Complexity Theory Exercise 2

Submitter: Daniel Alfasi 308399542

Question 1:

We call $L_1 \in C$ a complete language in C if for every other language $L_2 \in C$, L_2 is reducible to L_1.

A. Prove or disprove, there exist a complete language in $RE \bigcup coRE$:
   Suppose that there exist a complete language L_1 in $RE \bigcup coRE$ and divide into 3 different situations:
   1. If $L_1 \in RE / R$, one can choose $L_2 = \overline{L_1}$. By definition of RE and $coRE$ we have that $L_2 \in coRE / R$. By the assumption we have that L_2 is reducible to L_1 which means that is also in RE / R **which is not possible!** since RE / R and $coRE / R$ are disjoint sets.
   2. If $L_1 \in coRE / R$ we can use the same claim as (1).
   3. $L_1 \in R$, one can choose $L_2 \in RE / R$. By the assumption any language in $RE \bigcup coRE$ is reducible to L_1 but we have that $L_2 \in RE / R \subseteq RE \cup coRE$ **is not reducible to** L_1. Since then we have that $L_2 \in R$ by the reduction **which is not possible!** .

B. Prove or disprove, there exist a complete language in $coRE$:
   We have that any language $L_2 \in RE$ is reducible to HP $\Rightarrow HP \geq L_2$, By the reduction rules also $\overline{HP} \geq \overline{L_2} \in coRE$. Therefore $\overline{HP}$ is an example for complete language in $coRE$.

C. Prove or disprove, there exist two non-trivial languages L_1, L_2 such that neither of them is reducible to the other:
   One can choose $L_1 = HP$ and $L_2 = \overline{HP}$, we have that $L_1 \in RE / R$ and $L_2 \in coRE / R$ which means neither of them is reducible to the other (We showed that idea also in A).

Question 2:

For each language prove or disprove if it's in R and if it is in RE:

A. $L = \{<M> \mid L(M) \in R\}$
   We have that property $S = R$ is not trivial, therefore the given language is not in R.
   Now, we have that $\varnothing \in R \Rightarrow \varnothing \in S$ (one can build a Turing machine which decides $\varnothing$), by Rice theorem for RE we have that $L \notin RE$.

B. $L = \{<M> \mid M \text{ halts on all inputs}\}$
   $L \notin RE$:
   We show by reduction that $\overline{HP} \leq L$ using the following reduction function $f(<M'>, <w>) = <M>$. The reduction function is actually a Turing machine which acts very simply, for input x run M in the following way $M(x)$:
   - Run M' on w for $|x|$ steps:
   - If M' halts (during the $|x|$ steps) then enter a **endless loop**
   - Otherwise, if M' didn't stop after $|x|$ steps (we are on the step $|x|+1$) then **halt**

Correctness of f:

- If $(<M'>,<w>) \in \overline{HP}$ then $M'(w)$ never halts, so $M(x)$ will eventually stop after $|x|$ steps so $M \in L$
- If $(<M'>,<w>) \notin \overline{HP}$ then $M'(w)$ will halt eventually, so $M(x)$ should never halt! since $M \notin L$ (language description, we accept Turing machines which halts on all inputs).

C. $L = \{<M> | L(M) \in coRE\}$

We have that the property $S = coRE$ is not trivial, therefore , $L \notin R$.
Since we know that there exist such a Turing machine which accept L for each $L(M) \in coRE$ **<u>from our language</u>** (which is M), we have that $S' = L_s \cap RE = R \Rightarrow$ we are dealing with the same language as (A). therefore, $L \notin RE$.

Question 3:

A. If $L_1 \leq_T L_2$ and $L_2 \in R$ then $L_1 \in R$, **proof**:

One can build a Turing machine M_1 which decide the language L_1, denote M_2 as the Turing machine which decides L_2 (We know that exist one as $L_2 \in R$).
M_1 acts in the following way for input x:

1. run x on M_{12}
2. Denote y as the input (the characters in the left side of the 2nd tape)
3. Whenever M_{12} gets into the special state q_{ask}:
   1. **instead of** running the "magical computation" on y **run** M_2 with y
   2. If $M_2(y) = accept$ then move to q_{yes}
   3. Otherwise, move to q_{no}
4. Continue the computation and if M_{12} gets again to the special state q_{ask} jump to (3)
5. Output same as M_{12}

Note that the building of M_1 is valid since we no more using the magical computation. Instead of that as we know that $L_2 \in R$ we use it's machine for this part of computation and continue "normally" the computation, this is no more a "magical Turing machine".

B. If $L_1 \leq_T L_2$ and $L_2 \in RE$ then $L_1 \in RE$, **disproof**:

Let $L_2 = HP$ and $L_1 = \overline{HP}$, since we know that $L_1 \notin RE$ we have to show such a Turing machine M_{12} (which disproof the claim).
The machine M_{12} takes an input <M>,<x> and acts in the following way:

1. Start the computation by copy the input from the first tape to the second tape
2. After copying the input move to q_{ask} (Where the magic happens :))
3. If the machine move to q_{yes} then the next move will be to q_{reject}
4. Otherwise, the machine moved to q_{no} and out next move will be to q_{accept}

This example work since we know that M_{12} is able to decide RE languages, no extra power were added to the machine offered above.

Since we can decide HP one can use this fact to decide $\overline{HP}$ just by “flipping” the answer.