COSC 4111 3.0—Fall 2002

Posted: Oct 24, 2002
Due: In about four weeks [Exact date TBA]

Problem Set No. 2

(1) Prove that $\lambda x.A_x(2) \notin \mathcal{PR}$, where $\lambda nx.A_n(x)$ is **our version** of the Ackermann function. Your proof must follow this path:

  (a) Prove that $A_n(x) < A_x(2)$ a.e. with respect to x.

  (b) Using the previous result, prove that if $\lambda x.f(x) \in \mathcal{PR}$, then $f(x) < A_x(2)$ a.e.

  (c) Conclude the argument.

(2) Prove that if $\lambda\vec{y}.f(\vec{y}) \in \mathcal{P}$ and $Q(\vec{x}, z) \in \mathcal{P}_*$, then $Q(\vec{x}, f(\vec{y})) \in \mathcal{P}_*$. Keep in mind the definition ("1-point-rule")

$$Q(\vec{x}, f(\vec{y})) \stackrel{\text{Def}}{\equiv} (\exists z)(z = f(\vec{y}) \wedge Q(\vec{x}, z))$$

(3) Prove that if the graph of f is r.e. then $f \in \mathcal{P}$. Is the converse true? Why?

(4) Using the above and closure properties of $\mathcal{P}_*$ (cf. posted "Kleene" paper) prove that $\mathcal{P}$ is closed under definition by so-called "positive cases" (these are r.e. cases). That is, if all the f_i are in $\mathcal{P}$, all the Q_i are in $\mathcal{P}_*$ and g below is a function, then $g \in \mathcal{P}$.

$$g(\vec{x}) = \begin{cases} f_1(\vec{x}) & \text{if } Q_1(\vec{x}) \\ f_2(\vec{x}) & \text{if } Q_2(\vec{x}) \\ \vdots & \vdots \\ f_k(\vec{x}) & \text{if } Q_k(\vec{x}) \\ \uparrow & \text{otherwise} \end{cases}$$

Hint. Use the previous exercise and work with the graph of g.

(5) Do #22 and #23, p.126 of text.

(6) Prove that there is an $e \in \mathbb{N}$ such that $\phi_e(x) = e$ for all x. That is, "program e outputs itself for every input value x".

Hint. Use the recursion theorem.

(7) For your amusement: Can you write a "self-reproducing" program, such as e above, in your favourite programming language? This program, on every input, just prints itself—exactly, i.e., prints nothing else—and halts.

(Actually, I am inviting you to write and fully test one; I am not just looking for a yes/no answer :-)