MATH 570: MATHEMATICAL LOGIC

HOMEWORK 8

Due date: Oct 27 (Tue)

Below, problems with $*$ are not mandatory.

1. Prove that Tarski's theorem on $\mathsf{Th}(\mathbf{N})$ not being arithmetical (Theorem 6.5) is equivalent to the fixed point lemma for $\mathsf{Th}(\mathbf{N})$ (Lemma 6.4). Don't just say "well, both are true and hence equivalent"; instead, using one as a black box, deduce the other, and vice versa.

2. Prove Lemma 6.8(e) as well as Lemma 6.14(a,b,c).

3. Put $\mathbb{N}^{<\mathbb{N}} := \bigcup_{l\in\mathbb{N}} \mathbb{N}^l$ and let $\langle\rangle : \mathbb{N}^{<\mathbb{N}} \to \mathbb{N}$ be a primitive recursive coding of tuples such that all of the decoding functions are also primitive recursive; for example,

$$\langle n_1, ..., n_l\rangle = p_1^{n_1+1}...p_l^{n_l+1},$$

where $l \in \mathbb{N}$ and p_i denotes the i^{th} prime number. For $f : \mathbb{N}^{k+1} \to \mathbb{N}$, define

$$\bar{f}(\vec{a}, n) = \langle f(\vec{a}, 0), f(\vec{a}, 1), ..., f(\vec{a}, n-1)\rangle.$$

Given $h : \mathbb{N}^{k+1} \to \mathbb{N}$, let $f : \mathbb{N}^{k+1} \to \mathbb{N}$ be defined by the identity

$$f(\vec{a}, n) = h(\vec{a}, \bar{f}(\vec{a}, n)).$$

Show that if h is primitive recursive, then so is f.

4. Let $g : \mathbb{N} \to \mathbb{N}$, $h : \mathbb{N}^3 \to \mathbb{N}$, $\tau : \mathbb{N}^2 \to \mathbb{N}$. We say that $f : \mathbb{N}^2 \to \mathbb{N}$ is defined by *nested recursion* from g, h, τ if for each $x, y \in \mathbb{N}$,

$$\begin{cases} f(0, y) &= g(y) \\ f(x+1, y) &= h(x, y, f(x, \tau(x, y))) \end{cases}.$$

Show that if g, h, τ are primitive recursive, then so is f.

5. Show that the the Ackermann function is recursive.

HINT: To prove that primitive recursion operation is a recursive operation, we used the Dedekind analysis of primitive recursion, namely, we *searched* for a tuple encoding all of the previous values of the function. In this tuple, the i^{th} term was equal to the value of the function at i; thus, we could equivalently search for a tuple of pairs (i, w), where w would be the value of the function at i. Now for the Ackermann function, a similar analysis can be done, where we again *search* for a tuple of triples (i, j, w), where we put appropriate conditions to ensure that $w = A(i, j)$. In other words, this tuple we are looking for is a **partial** matrix of values of A.

6. Below are some properties of the Ackermann function that will be used in the next homework in showing that the Ackermann function is not primitive recursive. Choose any two of these properties and prove them.

(a) $A(n, x+y) \geq A(n, x) + y$;

(b) $n \geq 1 \implies A(n+1, y) > A(n, y) + y$;

(c) $A(n+1, y) \geq A(n, y+1)$;

(d) $2A(n, y) < A(n+2, y)$;

(e) $x < y \implies A(n, x+y) \leq A(n+2, y)$.

7.* Show that the graph of the Ackermann function is primitive recursive. This implies, once again, that the Ackermann function itself is recursive.

HINT: Use a similar argument to the proof that the Ackermann function is recursive, but bound your search using the value of the function.