Assignment
CSE 211

Submitted by,
Sihat Afnan
St ID : 1705098

3. Show that the halting problem, the set of (M, w) pairs such that M halts when given input w is RE but not recursive.

Answer:
Let, H = {(M, w): Turing machine M halts on input w}
We can run M on w through universal TM and accept when M halts. Thus, it can be shown that H is RE.
Now, let H be recursive. So, there exists a Turing machine HM that always halts and L(HM) = H.
Let C be another machine that takes a machine encoding M as input. It runs forever if HM accepts (M, M), and halts if HM rejects (M, M).
Suppose we give encoding of C as input to C i.e. run C on itself, and HM accepts (C, C). So, C will run forever. But it contradicts the fact that HM accepted (C, C) i.e. C is supposed to halt. Now, suppose we run C on itself and HM rejects (C, C). So, C will halt. Again, we get a contradiction, since C is supposed to run forever as HM rejected (C, C). Hence, our assumption of H being recursive was false.
So, H is RE but not recursive.

4. (a) Suppose A and B are two problems in NP. Explain whether the following two statements are true or false:
(i) If B is an NP-complete problem and we find a polynomial time reduction from A to B, then the problem A is also NP-complete.
(ii) If B is an NP-complete problem and we find a polynomial time reduction from B to A, then the problem A is also NP-complete.
(b) Explain why if you find a polynomial time algorithm for an NP-complete problem, it implies P = NP.

Answer:
(a) To conclude that problem A is NP-complete, we need to prove:
1. A is NP
2. a known NP-hard problem is polynomial time reducible to A
The first point is already given in question. The first statement does not include the second point. Now, since all NP-complete problems are NP-hard, B is NP-hard. Hence, the second statement includes the second point.
So, the first statement is false and the second one is true.

(b) We know that all NP problems are polynomial time reducible to any NP-complete problem. If we find a polynomial time algorithm for an NP-complete problem A, we can reduce any NP problem to A in polynomial time and then solve A in polynomial time. Thus, all NP problems can be solved in polynomial time i.e. P = NP.