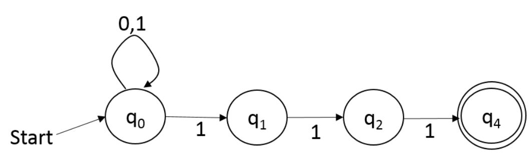
Mathematics of Computing
Final Exam Back-paper (Max Marks 50, Time 3h)

Indian Statistical Institute, Bangalore

June 2018

1. $(2 \times 5 = 10)$ Assume $A \leq_P B$ i.e., A reduces to B in polynomial time by a deterministic Turing machine. Prove or disprove each of these statements:
   (a) If A is NP-Complete then B is NP-Hard.
   (b) If B is Turing recognizable then so A.
   (c) If A is polynomial time Turing decidable then so is B.
   (d) If B has an exponential time algorithm then so does A.
   (e) If B is in P then $P = NP$.

2. $(3 + 3 + 4 = 10)$
   (a) Draw a DFA that accepts all non-negative integer multiples of 3 which are odd.
   (b) For the following NFA give the corresponding regular expression. Then convert the NFA to a DFA using subset construction.



   (c) Show that the language $L = \{a^n | n \text{ is prime}\}$ on the alphabet $\{a\}$ is not regular by using the pumping lemma for regular languages.

3. $(2 + 4 + 4 = 10)$
   (a) Consider the language $L = \{w | w \text{ is a well formed sequence of parentheses}\}$. E.g., the string `(())()(())` belongs to L but `)()(` does not belong to L. Give a CFG for L.
   (b) Consider the grammar with these two rules:

   $$S \rightarrow S + S$$
   $$S \rightarrow a$$

   Show that the grammar is ambiguous. Further, give an unambiguous grammar for the same language.

(c) Use the pumping lemma to prove that the language $L = \{a^n b^n c^n | n > 0\}$ is not context free.

4. $(4 + 6 = 10)$ You may assume, as proved in class, that A_{TM} is Turing recognizable and not Turing decidable.

   (a) Show that the language $HALT_{TM}$ consisting of all $< M, w >$ pairs where the TM M halts on input w is not Turing decidable.

   (b) We know that if $A \leq_m B$, then if A is not Turing recognizable then B is not Turing recognizable. Use this to show that EQ_{TM} is not Turing recognizable and also not co-Turing recognizable. Here EQ_{TM} is the set consisting of pairs $< M_1, M_2 >$ of Turing machines which recognize the same language.

5. $(2 + 2 + 6 = 10)$

   (a) Give an example of an NP-Complete problem and pose it as a language decision problem. Show that members of your example language have a certificate that is checkable in (deterministic) polynomial time.

   (b) Consider the following algorithm to test if a number $n > 2$ is prime: *"For each i in $2..n-1$ test if i divides n. If any of them divides n declare n as composite, else declare n as prime."* Why is this not considered a polynomial time algorithm for determining primeness?

   (c) Consider an undirected graph G. Define a *Hamiltonian Cycle* in a graph G as a simple cycle in G that contains all the vertices of G. Similarly, an $s - t-$*Hamiltonian Path* of a graph G is a simple path that has all vertices of G and has end points s and t which are two vertices of G. Assume that *Hamiltonian Cycle* is NP-Complete and give a reduction to show that $s - t-$*Hamiltonian Path* is NP-Complete.