CS 420
Second Test Practice Questions Solutions

Below are some practice questions for the second test. There will be questions on the actual test that do not resemble any of these questions. Other questions on the test may ask about similar material but be phrased in a different way. Still, doing all of these practice questions will help you prepare for the test.

1. Let M_2 be the Turing machine of Figure 3.8 on page 172 of the text. For each of the following configurations of M_2, show the configuration that the given configuration yields, i.e., the next configuration after the given one. (Do not try to trace the entire computation starting from the given configuration. You are only being asked for the next configuration.)

   (a) $\sqcup q_2 00$

   Next configuration: $\underline{\sqcup x q_3 0}$

   (b) $\sqcup x 00 q_3 \sqcup$

   Next configuration: $\underline{\sqcup x 0 q_5 0 \sqcup}$

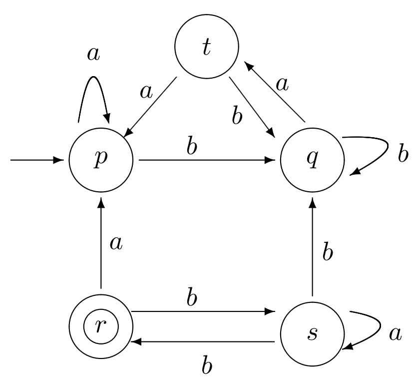
   (c) $\sqcup x q_3 x 00$

   Next configuration: $\underline{\sqcup x x q_3 00}$

   (d) $q_5 00 \sqcup$

   Next configuration: $\underline{q_5 00 \sqcup}$

2. Apply the method from class that decides E_{DFA} to the following DFA and answer the questions below.



   (a) List the states you mark in the order they get marked.

   p,q,t

   (b) Does the DFA belong to E_{DFA}? yes

(c) How does your answer to (b) follow from your answer to (a)?

No accept state is marked.

3. Apply the method from class that decides E_{CFG} to the following CFG and answer the questions below.

$$\begin{aligned} S &\to AB \\ A &\to Aa|BaB \\ B &\to bB|aXZ \\ X &\to \varepsilon|aX|Z \\ Z &\to ZaB|Ya \\ Y &\to aY|a \end{aligned}$$

(a) List the terminals and variables you mark in the order they get marked. (List each terminal and variable only the first time you mark it. There is more than one possible order.)
$a, b|Y, X|Z|B|A|S$

(b) Does the CFG belong to E_{CFG}? No

(c) How does your answer to (b) follow from your answer to (a)?
S is marked

4. Let $A = \{\langle M\rangle | M$ is a DFA and M accepts at least one string of odd length$\}$. Prove that A is decidable.

Solution:

The set of strings of odd length is a regular language. Let N be a DFA that recognizes this language.

A Turing machine U to decide the language A works as follows

$U =$ "On input $\langle M\rangle$ where M is a DFA,

1. Form a DFA C with $L(C) = L(M) \cap L(N)$, where N is the DFA described above.
2. Run the TM T that decides E_{DFA} on $\langle C\rangle$.
3. If T accepts, then *reject*. If T rejects, then *accept*. "

5. Let A be a decidable language and let B be defined by $B = \{x|$ for some string w, $xw \in A\}$. (In other words, B consists of all strings that are prefixes of strings in A.) Show that B is Turing recognizable.

Solution:

Let M be a TM that decides A. A TM N that recognizes B is given by:

$N =$ "On input x,

1. Let $s_1, s_2, \ldots$ be all the strings over Σ.

2. For $i = 1, 2, 3, \ldots$
3. Run M on xs_i.
4. If M accepts, then *accept*. If M rejects, then go to the next i."

6. Suppose that A and B are both countably infinite sets. Show that $A \cup B$ is also countably infinite.

   Solution:

   Since $A \cup B$ contains the infinite sets A and B, it is infinite. To see that $A \cup B$ is countable, let A be listed as $a_1, a_2, \ldots$ and B be listed as $b_1, b_2, \ldots$. (This is possible since A and B are countably infinite.) Then the list $a_1, b_1, a_2, b_2, a_3, b_3, \ldots$ contains all the elements of $A \cup B$. By eliminating duplicates from this list, we get a list $c_1, c_2, \ldots$ of the elements of $A \cup B$ without repetition, so $A \cup B$ is countable.

7. Use diagonalization to show that $\{f : \mathcal{N} \to \mathcal{N} | \text{for all } n,\ f(n) \text{ is a prime number}\}$ is uncountable.

   Solution:

   Suppose that the set given is countable. Since it is clerly infinite, it must have the same size as $\mathcal{N}$. Let $f_1, f_2, f_3, \ldots$ list all the elements of the set. We define a function $d : \mathcal{N} \to \mathcal{N}$ by letting for each n $d(n)$ be a prime number different from $f_n(n)$. One way to do this is to define

   $$d(n) = \begin{cases} 2 & \text{if } f_n(n) \neq 2 \\ 3 & \text{if } f_n(n) = 2 \end{cases}$$

   Then d is in the set of functions, but not on the list $f_1, f_2, \ldots$. This contradicts the fact that every function in the set is on the list. Thus, the set is uncountable.

8. Is it possible to m-reduce A_{TM} to EQ_{CFG}? Explain your answer. (You may use results in the book, on the homework, and in the homework solutions without reproving them.)

   Solution:

   It is not possible to m-reduce A_{TM} to EQ_{CFG}. To see this, suppose that $A_{TM} \leq_m EQ_{CFG}$, then $\overline{A_{TM}} \leq_m \overline{EQ_{CFG}}$. By Corollary 4.23, $\overline{A_{TM}}$ is not Turing-recognizable, so by Corollary 5.29, $\overline{EQ_{CFG}}$ is not Turing-recognizable. This contradicts Exercise 5.2.

9. Let $B = \{\langle M \rangle | M \text{ is a Turing machine and } L(M) = (01)^*\}$. Suppose that you want to show that $A_{TM} \leq_m B$ using a reduction f that maps $\langle M, w \rangle$ to $\langle M_1 \rangle$.

   (a) Fill in the blanks in the following two statements in a way that states what you have to do to make the reduction work. (In both cases you will be writing down something about the behavior of the Turing machine M_1.)

- If M accepts w, then

  $\underline{L(M_1) = (01)^*}$.

- If M does not accept w, then

  $\underline{L(M_1) \neq (01)^*.}$

(b) Give the definition of the desired Turing machine M_1, given M and w.

$M_1 =$ "On input x

1. If $x \notin (01)^*$, *reject*.
2. If $x = (01)^n$ for some n, run M on w.
3. If M accepts, *accept*. If M rejects, *reject*."

(c) What conclusion about B follows from the fact that $A_{TM} \leq_m B$. (You can get credit for this part without doing the previous two parts.)

$\underline{B \text{ is undecidable.}}$