UR Honor Pledge for Exams “I affirm that I will not give or receive any unauthorized help on this exam, and that all work will be my own.” (UR now requires you to by hand copy the above sentence onto the blank lines below and then sign it.)

Signature: _____________________________________________________________

Instructions: There are six problems here. Do four. If you choose to answer more than four problems (you may if you want to), we will grade all that you answer, and will count the highest (that is not a typo; it really says and means “highest”) four scores.

The highest possible score on this exam is 100%, i.e., 100/100 (see, at your leisure, the 150928 “Midterm I Tidbit,” which is related to this.)

Throughout this exam, $HP$ denotes $\{i | M_i(i) \text{ accepts}\}$. Note carefully that the word there is not “halts” but is “accepts.”

Don’t just jot down random stuff (doing so may LOSE points)—think before answering (we’ll hand out scrap paper so you have scrap paper on which to doodle/plan/try-things).

Note that not all questions are necessarily identical in difficulty, so don’t, for example, make the mistake of spending all your time on one hard problem and leaving the others blank.

Don’t confuse “from” and “to” on reductions (if when trying to prove $A \leq_m B$ you accidentally just prove $B \leq_m A$, you’ll probably get no points). Note, for example, that there is a many-one reduction from $\emptyset$ to $HP$, but there exists no many-one reduction from $HP$ to $\emptyset$. (On this exam, “many-one” of course refers to recursive many-one reductions.)

Your handwriting must be clear and readable. We will not guess that some unclear character is what is needed to make your answer correct; your written characters must be clearly readable on their own.

Rules: Closed book, closed notes, no computers or calculators, use pen not pencil. An exception to the “closed notes” rule is that each person can have one sheet of self-prepared, hand-written notes.

Make sure to put all your answers that you want graded onto (not the scrap paper but rather) the stapled test sheets.

Scoring: To repeat: There are six problems here. Do four. If you choose to answer more than four problems (you may if you want to), we will grade all that you answer, and will count the highest four scores. Since your grade will be based on four 25-point problems, there are 100 points available here.

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td></td>
</tr>
</tbody>
</table>

Total (of highest 4)
Question 1 [25 points]

(a) [5 points] Give the definition of what it means for $A$ to be a recursive set.

Answer: A is a recursive set exactly if there exists a Turing machine $M$ such that $L(M) = A$ and $M$ halts on each input.

(b) [20 points] Professor Foo claims that the following result holds: A set $B$ is recursive if and only if every Turing machine accepting $B$ is a total Turing machine. Prove or disprove this result.

Answer: We will disprove Foo’s claim by breaking the “only if” direction. Consider $B = \emptyset$. $B$ is recursive and there exists a TM $M$ that accepts $B$ by running forever on all inputs. Clearly, $M$ is not a total machine.

Question 2 [25 points] For each part, please place each of the sets given into the lowest level of the Kleene hierarchy (among the ones listed in the next sentence) that contains it. Choose from the levels: $\Sigma_0$, $\Sigma_1$, $\Pi_1$, $\Sigma_2$, $\Pi_2$, etc.

Hints: Recall that $\Sigma_0 = \Pi_0 = \text{the recursive sets}; \Sigma_1 = \text{the recursively enumerable sets}; \Pi_1 = \text{coRE sets}$. Recall the quantifier structures of the levels (informally put, $\Sigma_1$ is $\exists$, $\Sigma_2$ is $\exists\forall$, $\Pi_1$ is $\forall$, $\Pi_2$ is $\forall\exists$, $\Pi_3$ is $\forall\exists\forall$, $\Sigma_3$ is $\exists\forall\exists$, etc.). Rice’s Theorems might (or might not) be helpful on some of these. Your knowledge of complete sets, many-one reductions, and Turing reductions might (or might not) be helpful. Your ability to see quantifier structure might (or might not) be helpful.

There is no penalty for guessing on this problem so you might want to answer everything. Please use the 1-inch answer lines for the answers. Do all scrap work on scrap paper; the scrap work will not be collected.

**Note:** The order of these 5 parts might be different on your exam paper.

1. [Place in the best level from the choices mentioned above]
   \{i | L(M_i) is finite\}.
   Answer: $\Sigma_2$.

2. [Place in the best level from the choices mentioned above]
   \{(x, i) | x \notin L(M_i)\}.
   Answer: $\Pi_1$.

3. [Place in the best level from the choices mentioned above]
   \{i | L(M_i) is an r.e. set\}.
   Answer: $\Sigma_0$.

4. [Place in the best level from the choices mentioned above]
   \{(i, j) | L(M_i) = \emptyset \text{ and } L(M_j) = \Sigma^*\}.
   Answer: $\Pi_2$.

5. [Place in the best level from the choices mentioned above]
   \{i | L(M_i) \in \Pi_{2009} - \Sigma_0\}.
   Answer: $\Pi_3$.

**Question 3 [25 points] (a) [12.5 points]** Prove by a clear Tarski-Kuratowski quantifier analysis that \{i#j | L(M_i) = L(M_j)\} is in $\Pi_2$.

**Answer:**

\[\{i \# j | (\forall x)((\forall t)[M_i \text{ does not accept } x \text{ within } t \text{ steps}] \lor (\exists t)[M_i \text{ accepts } x \text{ within } t \text{ steps}]) \land ((\forall t)[M_j \text{ does not accept } x \text{ within } t \text{ steps}] \lor (\exists t)[M_j \text{ accepts } x \text{ within } t \text{ steps}])\}\]

Thus, the longest alternating chain of quantifiers is $\forall \exists$ and so this set is in $\Pi_2$. 

2
(b) [12.5 points] Let $R = \{i \mid M_i$ is a total Turing machine$. State Rice’s Theorem (the easy 1-part theorem, not the 3-part theorem). Then either use it to prove that $R$ is not recursive, or explain clearly why Rice’s Theorem is not appropriate for this problem.

**Answer:** Rice’s Theorem: Any nontrivial property $P$ of the r.e. languages is undecidable.

It is not appropriate as this is not a language property. (There exists a total machine accepting $\Sigma^*$ and there exists a total machine accepting $\emptyset$, and so this is clearly not a language property.)

**Question 4** [25 points] Let $A = \{j \# k \mid L(M_j) \subseteq L(M_k)\}$. Recall that $HP = \{i \mid i \in L(M_i)\}$, i.e., those $i$ such that $M_i$ accepts $i$ (note it says “accepts,” not “halts”). Give an explicit, direct, many-one reduction (and show that it is indeed a reduction between those) from $HP$ to $A$ (i.e., $HP \leq_m A$).

**Answer:** $\sigma(i)$ will output $j(i)$#k. $k$ will be the index of a fixed TM accepting $\emptyset$. $j(i)$ will be the index of a TM that on arbitrary input $y$ ignores $y$ and simulates $M_i(i)$.

$\sigma$ is clearly a recursive function.

- $i \in HP \Rightarrow L(M_{j(i)}) = \Sigma^* \Rightarrow L(M_{j(i)}) \subseteq L(M_k) \Rightarrow \sigma(i) \in A$.
- $i \notin HP \Rightarrow L(M_{j(i)}) = \emptyset \Rightarrow L(M_{j(i)}) \subseteq L(M_k) \Rightarrow \sigma(i) \notin A$.

So $HP \leq_m A$ via $\sigma$.

**Question 5** [25 points] Prove that there exists an infinite set $A \subseteq \{0,1\}^*$ such that (a) $A$ has no infinite r.e. subset, and (b) $(\forall n \in \{0,1,2,\ldots\})(|A \cap \{0,1\}^n| = n)$. (Part (b) says that at each length $n$ the set $A$ contains exactly $n$ of the strings of length $n$.)

**Answer:** We will use a variant of diagonalization to construct a set $A$ as follows. Take our standard enumeration of TMs, remove all that accept finite sets, and consider a “table” where the $i$’th row (our row indexes start with row 1) corresponds to the (infinite!) language accepted by the $i$’th machine left on our list, call that r.e. language $S_i$, and the $n$’th “column” (it is actually a bunch of strings we will handle together) corresponds to all strings of length $n$ (our column indexes start with column 0). Don’t include $\epsilon$ in $A$ and initialize $n = 1$ and $i = 1$.

We execute the following steps ad infinitum. If some string $w$ belonging to the $n$’th column is in $S_i$, include $n$ length-$n$ strings in $A$ while making sure that $w$ is not among the included strings (i.e., put in the first $n$ strings related to that column that are now $w$), increment $n$ by one and increment $i$ by one. Otherwise, if there are no strings of the $n$’th column in $S_i$, include the first (lexicographically) $n$ length-$n$ strings in $A$ and increment $n$ by one (but as that each row is related to an infinite set, we eventually will get to a column of this row that does have a string in $S_i$).

Since there are infinitely many machines accepting infinite r.e. sets, the above process will include infinitely many strings in $A$ and at the same time we make sure that for each $S_i$ it holds that there exists a string $s$, such that $s \in S_i$ and $s \notin A$, so each $S_i$ (i.e., each infinite r.e. set) is not a subset of $A$.

**Question 6** [25 points] Let $B = \{i \mid L(M_i)$ is infinite$\}$. Give an explicit, direct, many-one reduction (and show that it is indeed a reduction between those) from $\overline{HP}$ to $B$ (i.e., $\overline{HP} \leq_m B$).

**Answer:** $\sigma(i)$ is the index of a TM that on arbitrary input $y$ does the following:

It simulates $M_i(i)$ for $|y|$ steps, and if $M_i(i)$ does not accept within $|y|$ steps then $M_{\sigma(i)}$ accepts $y$ and otherwise it rejects $y$.

$\sigma$ is clearly a recursive function.

- $i \in \overline{HP} \Rightarrow L(M_{\sigma(i)}) = \Sigma^* \Rightarrow \sigma(i) \in B$.
- $i \notin \overline{HP} \Rightarrow L(M_{\sigma(i)})$ finite $\Rightarrow \sigma(i) \notin B$. 
