CSc 30400 Introduction to Theory of Computer Science

1st Homework Set - Solutions

Due Date: 2/23

1 Practice Questions

P 1. The transition graph is shown in figure 1.

P 2.

	M_1	M_2
a.	q_1	q_1
b.	$\{q_2\}$	$\{q_3\}$
c.	q_1, $\{q_1, q_2\}$, $\{q_1, q_2\}$, $\{q_1, q_2\}$, $\{q_2\}$	q_1, $\{q_1, q_2, q_3\}$, $\{q_1, q_2, q_3\}$, $\{q_1, q_2, q_3\}$, $\{q_2, q_3\}$
d.	Yes	Yes
e.	No	Yes

2 Easy Questions

E 1. The transition graphs for each of the languages are shown in figures 2 - 11.

E 2. The transition graphs are shown in figures 12 - 16.

E 3. The NFA_ε, NFA, DFA and minimum DFA for this question are shown in figures 17 - 20.

E 4. The transition graph is shown in figure 21.

Since we can have up to 3 tries and in each try we should press a sequence of 4 digits, this sums up to 12 different states. We should also consider two more possible states: an accepting state (where the alarm has been deactivated) and a trap state (where the alarm sounds). We only need one trap state and one accepting state (we don't need to know neither the symbol that we missed in the last try nor the number of tries we used to deactivate the alarm).

The state $q_{ij}, i = 1, 2, 3 \wedge j = 1, 2, 3, 4$ represents the fact that we are in the i^{th} try and the alarm is expecting the j^{th} symbol in the code. We also have one accepting state which represents the fact that the alarm is deactivated and one trap state which represents the fact the the alarm sounds. The start state is the state where no buttons have been pressed at all (first try expecting the first digit).

3 Hard Questions

H 1. Suppose that a string $w_1 \dots w_n$ is accepted by a NFA $M = \{Q, \Sigma, \delta, q_0, F\}$. The states that M goes through in an accepting path are $q_0, \dots, q_f$, with $q_f \in F$. An automaton M' which accepts the reverse string $w_n \dots w_1$ should simulate the process "start from q_f, follow all the arrows backwards and accept in q_0".

For the sake of simplicity, first assume that q_f is the unique accept state of M. Create $M' = \{Q, \Sigma, \delta', q_f, \{q_0\}\}$, where $\forall q \in Q$, & $a \in \Sigma$, $\delta'(q, a) = Q_p$ such that $\forall q_p \in Q_p$, $\delta(q_p, a) = q$.

If M has many accepting states, we create a new unique accepting state q' and for each $q \in F$ we add ε-moves to q'. Now we have only one accepting state and we can create M' as shown above.

It suffices to show that M' accepts L^R, in other words to show that a string w is accepted by M' iff it belongs in L^R.

- For the first direction, consider a string $w = w_1 \dots w_n \in L^R$. The reverse $w^R = w_n \dots w_1$ belongs in L so it is accepted by M following an accepting path $q_0, \dots, q_f$, with $q_f \in F$. Thus $q', q_f, \dots, q_0$ is an accepting path under w in M', so w is accepted by M'.
- For the other direction, consider a string $x = x_1, \dots x_m$ that is an accepting string in M'. That means that there is an accepting path $q', p_1, p_2, \dots, q_0$ under x. Now obviously, the reverse path in M starts from q_0 and ends in p_1. But p_1 needs to be an accepting state in M because q' in M' is connected only with final states of M. Thus x^R belongs in L and this proves that $x \in L^R$.

H 2.

- A DFA is by definition an all-NFA: a DFA has a unique computation path for every input string and in order to accept, all the computation paths (the unique one) should end to an accept state.

- For the other direction we proceed similarly to the NFA $\rightarrow$ DFA conversion. We simulate the computation of the all-NFA by keeping track of all the states that we can be after following a symbol of the alphabet.

  So take an all-NFA $M = \{Q, \Sigma, \delta, q_0, F\}$. An equivalent DFA is $M' = \{Q', \Sigma, \delta', q_0, F'\}$, where:

  - $Q' = \mathcal{P}(Q)$
  - $\forall q = (q_1, \ldots, q_k) \in Q' \& a \in \Sigma, \delta'(q, a) = \delta(q_1, a) \cup \ldots \cup \delta(q_k, a)$
  - $F' = \{R | R \subseteq F\}$ (all states in R are accepting states of M)

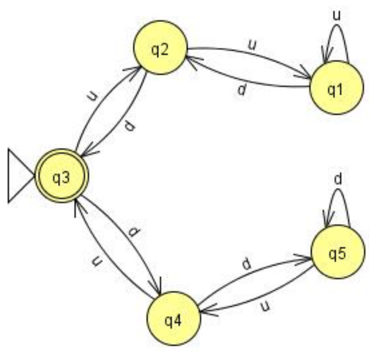

Figure 1: The transition graph for question P 1.

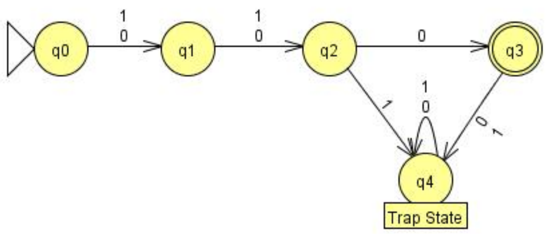

Figure 2: The transition graph for L_1 of exercise E 1.

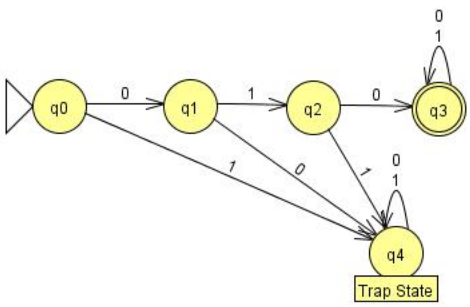

Figure 3: The transition graph for L_2 of exercise E 1.

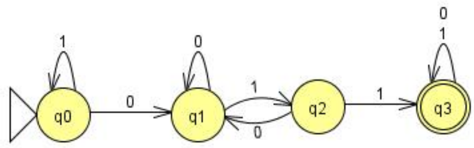

Figure 4: The transition graph for L_3 of exercise E 1.

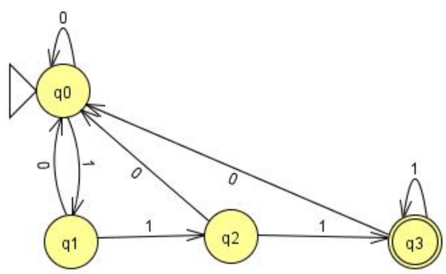

Figure 5: The transition graph for L_4 of exercise E 1.

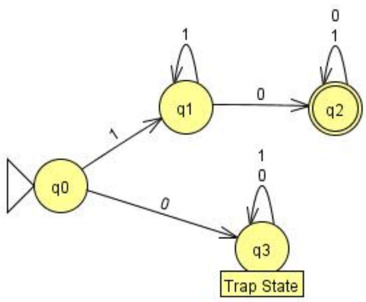

Figure 6: The transition graph for L_5 of exercise E 1.

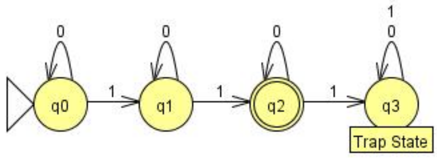

Figure 7: The transition graph for L_6 of exercise E 1.

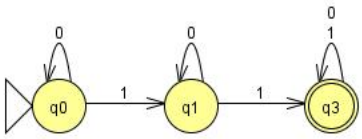

Figure 8: The transition graph for L_7 of exercise E 1.

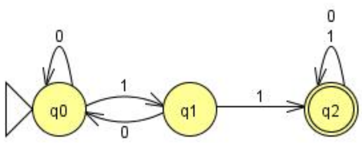

Figure 9: The transition graph for L_8 of exercise E 1.

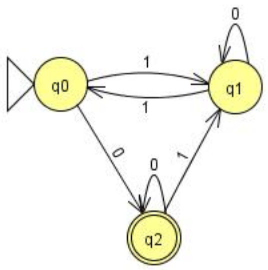

Figure 10: The transition graph for L_9 of exercise E 1.

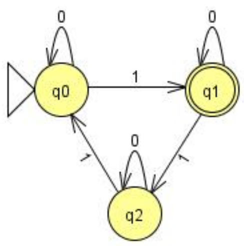

Figure 11: The transition graph for L_{10} of exercise E 1.

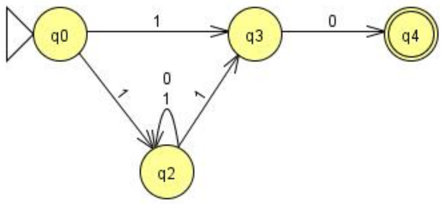

Figure 12: The transition graph for L_1 of exercise E 2.

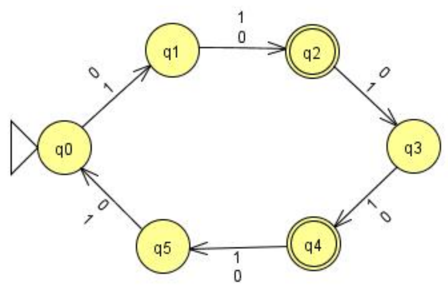

Figure 13: The transition graph for L_2 of exercise E 2.

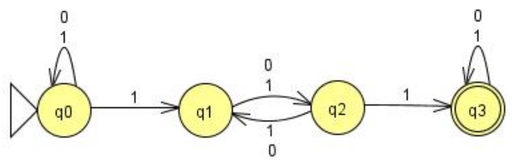

Figure 14: The transition graph for L_3 of exercise E 2.

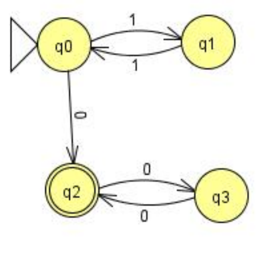

Figure 15: The transition graph for L_4 of exercise E 2.

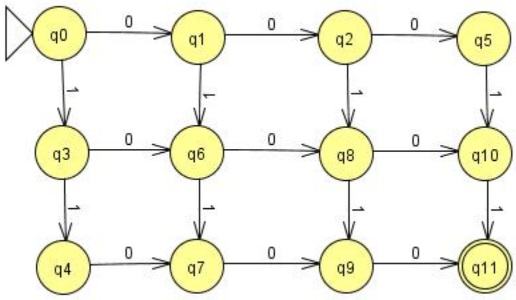

Figure 16: The transition graph for L_5 of exercise E 2.

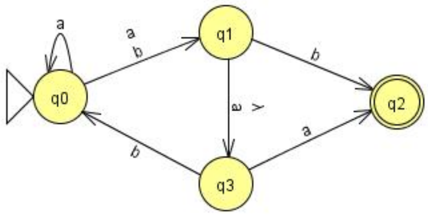

Figure 17: The NFA$_\varepsilon$ of exercise E 3.

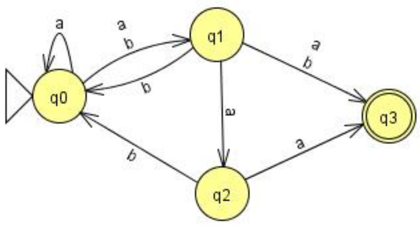

Figure 18: The NFA of exercise E 3.

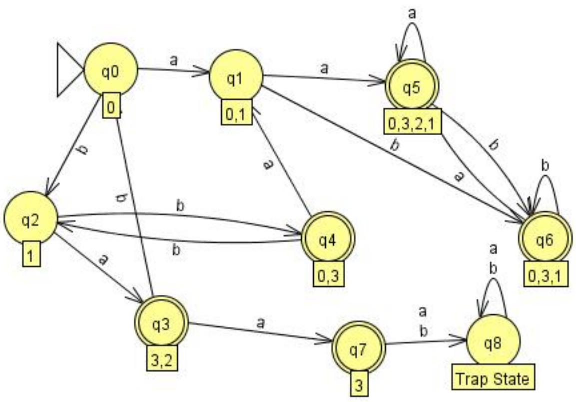

Figure 19: The DFA of exercise E 3.

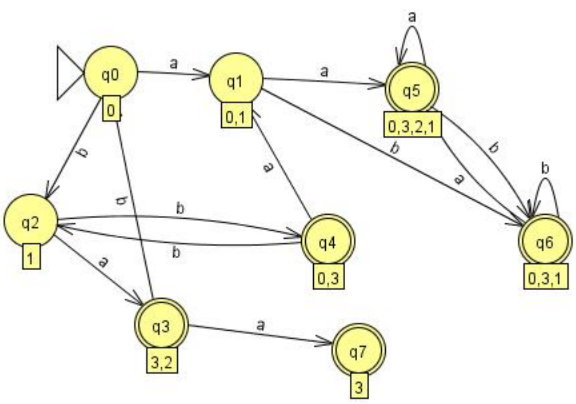

Figure 20: The minimum DFA of exercise E 3.

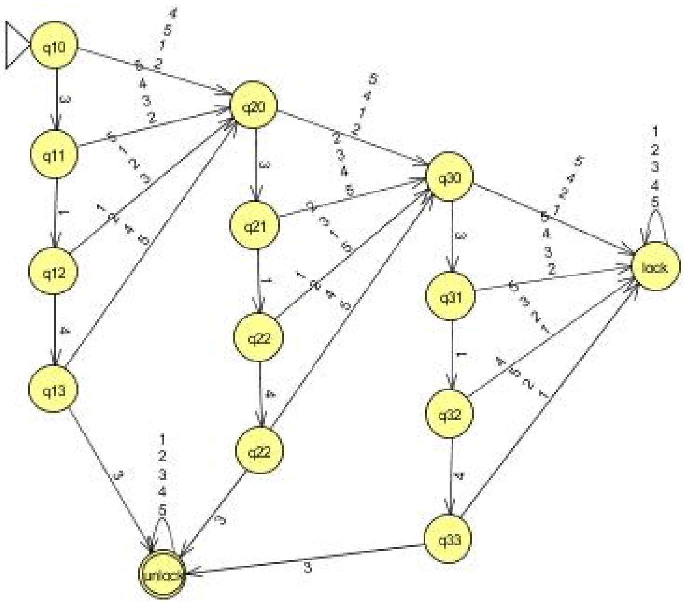

Figure 21: A DFA for the alarm of question E 4.