Homework 3
CS 5315 (Theory of Computation)
Instructor: Dr. Vladik Kreinovich

Prove the following theorem:

THERE EXISTS A COMPUTABLE FUNCTION THAT IS NOT PRIMITIVE RECURSIVE (PR).

It is necessary to find a computable function and prove that it is not PR.

Part 1 Describe a “code” of a PR function \( f \) as the unique integer that identifies that PR function \( f \). This code can be obtained by the following steps:

a. Translate the PR function into an expression in terms of 0, \( \pi^k_i \), \( \sigma \), \( \circ \) and PR,

b. Write this expression in some language that allows the \( \pi^k_i \), \( \sigma \), \( \circ \) symbols. For example, \( \LaTeX \). This will transform the function expression into its ASCII symbols representation,

c. Transform the ASCII symbols into their binary representation,

d. Interpret the sequence of 0’s and 1’s as an integer \( c \) in the binary form, this integer is the code of the PR function.

Part 2 For every integer \( c \) that is a code of a PR function, denote the corresponding PR function by \( f_c(n) \). This function \( f_c \) takes an integer \( n \) as input and returns \( f_c(n) \). Given \( c \) and \( n \), we can compute \( f_c(n) \) as follows:

a. The integer \( c \) is a binary sequence (of 0’s and 1’s),

b. Translate this binary sequence into the ASCII symbols representation,

c. Transform ASCII symbols using \( \LaTeX \) to a series of correct symbols that can be given as input to a PR compiler,

d. Parse tree of the compiler reconstruct the program of \( f_c \) which can be run on a computer.
PROOF

Part 1 Define function $f$ formally.

$$f(n) = \begin{cases} f_n(n) + 1 & \text{if a PR compiler verifies that } n \\
0 & \text{otherwise} \end{cases}$$

Since $f_n$ is computable, $f$ is also clearly computable.

Part 2 Prove that function $f$ is not PR.

Assume that function $f$ is PR. Therefore, there exists a code $c$ such that for all $n$, $f(n)$ coincides with $f_c(n)$, or $\exists c \forall n [f(n) = f_c(n)]$. Then, $f(c) = f_c(c)$, when $n = c$.

By definition (1) of function $f$, $f(c) = f_c(c) + 1$.

Since $f_c(c) = f(c) = f_c(c) + 1$ and $f_c(c) \neq f_c(c) + 1$, we reached a contradiction!

This means that $f(n)$ can not be PR.

Therefore, there exists a function $f$ that is not PR (primitive recursive). □