = \sum_{x \in S_0} x = \sum_{x \in A} x + \sum_{x \in S \setminus A} x = 2 \sum_{x \in A} x$. Thus, $\sum_{x \in A} x = t$ and A is a solution to the subset-sum problem.

Problem 3

Algorithm to Find the Largest Clique Using k-CLIQUE Black Box:

1. **Initialization**: Let n be the number of vertices in the graph G. Set $lower = 0$ and $upper = n$ which are the bounds for the clique size search.
2. **Binary Search for Maximum Clique Size**: Perform a binary search between *lower* and *upper* to find the largest value k for which k-CLIQUE returns yes.
3. While $lower < upper$:
   - Set $mid = \lceil (lower + upper + 1)/2 \rceil$ (using the upper middle to avoid infinite loop).
   - If k-CLIQUE(G, mid) returns yes, set $lower = mid$ (since at least a clique of size *mid* exists).
   - If k-CLIQUE(G, mid) returns no, set $upper = mid - 1$ (since no clique of size *mid* exists and we lower the search space by one to avoid infinite loop in the case where upper-middle is used).
4. After the loop, *lower* will be equal to the largest clique size.
5. **Identifying the Vertices**:
   - Start with the subgraph G' of G that includes all n vertices.
   - For each vertex v in G', test if the graph $G' - \{v\}$ (which is G' without the vertex v and its incident edges) contains a clique of size *lower* by calling k-CLIQUE($G' - \{v\}, lower$).
   - If it does, remove the vertex v from G', since v is not part of the maximum clique of size *lower*.
   - Repeat this process until all vertices in G' are part of the maximum clique.
6. **Output**: The remaining graph G' consists of the vertices in the largest clique.

Correctness: The binary search ensures we find the largest k for which a k-clique exists in the graph. Since the black box runs in polynomial time and binary search takes $O(\log n)$ iterations with each iteration taking polynomial time, the maximum clique size is found in polynomial time. The second part iteratively removes vertices that are not part of the maximum clique, taking $O(n)$ uses of the k-CLIQUE black box, which each takes polynomial time. Therefore, the whole process runs in polynomial time.

Polynomial Time: The overall algorithm is polynomial because:

- The binary search for the maximum clique size has $O(\log n)$ iterations.
- Each call to k-CLIQUE is polynomial.

- Identifying the vertices in the clique involves n calls to k-CLIQUE and a constant amount of work for each vertex.

Therefore, the algorithm calls the k-CLIQUE black box $O(n \log n)$ times, and since each call is polynomial, the overall algorithm is polynomial.

Problem 4

To show that the set of weakly decreasing infinite sequences of natural numbers is countable, we can use a correspondence between these sequences and partitions of integers, which are known to be countable.

- Step 1: Consider each weakly decreasing sequence as a representation of the multiset coefficients of an infinite polynomial series with nonnegative integer coefficients of the form $(1+x+x^2+x^3+\dots)^n$, where n is a nonnegative integer.
- Step 2: The first element of the sequence, a_1, can be any natural number, including zero. We map each sequence to a polynomial where powers of x up to x^{a_1} have a coefficient of 1, and the others are 0.
- Step 3: Given that weakly decreasing sequences will eventually become a sequence of zeros, this is equivalent to the sequence having a finite number of nonzero terms followed by zeros indefinitely, which confirms the mapping to polynomials is valid.
- Step 4: This constructs a one-to-one correspondence between each weakly decreasing sequence and a unique partition of a natural number, given by the sum of exponents with nonzero coefficients in the associated polynomial.
- Step 5: As the set of partitions of any natural number is countable and we have established a one-to-one correspondence with weakly decreasing sequences, it follows that the set of these sequences is also countable.

Hence, by demonstrating a bijective relation to the countable set of partitions of natural numbers, we conclude that the set of weakly decreasing infinite sequences of natural numbers is countable.