Formal Languages and Automata Theory

Third Short Test

Time: 20 minutes | 15–April–2021 | Maximum marks: 20

---

1. Let M be Turing machine over the alphabet Σ with $\mathscr{L}(M) = L$. Let M' be the Turing machine obtained from M by swapping the roles played by the accept and the reject states of M. Finally, let $L' = \mathscr{L}(M')$, and $\sim L$ the complement of L (in Σ^*). Which of the following statements is always true?

(A) $L' = \sim L$
(B) $L' \neq \sim L$
[(C)] $L' \subseteq \sim L$
(D) $\sim L \subseteq L'$

Solution M may implicitly reject some strings by looping.

2. Which of the following statements is true?

(A) RE languages are closed under union, intersection and complementation.
[(B)] RE languages are closed under union and intersection but not under complementation.
(C) RE languages are closed under union and complementation but not under intersection.
(D) RE languages are closed under neither of union, intersection and complementation.

3. Consider the language

$$L = \left\{ M \# N \# x \;\middle|\; M, N \text{ are Turing machines over } \Sigma,\ x \in \Sigma^*, \text{ and both } M \text{ and } N \text{ accept } x \right\}.$$

Which of the following statements is true?

[(A)] L is recursively enumerable.
(B) L is not recursively enumerable.
(C) If $x \in \mathscr{L}(M) \cup \mathscr{L}(N)$, then $M \# N \# x \in L$.
(D) If $x \in \mathscr{L}(M)$ but $x \notin \mathscr{L}(N)$, then $M \# N \# x \in L$.

Solution (A) Run M and N on x in parallel. Accept if both simulations accept. Reject if one of the simulations rejects. Loop if both simulations loop.

4. Let L_i, $i \in I$, be a family of recursive languages, and $L = \bigcup_{i \in I} L_i$. Which of the following statements is true?

[(A)] If I is finite, then L must be recursive.
(B) If I is infinite, then L must be recursive.
(C) If L is recursive, then I must be finite.
(D) If L is recursive, then I must be infinite.

Solution (A) Let D_i be a decider for L_i, and $|I| = n < \infty$. Upon input x, simulate the deciders D_i in parallel (on a time-sharing basis), each on input x. Accept if any of the deciders accept. Reject if all of the deciders reject. Since each D_i is a decider, it never loops on any input.

(B) Take $I = \mathbb{N}$, any non-recursive language $L = \{w_i \mid i \in \mathbb{N}\}$, and $L_i = \{w_i\}$.

(C) and (D) Take $L_i = L$ for all $i \in I$. Then irrespective of whether I is finite or not, L is recursive.

5. A 2-tape NTM is equivalent to:

[(A)] A 1-tape DTM
(B) A PDA
(C) An NFA
(D) None of the machines in the other options

6. What language is generated by the unrestricted grammar $G = (\{S,B,a,b,c\},\{a,b,c\},R,S)$, where R consists of the following productions?

$$\begin{aligned} S &\to aBSccc \mid aBccc \\ Ba &\to aB \\ Bc &\to bbc \\ Bb &\to bbb \end{aligned}$$

(A) $\{a^n b^{3n} c^{3n} \mid n \geqslant 0\}$
(B) $\{a^n b^{2n} c^{3n} \mid n \geqslant 0\}$
(C) $\{a^n b^n c^n \mid n > 0\}$
(D) $\{a^n b^{2n} c^{3n} \mid n > 0\}$

Solution Initially the productions involving S generate $(aB)^n c^{3n}$ for $n > 0$. In order that a B so generated vanishes, it must come in contact with a b or a c. When this happens, each B contributes two b's.

7. A language L over an alphabet Σ satisfies the condition

$$\forall M \Big[\big(L = \mathscr{L}(M)\big) \Rightarrow M \text{ is a total Turing machine} \Big],$$

where M is taken from the set of all Turing machines over Σ. Which of the following statements is true about the complement $\sim L$ (in Σ^*)? (**Hint:** The given condition may be vacuously true.)

(A) $\sim L$ must be $\emptyset$.
(B) $\sim L$ must be recursive.
(C) $\sim L$ may be recursively enumerable.
(D) $\sim L$ must be recursively enumerable.

Solution Consider two cases.

Case 1: $L \neq \mathscr{L}(M)$ for any Turing machine M.

In this case, the given condition is vacuously true, and L is not RE. The complement of a non-RE language may be non-RE or RE-but-not-recursive.

Case 2: $L = \mathscr{L}(M)$ for some Turing machine M.

By the given condition, M is total. Consider a family M_x of Turing machines parameterized by $x \in \Sigma^*$. On input y, M_x first checks whether $y = x$. If so, it goes to an infinite loop. Otherwise, M_x simulates M on y, and accepts/rejects as M does. By construction, no M_x is a total Turing machine. Now, if $L = \mathscr{L}(M) \neq \Sigma^*$, then $L = \mathscr{L}(M_x)$ for any $x \notin L$, a contradiction to the given condition. So we must have $L = \Sigma^*$. (Notice that this proof does not rely on the effective computability of an $x \notin L$.)

8. A language L admits a many-to-one reduction $L \leqslant_m$ HP. Which of the following statements is *false*?

(A) L may be empty.
(B) L may be a non-empty recursive language.
(C) L may be recursively enumerable but not recursive.
(D) L may be not recursively enumerable.

Solution Without loss of generality, we may assume that both M (in HP) and L are over the same input alphabet Σ.

(A) Consider the map $x \mapsto M \# \varepsilon$, where M, on any input, goes to an infinite loop.

(B) Take $L = \{\varepsilon\}$. Consider the map $x \mapsto M \# x$, where M, on input x, checks whether $x = \varepsilon$. If so, M immediately halts (after accepting or rejecting). Otherwise, M goes to an infinite loop.

(C) Take $L =$ HP, and consider the identity map.

(D) If L is non-RE, then by the reduction theorem, HP is non-RE too.

9. Consider the following reduction algorithm from HP to a language L over the alphabet $\Sigma = \{0, 1\}$. The input to the reduction algorithm is $M \# x$, and the output is a Turing machine N which, on input y, does the following.

1. Simulate M on x.
2. If the simulation halts, accept if y starts with 0 and ends with 1, reject otherwise.

Which of the following can be a possibility for L?

(A) $\{N \mid N \text{ is a total Turing machine}\}$

(B) $\{N \mid N \text{ does not accept } 10\}$

(C) $\{N \mid \mathscr{L}(N) \text{ is regular}\}$

(D) $\{N \mid \mathscr{L}(N) = \emptyset\}$

Solution (A) N halts on every input if and only if M halts on x.

For the other parts, note that

$$\mathscr{L}(N) = \begin{cases} 0\Sigma^*1 & \text{if } M \text{ halts on } x, \\ \emptyset & \text{if } M \text{ does not halt on } x. \end{cases}$$

(B) Neither $0\Sigma^*1$ nor $\emptyset$ contains the string 10.

(C) Both $0\Sigma^*1$ and $\emptyset$ are regular.

(D) The given map is a reduction from $\overline{\text{HP}}$ to the language of the option.

10. Which of the following is *not* a monotone property of RE languages over an input alphabet Σ containing at least two symbols? A property is specified by a Turing machine M, the language of which has the property.

(A) M accepts every string of length 2021.

(B) M accepts exactly one string of length 2021.

(C) M accepts at least one string of length 2021.

(D) M accepts every string of length $\neq 2021$.

Solution (B) Let A be an RE set containing exactly one string x of length 2021. Take any string $y \neq x$ of length 2021. Since $|\Sigma| \geqslant 2$, such a y can be found. But then, $B = A \cup \{y\}$ is again an RE set.