QUIZ 1

Course CS310: Autumn 2023

4 September, 2023

- The question paper carries 60 marks in total. It consists of multiple choice and long answer questions.
- Answers must be written in an answersheet which must be submitted with role number clearly marked.
- Answer to each question must ALSO be uploaded on SAFE.
- The quiz including uploading on SAFE must be completed in 55 minutes. Do not spend too much time on a single questions.
- Students may keep 3 printed or handwritten sheets with them. Use of books, notebooks, laptops etc. is not allowed.
- Good Luck!

1 Objective Questions

Question Q1 (5 marks) Consider the following three languages A, B, C.

$$\begin{aligned} L &= \{a^n b^m \mid n \geq m\} \\ M &= \{a^n b^m \mid n \geq m \wedge m \leq 10\} \\ N &= \{a^n b^m \mid n \geq m \wedge m \geq 10\} \end{aligned}$$

Choose the correct option.

(A) None of L, M, N are regular.

(B) M is regular but L, N are not regular **(correct)**

(C) M, N are regular but L is not regular.

(D) L, M, N are all regular.

Rubric Full marks for correct option. Zero for wrong answer.

Question Q2 (8 marks) Let $\hat{\Delta} : 2^Q \times \Sigma^* \to 2^Q$ be the extended transition function for an NFA with transition function Δ. State whether each of the following statements is true or false.

(A) If $P \subseteq R$ then $\hat{\Delta}(P, x) \subseteq \hat{\Delta}(R, x)$ **(true)**

(B) If $\hat{\Delta}(X, x) = \emptyset$ then $X = \emptyset$. **(false)**

(C) $\hat{\Delta}((P \cup R), x) = \hat{\Delta}(P, x) \cup \hat{\Delta}(R, x)$. **(true)**

(D) $\hat{\Delta}((P \cap R), x) = \hat{\Delta}(P, x) \cap \hat{\Delta}(R, x)$ **(false)**

Rubric Two marks for each correct answer.

Question Q3 (9 marks) State whether the following pair of regular expressions are equivalent (i.e. they define the same language). Answer in yes or no.

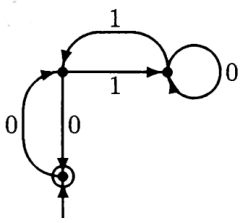
(A) $a^*(a^* + a)$ and a^*. **(yes)**

(B) $(a + b)^*$ and $(a^*b)^*$. **(no)**

(C) $(a + b)^*$ and $(a^*b^*)^*$. **(yes)**

Rubric Three marks for each correct answer.

Question Q4 (6 marks) Which one of the following regular expressions matches the following NFA?



(A) $\epsilon + 0(01^*1 + 00)^*01^*$

(B) $\epsilon + 0(10^*1 + 00)^*0$ **(correct)**

(C) $\epsilon + 0(10^*1 + 10)^*1$

(D) $\epsilon + 0(10^*1 + 10)^*10^*$

(Hint: think of eliminating the top left state in regexp construction).

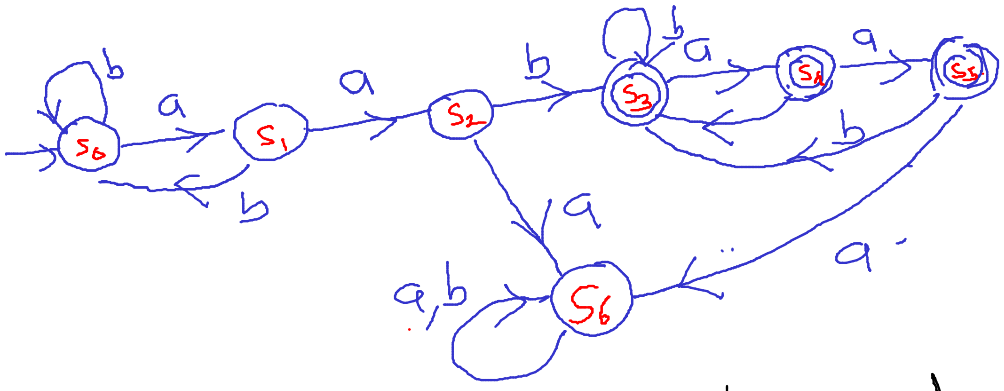
Rubric 6 Marks for correct option. 0 Marks for wrong option.
Solution: Let the states be called p, q, r with top left state as q and bottom left as p.

$$\alpha_{p,p}^{p,r} = \alpha_{p,p}^{p,r} + \alpha_{p,q}^{p,r} \cdot (\alpha_{q,q}^{p,r})^* \cdot \alpha_{q,p}^{p,r}$$

Notice that $\alpha_{p,q}^{p,r} = 0$, $\alpha_{q,p}^{p,r} = 0$ and $\alpha_{q,q}^{p,r} = 10^*1 + 00$ corresponding to two loops starting at q and ending at q. Also, $\alpha_{p,p}^{p,r} = \epsilon$. Hence the answer.

2 Questions with Long Anwers

Question Q5 (8 marks) Give a DFA over alphabet $\{a, b\}$ accepting words which have at least an occurrence of factor aab (consecutively) but no occurrence of factor aaa anywhere. Solution:



S_3 last 3 letters are aab and no aaa is encountered before.

Transition $S_2 \xrightarrow{a} S_6$ taken when aaa is encountered before any occurrence of aab

Transition $S_5 \to S_6$ taken when aaa is encountered after an occurance of aab.

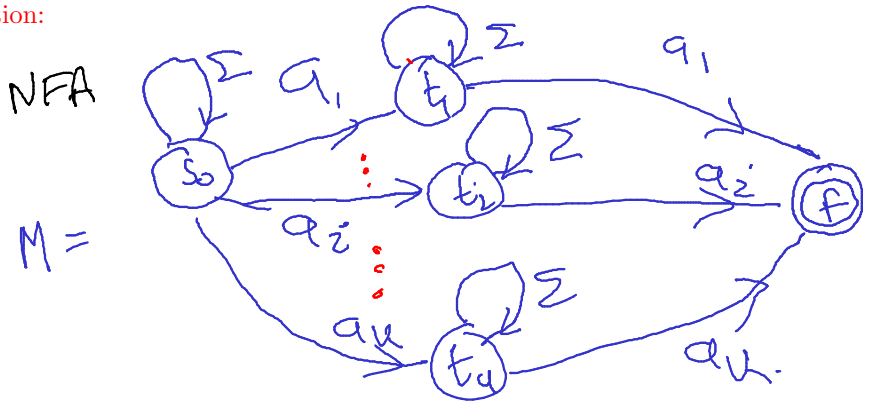
Rubric

1) 4 Marks for detecting subword aab
2) 2 Marks for rejecting if aaa occurs before aab
3) 2 Marks for rejecting if aaa occurs after aab.

Question Q6 **(8 marks)** Construct an NFA over alphabet $\Sigma = \{a_1, a_2, \ldots, a_k\}$ for the language $\{w \cdot b \mid w \in \Sigma^* \wedge b \in \Sigma \wedge b \text{ occurs at least once in } w\}$. For full marks your NFA should have size $O(k)$. Intuitively answer how many states will the **minimal DFA** for this automaton have? Give a brief justification.

Rubric 5 marks for correct DFA with $O(k)$ states. 3 marks if it is not $O(k)$. Additional 3 marks for correctly arguing that minimal DFA will have at least 2^k states.

Solution:



$|M| = k+2$

Minimal DFA on any word w must remember $\alpha(w)$ the set of letters occuring in w, in state $\hat{\delta}(s_0, w)$. Assume to contrady.

If $\hat{\delta}(s_0, u) = \hat{\delta}(s_0, v)$ and $\alpha(u) \neq \alpha(v)$

wlog let $a \in \alpha(u)$ & $a \notin \alpha(v)$.

Then $\hat{\delta}(s_0, ua) = \hat{\delta}(va)$.

Hence $ua \in L \iff va \in L$.

But this contradicts the L definiton which requires $ua \in L$ and $va \notin L$.

There are 2^k distinct $\alpha(w)$ possible.

Hence DFA must have at least 2^k states.

The DFA can be constructed with $K \times 2^K + 1$ states. Hence DFA is exponential in K.

DFA:

$Q = \{s_0\} \cup 2^{\Sigma} \times \Sigma$ $\quad \Sigma = \{a_1, \ldots, a_K\}$

- Each state is a pair (S, a) with $S \subseteq \Sigma, a \in \Sigma$
- For a word w a seen so far the automaton is in state $(\alpha(w), a)$.
- Final states: $\{(S, a) \mid a \in S\}$
- Transitions

$s_0 \xrightarrow{a_i} (\phi, a_i)$ $\quad (S, a) \xrightarrow{a_i} (S \cup \{a\}, a_i)$

DFA can be constructed in 2×2^k : states

DFA

$Q = 2^{\Sigma} \times \{0,1\}$ $\quad \Sigma = \{a_1, \ldots, a_k\}$

- For word w seen so far, automaton is in state $(\alpha(w), b)$ where bit b indicates that last letter of w occurs before in w.
- Initial state $(\phi, 0)$.
- Transitions
  Add one of following based on if

$(S, -) \xrightarrow{a_i} (S \cup \{a_i\}, 1)$ if $a_i \in S$

$(S, -) \xrightarrow{a_i} (S \cup \{a_i\}, 0)$ if $a_i \notin S$.

- $F = \{(S, 1) \mid S \subseteq \Sigma\}$

Rubric

1) 4 marks for correct NFA with $O(k)$ states

2) 2 Marks for identifying that DFA must remember $\alpha(w)$. Hence $|DFA| \geq 2^k$.

3) 1 Mark for arguing by Myhill-Nerode theorem we must also distinguish whether last letter repeats

4) 1 Mark for arguing that DFA can be constructed in $k \times 2^k$ or 2×2^k states

Hence Minimal DFA is exponential in size of k. Precise DFA construction NOT required

Question Q7 (8 marks) For the following ϵ-NFA construct DFA using the subset construction, retainining only the reachable states. Show steps of construction. Clearly label each state of DFA with the subset of ϵ-NFA states it represents. What language does the constructed automaton recognize (give the answer in regular expression form)? **(ans:** $(a+b+\epsilon)b^*$**)**.

	ϵ	a	b
$\rightarrow$ p	$\{q\}$	$\{r\}$	$\emptyset$
q F	$\emptyset$	$\emptyset$	$\{r\}$
r	$\{q\}$	$\emptyset$	$\emptyset$

Solution

	a	b
$\rightarrow$ $\{p,q\}$	$\{r,q\}$	$\{r,q\}$
F $\{r,q\}$	ϕ	$\{r,q\}$
F ϕ	ϕ	ϕ

$C_\epsilon(\{p\}) = \{p,q\}$

$\hat{\delta}(\{p,q\},a) = \{r\}$

$\hat{\delta}(\{p,q\},b) = \{r\}$

$\hat{\delta}(\{r,q\},a) = \phi$

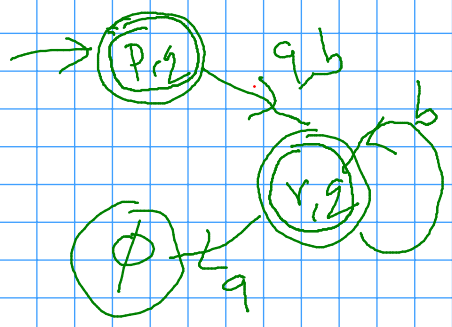
$\hat{\delta}(\{r,q\},b) = \{r\}$

$C_\epsilon\{r\} = \{r,q\}$

...

$C_\epsilon(\phi) = \phi$

Minimal DFA



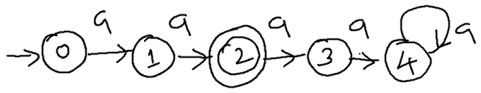
Regular Expression

$L = (\epsilon + q)b^*$

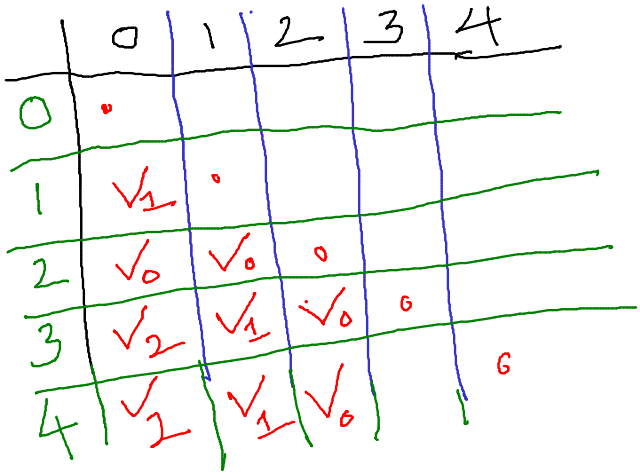
Rubric

1) 2 mark for correct initial state
2) 1 mark for correct set of final states
3) 3 marks for correct transitions.
4) 1 mark for including transitions for state marked ϕ
5) 1 mark for only expanding reachable subsets.

Question Q8 (8 marks) Consider the DFA M_1 below.



Using the Hopcroft partition refinement algorithm for DFA minimization, determine which pairs of states are not equivalent. (Use the table data-structure introduced in class/Kozen Book). Clearly show in the table the phase in which pairs get separated. Draw the minimized DFA.



Phase 0: 2 is separated from all other states

Phase 1:

(0,1) $\xrightarrow{a}$ (1,2) Mark.
(0,3) $\xrightarrow{a}$ (1,4) Unmark
(0,4) $\xrightarrow{a}$ (1,4) Unmark
(1,3) $\xrightarrow{a}$ (2,4) Mark.
(1,4) $\xrightarrow{a}$ (2,4) Mark
(3,4) $\xrightarrow{a}$ (4,4) Unmark

Phase 2

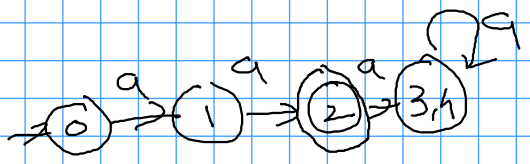
(0,3) $\xrightarrow{a}$ (1,4) Mark
(0,4) $\xrightarrow{a}$ (1,4) Mark
(3,4) $\xrightarrow{a}$ (4,4) Unmark

phase 3

(3,4) $\xrightarrow{a}$ (4,4) Unmark
No change

$\Pi_2 = \{\{0\}, \{1\}, \{2\}, \{3,4\}\}$

Equiv. Partition states.



Minimal DFA.

Rubric

1) 1 mark for correct table format
2) 2 marks for correct phase 0
3) 2 marks for correct phases 1, 2
4) 1 mark for identify termination in phase 3
5) 2 Mark for drawing Correct DFA