School of Mathematical and Computational Sciences
Indian Association for the Cultivation of Science

Master's/Integrated Master's-PhD Program/ Integrated Bachelor's-Master's Program/PhD Course

Theory of Computation II: COM 5108

Quiz III (16 November 2023)

Answer All Questions with Brief Justifications

Marks: 6 $\times$ 3 = 18

1. What is the maximum possible number of configurations of an $f(n) \geq \log n$ space bounded (work tape) TM, where n is the input length.

   Ans. Let the number of states be $q = |Q|$, number of tape symbols be $|\Gamma| = c$. The number of configurations are specified by state, head position on input tape, head position on work-tape, and the content of the work-tape. It is $q \times n \times f(n) \times c^{f(n)} = 2^{\log_2 q} \times 2^{\log_2 n} \times 2^{\log_2 f(n)} \times 2^{f(n)\log_2 c} = 2^{(\log_2 q + \log_2 n + \log_2 f(n) + f(n)\log_2 c)} = 2^{O(f(n))}$.

2. Let $\phi(C_1, C_2, t)$ be a TQBF formula corresponding to a $O(n^k)$ space bounded computation of a TM going from configuration C_1 to configuration C_2 in t steps. An intermediate configuration C_m is such that both C_1 to C_m and C_m to C_2 are reached in $\frac{t}{2}$ steps. So we can write

   $$\phi(C_1, C_2, t) = \exists C_m(\phi(C_1, C_m, t/2) \wedge \phi(C_m, C_2, t/2)).$$

   Explain why this scheme cannot give a polynomial size formula.

   Ans. The size of a configuration is cn^k. The t step computation is broken into two $t/2$ step computations. But that almost doubles the size of the formula. So at the end of recursion the size of the formula is t.
   As the total number of steps are 2^{cn^k}, the formula size will be exponential and the reduction cannot be in polynomial time..

3. Let $A = \{x \in \{0,1\}^* :$ number of 0's in x is twice the number of 1's in it$\}$. Is $A \in \mathbf{L}$?

   Ans. The language is in $\mathbf{L}$. Following logspace machine decides A.
   M : input $x = x_0 \cdots x_{n-1}$

   (i) $c_0 \leftarrow 0$, $c_1 \leftarrow 0$.

   (ii) for $i \leftarrow 0$ to $n-1$ do (iii)

(iii) if $x_i = 0$ then $c_0 \leftarrow c_0 + 1$
else $c_1 \leftarrow c_1 + 1$

(iv) if c_0 is odd then **reject**

(v) $c_0 \leftarrow c_0/2$

(vi) if $c_0 = c_1$ then **accept** else **reject**.

The workspace required for two counters is $O(\log n)$.

4. We define the language

$$H_f = \{< M, x >: \text{ the DTM } M \text{ accepts the input } x \text{ in } f(|x|) \text{ steps}\}.$$

We claim that $H_f \notin \mathbf{DTIME}(f(\lfloor \frac{n}{2} \rfloor))$. We prove this by *diagonalization*. Suppose the TM M_f decides H_f within $f(\lfloor \frac{n}{2} \rfloor)$ number of steps. We construct the following TM D_f for diagonalization.

D_f: input: $< M >$
Simulate N_f on $< M, M >$.
if N_f *accepts*, then **reject** else **accept**.

(a) Find the running time of D_f on input of length $n = | < M > |$.

(b) Apply D_f on its own description $< D_f >$ and give the argument for contradiction.

Ans.

(a) The running time of D_f on input $< M >$ ($|M| = n$) is the running time of N_f on input $< M, M >$. So it is order of $f\left(\left\lfloor \frac{|<M,M>|}{2} \right\rfloor\right) = f\left(\lfloor \frac{2n+1}{2} \rfloor\right) = f(n)$.

(b) Apply D_f on its own description $< D_f >$. D_f will halt within $f(|D_f|)$ number of steps. There are two possibilities.
If $D_f(< D_f >) = 1$, then $N_f(< D_f, D_f >) = 0 \equiv < D_f, D_f > \notin H_f \Rightarrow D_f(D_f) = 0$ (D_f does not accept $< D_f >$ in $f(| < D_f > |)$ steps).
If $D_f(< D_f >) = 0$, then $N_f(< D_f, D_f >) = 1 \equiv < D_f, D_f > \in H_f \equiv D_f(< D_f >) = 1$.
So $D_f(< D_f >) = 1$ if and only if $D_f(< D_f >) = 0$ - a contradiction.

5. $A_{cof} = \{< M >: \overline{L(M)} = \Sigma^* \setminus L(M) \text{ is finite}\} \in \Sigma^0_n$. What is the least value of n? Give a definition of A_{cof} based on recursive predicate.

Ans. $A_{cof} \in \Sigma^0_3$. $A_{cof} = \{< M >: \exists n \forall x \exists t\ R(< M >, n, x, t)\}$, where the recursive predicate $R(< M >, n, x, t)$ is "if $|x| \geq n$, then M *accepts* x in t steps".

6. People claim that $\mathbf{PH} = \bigcup_{n \geq 0} \Sigma^p_n$ cannot have a complete language. If L is a **PH**-complete language it must belong to Σ^p_i for some i.
Explain what can be the conclusion in that case?

Ans. All $L' \in \mathbf{PH}$ can be reduced to L, $L' \leq^p_m L$. This implies that $L' \in \Sigma^p_i$ i.e. $PH = \Sigma^p_i$.