Problem Set 6

Fall 11

Due: Tuesday, 8th November, 2011, 11:00 am before class begins
Please follow the homework format guidelines posted on the class web page:
`http://www.cs.illinois.edu/class/fa11/cs373/`

1. Reduction à la Rice's Theorem [**Category**: Proof, **Points**: 30]

   Show the following languages are undecidable. You may *not* simply appeal to Rice's theorem (however, you can *adapt* the proof of Rice's theorem to solve these particular problems).

   (a) $L_{\texttt{cs373}} = \{\langle M \rangle \mid M \text{ is a TM and } L(M) \text{ contains the string “CS373”.}\}$
   (b) $L_{\mathit{finite}} = \{\langle M \rangle \mid M \text{ is a TM and } L(M) \text{ is finite.}\}$
   (c) $L_{\mathit{reject}} = \{\langle M \rangle \mid M \text{ is a TM and } M \text{ rejects all inputs.}\}$

Solution:

(a) We will show this by reducing A_{TM} to $L_{\texttt{cs373}}$. Since A_{TM} is undecidable, it follows that $L_{\texttt{cs373}}$ is undecidable.

Assume there is a TM/oracle $M_{\texttt{cs373}}$ that decides $L_{\texttt{cs373}}$.

We build a Turing machine M_u that uses $M_{\texttt{cs373}}$ and decides A_{TM}, as follows.

On input $\langle M, w\rangle$, M_u will build a TM M_w (shown below) such that M_w accepts "CS373" iff M accepts w.
It will feed $\langle M_w\rangle$ to $M_{\texttt{cs373}}$.
If $M_{\texttt{cs373}}$ accepts M_w, then M_u will halt and accept $\langle M, w\rangle$; otherwise M_u halts and rejects $\langle M, w\rangle$.

What remains is the construction of M_w; M_u builds M_w as the following TM.

1. Input: x
2. Erase x and replace it with the constant string w.
3. Simulate M on w.
4. If M accepts w, then accept; if M rejects w, then reject.

Note that:
If M accepts w, then $L(M_w) = \Sigma^*$, and thus contains "CS373";
If M does not accept w, $L(M_w) = \emptyset$, and thus does not contain "CS373".

Hence M_w accepts "CS373" iff M accepts w.

Hence the Turing machine M_u above decides A_{TM}, proving the reduction.
Hence $L_{\texttt{cs373}}$ is undecidable.

(b) We will show this by reducing A_{TM} to $L_{\texttt{finite}}$. Since A_{TM} is undecidable, it follows that $L_{\texttt{finite}}$ is undecidable.

Assume there is a TM/oracle $M_{\texttt{finite}}$ that decides $L_{\texttt{finite}}$.

We build a Turing machine M_u that uses $M_{\texttt{finite}}$ and decides A_{TM}, as follows.

On input $\langle M, w\rangle$, M_u will build a TM M_w (shown below) such that $L(M_w)$ is *not* finite iff M accepts w.
It will feed $\langle M_w\rangle$ to $M_{\texttt{finite}}$.
If $M_{\texttt{finite}}$ accepts M_w, then M_u will halt and reject $\langle M, w\rangle$; otherwise M_u halts and accepts $\langle M, w\rangle$.

What remains is the construction of M_w; M_u builds M_w as the following TM.

1. Input: x
2. Erase x and replace it with the constant string w.
3. Simulate M on w.
4. If M accepts w, then accept; if M rejects w, then reject.

Note that:
If M accepts w, then $L(M_w) = \Sigma^*$, and thus is not finite;
If M does not accept w, $L(M_w) = \emptyset$, and thus is finite.

Hence $L(M_w)$ is *not* finite iff M accepts w.

Hence the Turing machine M_u above decides A_{TM}, proving the reduction.
Hence $L_{\texttt{finite}}$ is undecidable.

(c) We will show this by reducing A_{TM} to $L_{\texttt{reject}}$. Since A_{TM} is undecidable, it follows that $L_{\texttt{reject}}$ is undecidable.

Assume there is a TM/oracle $M_{\texttt{reject}}$ that decides $L_{\texttt{reject}}$.

We build a Turing machine M_u that uses $M_{\texttt{reject}}$ and decides A_{TM}, as follows.

On input $\langle M, w\rangle$, M_u will build a TM M_w (shown below) such that M_w *does not* reject all inputs iff M accepts w.
It will feed $\langle M_w\rangle$ to $M_{\texttt{reject}}$.
If $M_{\texttt{reject}}$ accepts M_w, then M_u will halt and reject $\langle M, w\rangle$; otherwise M_u halts and accepts $\langle M, w\rangle$.

What remains is the construction of M_w; M_u builds M_w as the following TM.

1. Input: x
2. Erase x and replace it with the constant string w.
3. Simulate M on w.
4. If M accepts w, then reject; if M rejects w, then accept.

Note that:
If M accepts w, then $L(M_w) = \Sigma^*$, and thus does not reject all inputs;
If M does not accept w, $L(M_w) = \emptyset$, and thus rejects all inputs.

Hence M_w does not reject all inputs iff M accepts w.

Hence the Turing machine M_u above decides A_{TM}, proving the reduction.
Hence $L_{\texttt{reject}}$ is undecidable.