(Extra) Exercises Week 3
Recursion Theory

Joost J. Joosten

September 21, 2006

1. In Example 2.1.4 from the book, write out the recursive scheme for times in a fully formal way as was done on the last two lines of Page 13. Of course, you are allowed to use plus as an already defined function.

2. (a) Define $<\subseteq \mathbb{N}^2 \times \mathbb{N}^2$ as follows:

$$(m_1, n_1) < (m_2, n_2) \text{ if } n_1 < n_2 \text{ or } (n_1 = n_2 \text{ and } m_1 < m_2).$$

   Prove that for any non-empty subset P of $\mathbb{N}^2$, there is a $<$-least element in P. (Hint: Suppose this fails and derive a contradiction.)

   (b) For any statement ϕ, prove the following:

$$(\forall m, n \in \mathbb{N}\ (\forall (m', n') < (m, n)\ \phi(m', n')) \implies \phi(m, n))$$
$$\implies \forall m, n \in \mathbb{N}\ \phi(m, n)$$

   (Hint: Suppose this fails and use the previous exercise to derive a contradiction.)

   (c) Prove that Ackermann function is total. (Hint: Use the previous exercise.)

3. Prove that for each n, we have that $\alpha_n \in \mathsf{PRIM}$, where $\alpha_n(x) := A(x, n)$ for any $x \in \mathbb{N}$.

4. (a) Let k be a natural number with $k \geq 1$. For any k-ary functions f and g, $f \leq g$ if $f(\vec{x}) \leq g(\vec{x})$ for any $\vec{x}$. For a k-ary function f, f is increasing if $f(x_1, \cdots, x_k) \leq f(x'_1, \cdots, x'_k)$ holds if $x_1 \leq x'_1, \cdots, x_k \leq x'_k$. Prove that for any k-ary primitive recursive function f, there is a k-ary primitive recursive function g such that $f \leq g$ and g is increasing. (Hint: Use bounded sum.)

   (b) Let f and f' be l-ary functions obtained by the composition $h(g_1, \cdots, g_k)$ and $h'(g'_1, \cdots, g'_k)$ respectively. Prove that if $h \leq h'$, $g_1 \leq g'_1, \cdots, g_k \leq g'_k$ hold and h', $g'_1, \cdots, g'_k$ are increasing, then $f \leq f'$.

(c) Let f, f' be $(k+1)$-ary functions, g, g' be k-ary functions, h, h' be $(k+2)$-ary functions and assume that f, f' are obtained by primitive recursion scheme from g, h and g', h' respectively. Prove that if $g \leq g'$, $h \leq h'$ hold and g', h' are increasing, then $f \leq f'$.

(d) For any natural numbers n, m with $n \geq 1$, prove the following:

i. $\alpha_n(m) \geq m + 2$.

ii. $\alpha_n(m+1) > \alpha_n(m)$.

iii. $\alpha_{n+1}(m) \geq \alpha_n(m+1)$.

iv. $\alpha_{n+1}(m) > \alpha_n(m)$.

v. $\alpha_{n+1}(m) \geq \alpha_n(2m)$.

(e) For a k-ary function f and a 1-ary function g, f is bounded by g if $f(m, \cdots, m) \leq g(m)$ holds for any natural number m. Prove that every primitive recursive function is bounded by some α_n for some n.

(f) Prove that the Ackermann function, as formulated by Rósza Péter, is not a primitive recursive function.

5. Prove (without referring to Church's Thesis) that the Ackermann function, as formulated by Rósza Péter, is a recursive function. (Hint: describe a (coded) sequence of sequences that describes the calculation of the Ackermann function. Use minimalisation to find the minimal such sequence and extract the necessary value from the sequence.)

6. Let $g(\vec{x}), h(\vec{x}, y) \in \mathsf{PRIM}$. Assume that for any $\vec{x}$ there is a $y \leq g(\vec{x})$ such that $h(\vec{x}, y) = 0$. Prove that the function $f(\vec{x})$, which is defined by *bounded minimalisation* from g and h in the following way

$$f(\vec{x}) := \mu y \leq g(\vec{x})[h(\vec{x}, y) = 0]$$

is also a primitive recursive function. Here, the $\mu y \leq g(\vec{x})$ should be read as "the minimal y less than or equal to $g(\vec{x})$ such that".

7. (a) Prove that the relation Pr ($\mathrm{Pr}(x)$ if x is a prime) is primitive recursive.

(b) Prove that the Prime function $n \mapsto p_n$ (n-th prime (e.g. $p_0 = 2$)) is primitive recursive. (Hint: Use the fact that there is a prime number between n and $2n$ for arbitrary $n \geq 1$.)

8. Consider the following pairing function P.

$$P(x, y) = \frac{1}{2}[(x+y)^2 + 3x + y]$$

(a) Draw in a picture in what order the pairs are enumerated by this function.

(b) Prove that $\Sigma_{i=1}^{n} i = \frac{n(n+1)}{2}$.

(c) Describe your answer from 8a in some free-style algorithmic way. Prove that $P(x, y)$ is the formula that enumerates all pairs according to this algorithm. (You might find Item 8b useful here.)