

Theory of Computation, Fall 2021

Exercises for Grammars and Numerical Functions

Exercises

Q1 [1, Problem 4.6.2(a)] Construct a grammar that generates $\{ww : w \in \{a, b\}^*\}$.

Q2 [1, Problem 4.7.1] Let $f : \mathcal{N} \rightarrow \mathcal{N}$ be a primitive recursive function. Define $F : \mathcal{N} \rightarrow \mathcal{N}$ to be

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

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

Q3 [1, Problem 4.7.2(a)] Show that $factorial(n) = n!$ is primitive recursive.

Q4 Show that for any $k \geq 2$, the following function is primitive recursive.

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

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

References

- [1] Lewis H., Papadimitriou C.. Elements of the Theory of Computation. Prentice-Hall (1998)