Subject 24.242. Logic II. Homework due Wednesday, April 3.

For problems 1-4, a *register machine* consists of an infinite number of memory locations, named Register 1, Register 2, Register 3, and so on, each of which is capable of holding a natural number. A *register program* is a finite numbered list of instructions, which take the following five forms:

Add 1 to the number in Register i.
Subtract 1 from the number in Register j, unless that number is already 0.
If the number in Register k is 0, go to instruction m
Go to instruction n.
STOP.

A computation starts at the first instruction, and proceeds from an instruction to the next, unless instructed otherwise. To calculate an n-ary function, begin with the inputs in Registers 1 through n. If the computation eventually reaches the STOP instruction, the computation halts, and the number in Register 1 is the output. If the computation never reaches the STOP instruction, the function is undefined for that input. For example, the following program computes the successor function:

1. Add 1 to Register 1.
2. Stop.

The following program computes the characteristic function of the identity relation, the binary function that yields output 1 if x = y and 0 if x ≠ y:

1. If the number in Register 1 is 0, go to instruction 6.
2. If the number in Register 2 is 0, go to instruction 10.
3. Subtract 1 from the number in Register 1, unless that number is already 0.
4. Subtract 1 from the number in Register 2, unless that number is already 0.
5. Go to instruction 1.
6. If the number in Register 2 is 0, go to instruction 8.
7. STOP.
8. Add 1 to the number in Register 1.
9. STOP.
10. Subtract 1 from the number in Register 1, unless that number is already 0.
11. If the number in Register 1 is 0, go to instruction 9.
12. Go to instruction 10.

1. Write a register program that calculates Min(x,y).

2. Write a register program that calculates Max(x,y).

3. Write a register program that calculates (x + y).

4. Write a register program that calculates (x• y).

5. Show that the function #, given by the following specification, is a Σ function:

   x # 0 = 1.
   x # sy = (x E (x # y)).

6. Show that, if f is a Σ total function, the function g given by

$$g(n) = \sum_{i=0}^{n} f(i)$$

is Σ.

7. Recall that the closed terms of the language of arithmetic constitute the smallest class of expressions that:

    contains "0"

    contains $s\tau$ whenever it contains $\tau$: and

    contains $(\tau + \rho)$, $(\tau \bullet \rho)$, and $(\tau \mathrm{E} \rho)$ whenever it contains $\tau$ and $\rho$.

    Give an algorithm for determining whether a string of symbols is a closed term.

8. Let $\mathfrak{A}$ be a nonstandard model of true arithmetic. Show that there is no formula $\phi(x)$ of the language of arithmetic that is satisfied by all the standard numbers of the model and none of the nonstandard numbers.

9. Prove the Overspill Principle: If $\phi(x)$ is a formula of the language of arithmetic that is satisfied by arbitrarily large standard elements of a nonstandard model $\mathfrak{A}$ of true arithmetic, then $\phi(x)$ is satisfied by some nonstandard elements as well.