Due: Monday, Dec 7, beginning of lecture

General Hint: The Kleene Normal Form Theorem (KNFT) is useful for the solutions for several of the following problems.

1. Do Exercise 11, page 79 in the Notes. (A function $f(x)$ is computable iff its graph is recursively enumerable. To show the 'if' direction, first give an informal algorithm for computing f from an enumeration of the tuples in graph(f). Then formalize the argument by showing that f is recursive, using the least number operator μ.)

   Solution:
   ($\Rightarrow$) Suppose that $f(x)$ is computable by program $\{e\}$. Then by the KNFT its graph can be defined as follows:

   $$R_f(x,y) = (y = f(x)) = \exists w(T(e,x,w) \wedge U(w) = y)$$

   Thus R_f satisfies the definition of r.e. using the recursive relation $S(x,y,w)$ defined by

   $$S(x,y,w) = (T(e,x,w) \wedge U(w) = y)$$

   ($\Leftarrow$) Informal proof: To compute $f(x)$, enumerate all pairs (z,y) in the graph of f, until a pair of the form (x,y) appears. Then set $f(x) = y$. Note that if $f(x)$ is not defined, then this procedure never halts (which is fine).

   ($\Leftarrow$) Formal proof: Suppose $R_f(x,y) = (y = f(x))$ is r.e. We will show that f is recursive.

   By definition of r.e. there is a recursive relation $S(x,y,w)$ such that

   $$(y = f(x)) = \exists w S(x,y,w)$$

   We use the pairing functions and inverses J, K, L (page 77) to show f is recursive:

   $$f(x) = K(\mu z S(x, K(z), L(z))$$

2. Let ODD be the set of all odd integers $n \geq 0$. Define

   $$A = \{x \mid range(\{x\}_1) \subseteq ODD\}$$

   Is A r.e.? Is A^c r.e.? Justify your answers, using the special case of the S-m-n Theorem (page 73 of the Notes.)

   Solution:
   First we show that A is not r.e. by showing $K^c \leq_m A$. We want a total computable f such that

   $$x \in K^c \Leftrightarrow f(x) \in A$$

This is equivalent to

$$\{x\}_1(x) = \infty \Leftrightarrow range(\{f(x)\}_1) \subseteq ODD$$

We use the S-m-n Theorem to define f as follows:

$$\{f(x)\}_1(y) = g(x, y) = 0 \cdot \mu w T(x, x, w)$$

Thus if $x \in K^c$ then $range(\{f(x)\}_1) = \varnothing$ so $f(x) \in A$, and if $x \in K$ then $0 \in range(\{f(x)\}_1)$ so $f(x) \notin A$.

Now we show that A^c is r.e. Note that $x \in A^c$ iff there is at least one even number in the range of $\{x\}_1$. Therefore, by the Kleene Normal Form Theorem and the inverse pairing functions K, L (Notes page 77) we have

$$x \in A^c \Leftrightarrow \exists y(T(x, K(y), L(y)) \wedge EVEN(U(L(y))))$$

where $EVEN(z)$ holds iff z is an even number. Since the relation $R(x, y)$ defined after $\exists y$ is recursive, it follows that A^c is r.e.

3. Let $R(x, y)$ be a recursive relation. Give a primitive recursive relation $S(x, z)$ such that for all $x \in \mathbb{N}$

$$\exists y R(x, y) \Leftrightarrow \exists z S(x, z)$$

Justify your answer.

Solution:
Since R is computable, by the Kleene Normal Form Theorem there is a number e such that

$$R(x, y) = U(\mu z T_2(e, x, y, z))$$

Since the number z coding a halting computation of a Register Machine always exceeds the inputs x, y to the machine it follows that

$$\begin{aligned} \exists y R(x, y) &\Leftrightarrow \exists y \exists z \big(T_2(e, x, y, z) \wedge U(z) = 0\big) \\ &\Leftrightarrow \exists z \exists y \leq z \big(T_2(e, x, y, z) \wedge U(z) = 0\big) \end{aligned}$$

Thus let $S(x, z) = \exists y \leq z\big(T_2(e, x, y, z) \wedge U(z) = 0\big)$.

4. Show how to handle the $\exists \leq$ case in the proof of the MAIN LEMMA (Exercise 6, page 104). (See Lemma A.)

Solution:
Suppose $A =_{syn} \exists x \leq t B(x)$ where $B(x)$ is a bounded formula, and suppose that A is in **TA**. We are to show $\mathbf{RA}_{\leq} \vdash A$; that is

$$\mathbf{RA}_{\leq} \vdash \exists x(x \leq t \wedge B(x)) \tag{1}$$

Since A is a sentence, t is a closed term (no variables), so by Lemma A, $\mathbf{RA}_{\leq} \vdash t = s_n$ for some $n \in \mathbb{N}$. Since A is true, it follows that $B(s_k)$ is true for some $k \leq n$. Hence by the induction hypothesis in the proof of the MAIN LEMMA (bottom of page 102), $\mathbf{RA}_{\leq} \vdash B(s_k)$. Also $\mathbf{RA}_{\leq} \vdash s_k \leq s_n$ by the base case of the induction. Hence

$$\mathbf{RA}_{\leq} \vdash (s_k \leq t \wedge B(s_k))$$

Thus (1) follows.

5. Show that $\mathbf{PA} \vdash \forall x \forall y\; x + y = y + x$. Your proof should follow the style of Example 2, page 98 of the Notes.

   Solution: First we show

   Lemma 1: PA proves $\forall x\; 0 + x = x$.

   Proof: We show $0 + x = x$ by induction on x.

   BASIS: $x = 0$

$$0 + 0 = 0 \text{ by P3}$$

   Induction Step: $x \leftarrow sx$

$$\begin{aligned} 0 + sx &= s(0 + x) && \text{P4} \\ &= sx && \text{Induction Hypothesis} \end{aligned}$$

   Lemma 2: PA proves $\forall x \forall y (s(y + x) = sy + x)$.

   Proof: We show $s(y + x) = sy + x$ by induction on x with parameter y.

   BASIS: $x = 0$

$$\begin{aligned} s(y + 0) &= sy && \text{P3} \\ &= sy + 0 && \text{P3} \end{aligned}$$

   Induction Step: $x \leftarrow sx$

$$\begin{aligned} s(y + sx) &= ss(y + x) && \text{P4} \\ &= s(sy + x) && \text{Induction Hypothesis} \\ &= sy + sx && \text{P4} \end{aligned}$$

   Now we show $x + y = y + x$ by induction on y, with parameter x.

   BASIS: $y = 0$

$$\begin{aligned} x + 0 &= x && \text{P3} \\ &= 0 + x && \text{Lemma 1} \end{aligned}$$

   Induction Step: $y \leftarrow sy$

$$\begin{aligned} x + sy &= s(x + y) && \text{P4} \\ &= s(y + x) && \text{Induction Hypothesis} \\ &= sy + x && \text{Lemma 2} \end{aligned}$$