Computer Science 456/656 Fall 2013

Practice for the Third Examination, Thursday November 7, 2013

The entire practice examination is 285 points.

1. True or False. [5 points each] If the question is currently open, write "O" or "Open."

   (a) _______ Every subset of a regular language is regular.

   (b) _______ The intersection of any two recursive languages is recursive.

   (c) _______ The intersection of any two recursively enumerable languages is recursively enumerable.

   (d) _______ The complement of any recursive language is recursive.

   (e) _______ The complement of any recursively enumerable language is recursively enumerable.

   (f) _______ Every context-free language is recursive. _______

   (g) _______ The problem of whether a given context-free grammar generates all binary strings is decidable.

   (h) _______ For any real number x, Q_x is decidable, where Q_x be the set of all rational numbers which are less than or equal to x. (For example, $\frac{25}{8} \in Q_\pi$, and $\frac{22}{7} \notin Q_\pi$.)

   (i) _______ If $f(n)$ is any integral function on integers, there is some recursive function $F(n)$ such that $F(n) \geq f(n)$.

   (j) _______ The question of whether a given string is generated by a given general grammer is decidable.

   (k) _______ There is a general grammar which generates the set of all binary numerals for primes.

   (l) _______ There is a general grammar which generates HALT, the language equivalent to the halting problem.

   (m) _______ There exists a Turing machine M and string w such that:

      i. M halts with input w.

      ii. There does not exist any proof that M halts with input w.

   (n) _______ If L is a canonically enumerable language, then L must be recursive.

   (o) _______ Every $\mathcal{NP}$-TIME language is recursive.

2. [30 points] Prove that every recursive language is canonically enumerable.

3. [30 points] What is the Church-Turing thesis, and why is it important?

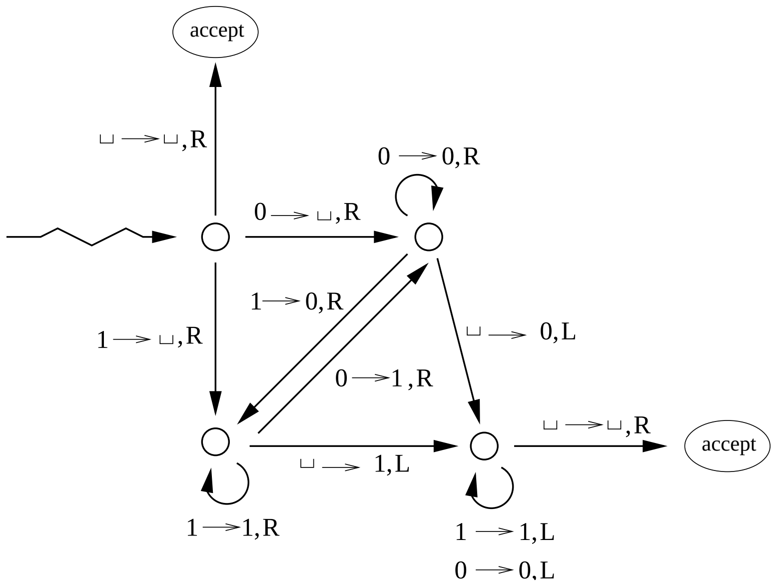
4. [30 points] Prove that every Turing acceptable language is recursively enumerable.

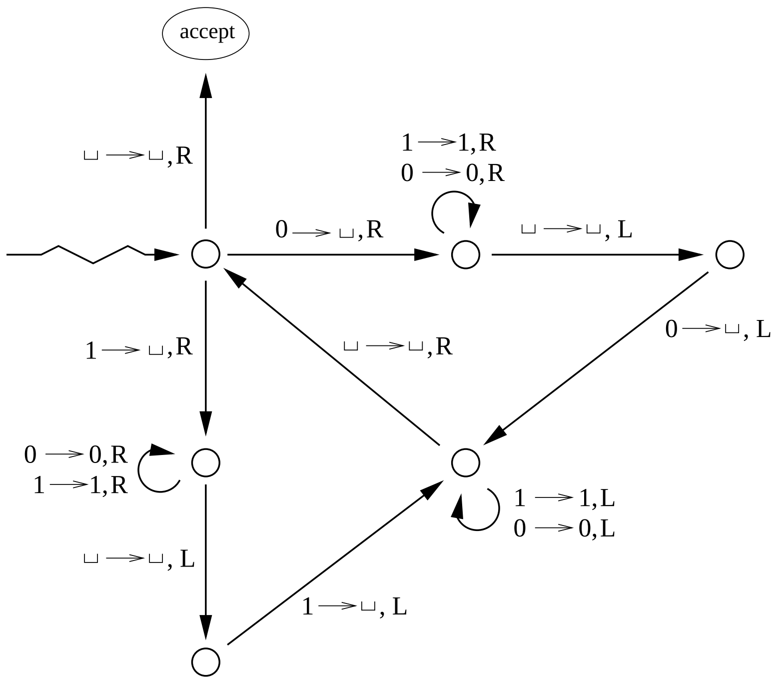
5. [30 points] We say that two programs P_1 and P_2 are *equivalent* if they always produce the same output given the same input. The *grader's problem* is to determine whether two given programs are equivalent. Prove that the grader's problem is undecidable. (You may assume that HALT is undecidable.)

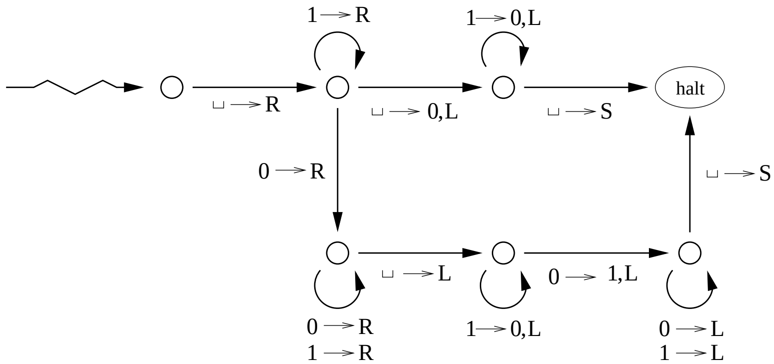
6. [30 points] Give a state diagram for a Turing machine M that computes the canonical incrementation function on binary strings. That is, if w is the input string of M, then the output string will be the successor of w in the canonical order.

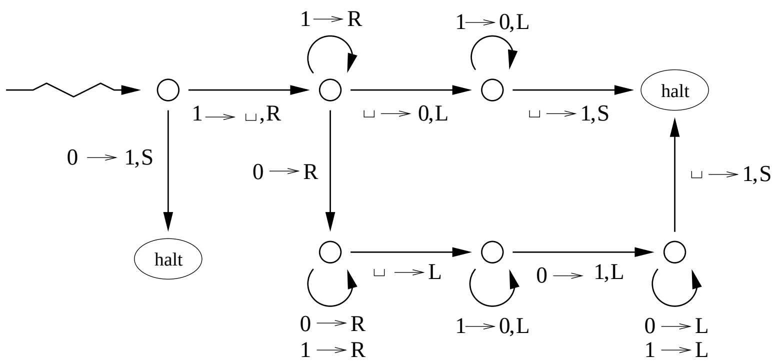
7. [30 points] "Sketch" the construction which shows that a multi-tape TM can be emulated by a single-tape multi-track TM. A formal proof is not necessary. I only want you to convince me that it's true in an informal way, drawing some figures and putting in a little text.

8. [30 points] What does the TM illustrated here do? The answer can be expressed in at most one line. (At most one such problem will be on the test.)


accept
⊔ → ⊔,R
0 → 0,R
0 → ⊔,R
1 → 0,R
1 → ⊔,R
⊔ → 0,L
0 → 1 ,R
⊔ → ⊔,R
accept
⊔ → 1,L
1 → 1,R
1 → 1,L
0 → 0,L


accept
⊔ → ⊔,R
1 → 1,R
0 → 0,R
0 → ⊔,R
⊔ → ⊔, L
0 → ⊔, L
1 → ⊔,R
⊔ → ⊔,R
0 → 0,R
1 → 1,R
1 → 1,L
0 → 0,L
⊔ → ⊔, L
1 → ⊔, L


1 → R
1 → 0,L
⊔ → R
⊔ → 0,L
⊔ → S
halt
0 → R
⊔ → S
⊔ → L
0 → 1,L
0 → R
1 → R
1 → 0,L
0 → L
1 → L



1 → R
1 → 0,L
halt
1 → ⊔,R
⊔ → 0,L
⊔ → 1,S
0 → 1,S
0 → R
⊔ → 1,S
halt
⊔ → L
0 → 1,L
0 → R
1 → R
1 → 0,L
0 → L
1 → L