Theory of Computation, Fall 2023
Assignment 9 (Due December 27 Wednesday 10:00 am)

Only part I will be graded.

Part I

Q1. Let $f : \mathcal{N} \to \mathcal{N}$ be a primitive recursive function. Define $F : \mathcal{N} \to \mathcal{N}$ to be

$$F(n) = f(f(\ldots f(n)\ldots))$$

where there are n compositions. For example, $F(0) = f(0)$ and $F(1) = f(f(1))$. Show that F is primitive recursive.

Q2. Show that for any $k \geq 2$, the following function is primitive recursive.

$$\varphi_k(n_1, \ldots, n_k) = max\{n_1, \ldots, n_k\}$$

for any $n_1, \ldots, n_k \in \mathcal{N}$.

Part II

Q3. Prove that if A is in $\mathcal{P}$, so is $\overline{A}$.

Q4. Define co-$\mathcal{NP}$ to be the following set of languages.

$$\text{co-}\mathcal{NP} = \{A : \overline{A} \in \mathcal{NP}\}$$

Prove that $\mathcal{P} \subseteq \mathcal{NP} \cap \text{co-}\mathcal{NP}$.

Q5. Construct a polynomial-time verifier for the following language.

$$L = \{G : G \text{ is a graph that contains a Hamiltonian cycle}\}$$

We say a cycle is a Hamiltonian cycle in G if it visits every vertex of G exactly once.