Higher-Order Computability
1. Exercise Sheet

TECHNISCHE UNIVERSITÄT DARMSTADT

Department of Mathematics
Dr. Thomas Powell

Summer Semester 2019
30 April

Homework

Key to exercises: (P) = programming component (+) = more difficult or open ended.

Exercise H1
Define the usual addition function $+ : \mathbb{N} \times \mathbb{N} \to \mathbb{N}$ as a closed term $t : N \to N \to N$ of System T. Do the same for the multiplication and exponential functions.

Exercise H2
Prove that all primitive recursive functions $f : \mathbb{N}^k \to \mathbb{N}$ are definable as a closed term $t : \underbrace{N \to \ldots \to N}_{k \text{ times}} \to N$ of System T.

Recall the definition of the Ackermann function $A : \mathbb{N} \times \mathbb{N} \to \mathbb{N}$:

$$A(m,n) = \begin{cases} n+1 & \text{if } m = 0 \\ A(m-1,1) & \text{if } m > 0 \text{ and } n = 0 \\ A(m-1, A(m,n-1)) & \text{if } m > 0 \text{ and } n > 0 \end{cases}$$

Exercise H3
Give closed expressions for $A(1,n)$ and $A(2,n)$. What about $A(3,n)$ and $A(4,n)$?

Exercise H4 (P)
Write a program in your favourite language which implements the Ackermann function. What is the smallest input that breaks your computer?

Exercise H5
Prove the following for all $m,n \in \mathbb{N}$:

(a) $A(m,n) > n$.

(b) $A(m,n+1) > A(m,n)$.

(c) $A(m+1,n) > A(m,n)$.

(d) $A(m+1,n) \geq A(m,n+1)$.

Exercise H6
Show that A is definable in System T as a closed term t of type $N \to N \to N$.

Exercise H7 (+)
Let $\succ$ be the ordering on $\mathbb{N} \times \mathbb{N}$ define by $(m,n) \succ (m',n')$ if $m = m'+1$ or $m = m'$ and $n = n'+1$. Consider the following scheme of recursion over $\succ$:

$$f(m,n) = h(m,n,\lambda m',n'.f(m',n') \text{ if } (m,n) \succ (m',n') \text{ else } 0).$$

Show that the Ackermann function can be defined using recursion of this kind. Can recursion over $\succ$ be simulated in System T?

Exercise H8 (P+)
Design your own programming language which comprises the terms of System T and write a compiler for it.