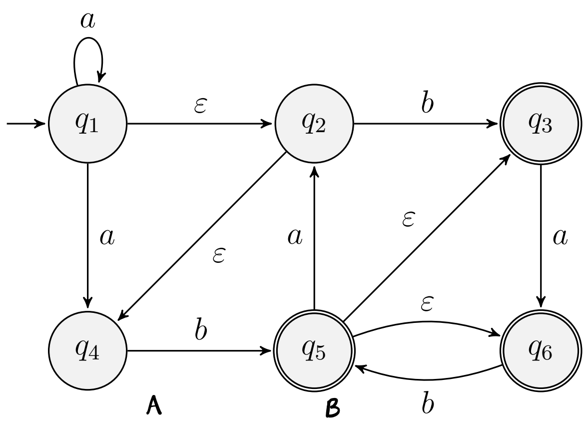
CSCI 2210: Theory of Computation

Problem Set 3 (due 09/30)

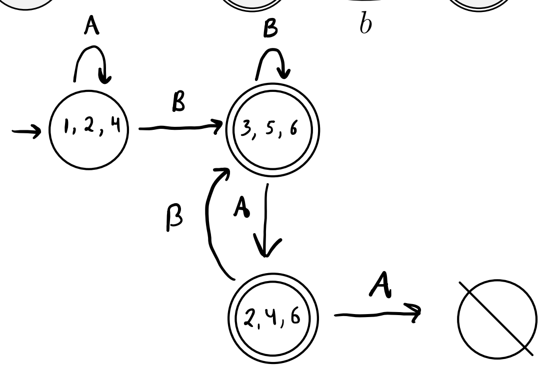
Student: *Collaborators:*

Problem 1. Use the subset construction to convert the NFA below to an equivalent DFA.



E

	A	B	ε
1	1,4	∅	1,2,4
2	∅	3	2,4
3	6	∅	3
4	∅	5	4
5	2	∅	5,3,6
6	∅	5	6



Problem 2. List all strings of lengths 3 and 4 that do NOT belong to the language $L(a^*b^*a^*)$.

ABAB

BABA

BBAB

BABB

BAAB

BAB

Problem 3. Give regular expressions generating the following languages. Consider the alphabet $\Sigma = \{a, b\}$.

(a) $L_a = \{w \mid w \text{ contains at least two } b\text{'s}\}$

$$(a+b)^*b(a+b)^*b(a+b)^*$$

(b) $L_b = \{w \mid w \text{ has length at least 3 and its third symbol is } a\}$

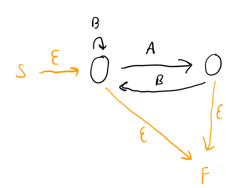
$$(a+b)(a+b)a(a+b)^*$$

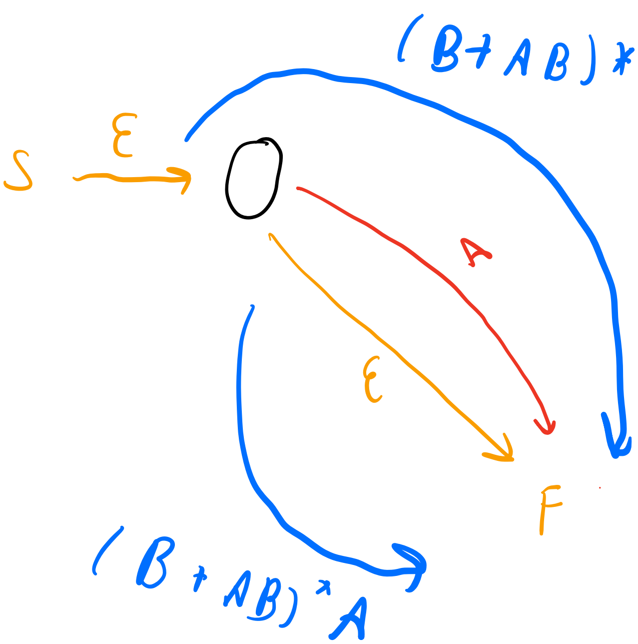
(c) $L_c = \{w \mid w \text{ starts with } a \text{ and has odd length, or starts with a } b \text{ and has even length}\}$

$$(a((a+b)(a+b))^*) + (b(a+b)((a+b)(a+b))^*)$$

(d) $L_d = \{w \mid w \text{ does not contain two consecutive } a\text{'s}\}$

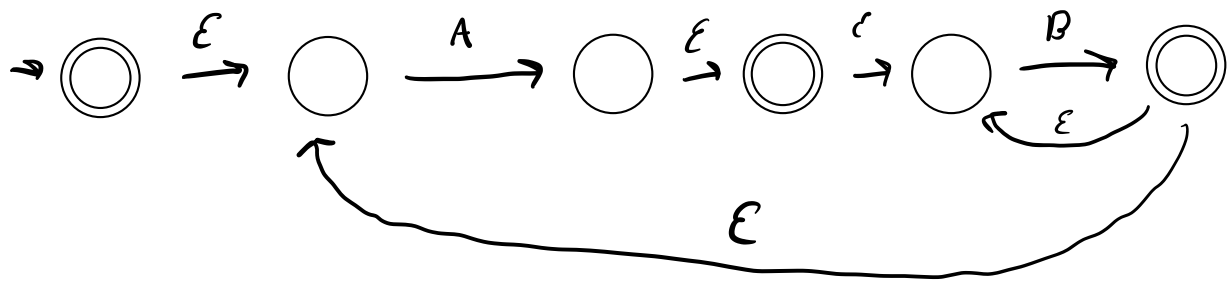
$$\varepsilon + (B+AB)^*(A+\varepsilon)$$



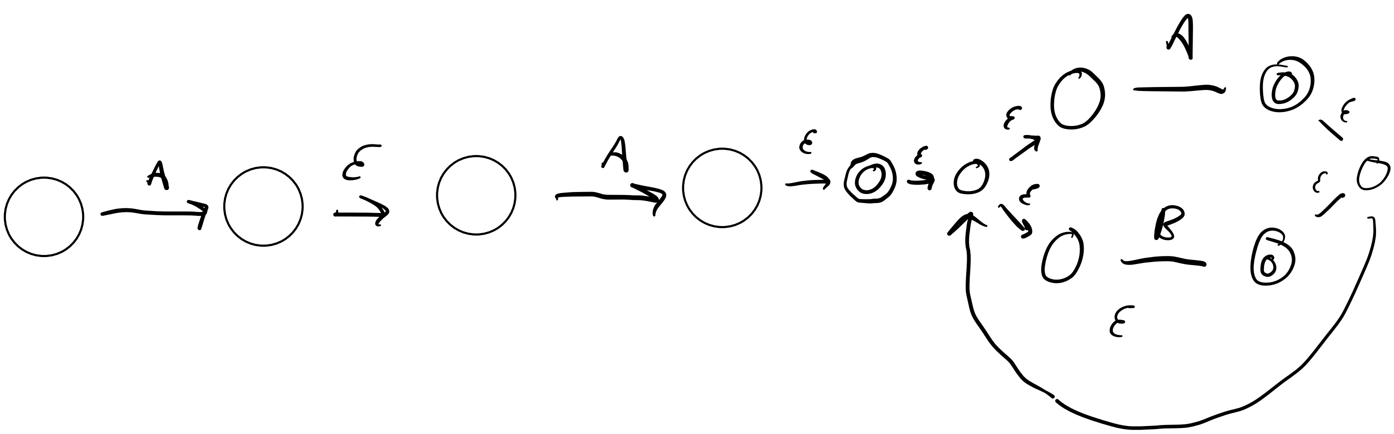


Problem 4. For each regular expressions below give an NFA or DFA that recognizes the same language.

(a) $(ab^*)^*$

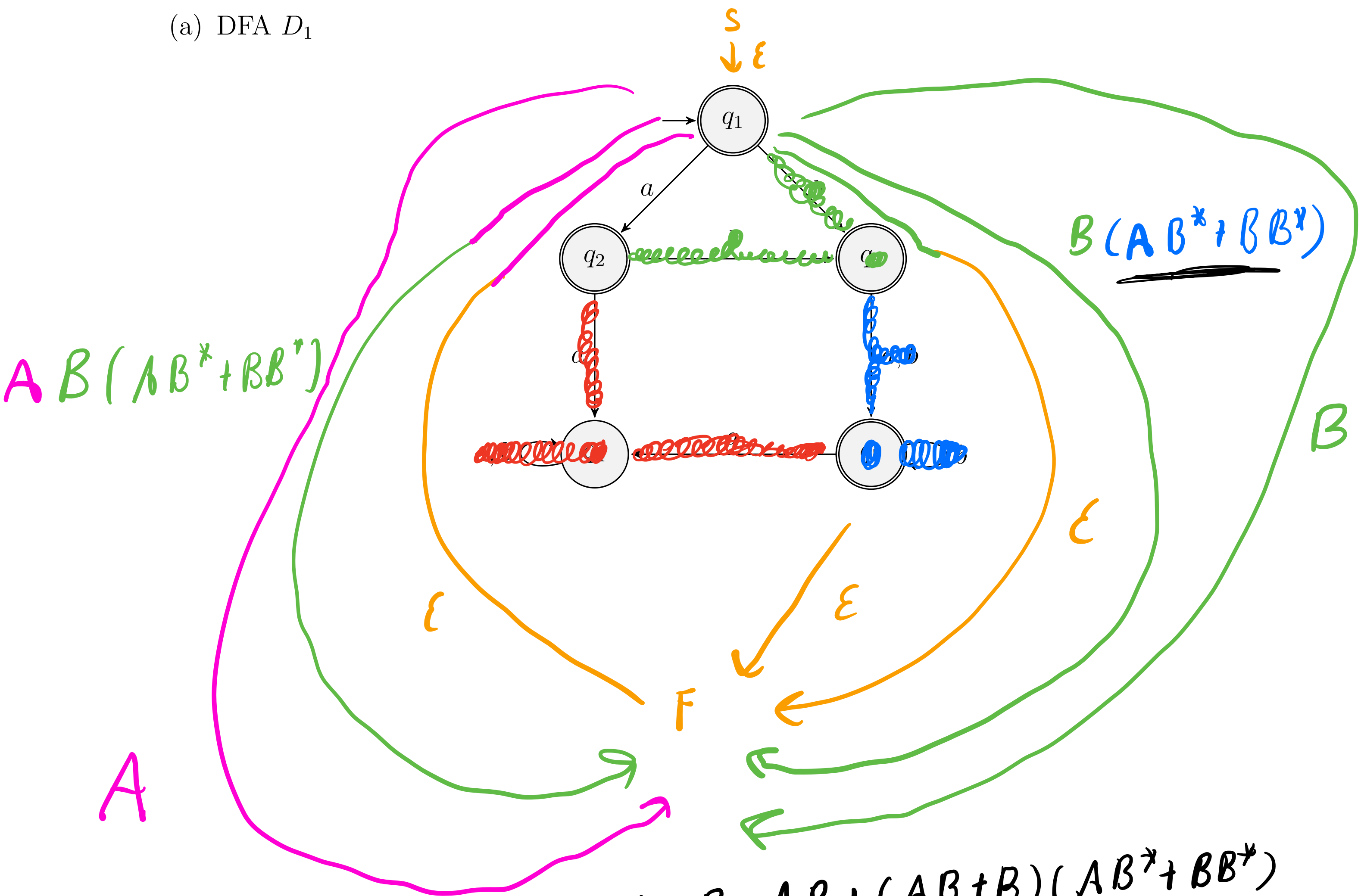


(b) $aa(a \cup b)^*$



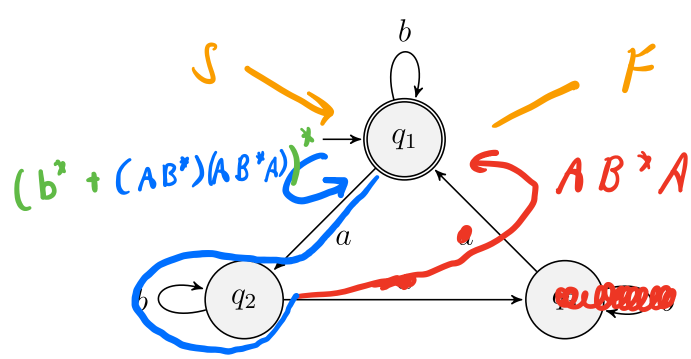
Problem 5. Observe the type of strings that are accepted by the following DFAs and state the regular expression for the language it recognizes.

(a) DFA D_1



ε + A + B + AB + (AB + B)(AB* + BB*)

(b) DFA D_2



(B* + (AB*)(AB*A))*

Problem 6. Recall that we have seen some operations with languages, like complement, union, intersection, and how regular languages are closed under them. For this problem we are going to go over the same process for a newly defined operation.

Consider the operation $\mathsf{twice}(L)$ defined for languages over some alphabet Σ as follows:

$$\mathsf{twice}(L) = \{w \mid w = x_1x_1x_2x_2 \ldots x_kx_k \text{ for some } x_1x_2 \ldots x_k \in L \text{ where } x_i \in \Sigma \text{ and } i > 0\}$$

In other words, strings in $\mathsf{twice}(L)$ are obtained by doubling each symbol from a string coming from L. For example, if $aaba \in L$, then $aaaabbaa \in \mathsf{twice}(L)$. Note that ε can never be a part of $\mathsf{twice}(L)$ as it doesn't have any symbols.

Consider $\Sigma = \{a, b\}$ and answer the following questions.

(a) Describe the language $\mathsf{twice}(L_{odd})$ where $L_{odd} = \{w \mid w \in \Sigma^* \text{ and } |w| \text{ is odd}\}$.

Strings in `twice($L_{odd}$)` are twice the length (always even) and the characters of `twice($L_{odd}$)` always come in pairs of the same

(b) Show that regular languages are closed under the operation twice.

$M_1 = (Q_1, \Sigma, \delta_1, q_{0_1}, F_1)$ $\quad L(M_1) = L_1$

$M_2 = (Q_1,$ $\quad L(M_2) = \text{twice}(L_1)$

$\Sigma,$

$\delta(q, aa) = \delta_1(q, a),$

$q_{0_1},$

$F_1)$

The intuition here comes from the insight above that the all characters of any string that passes through the twice operation come in pairs of the same. Therefore, to show that regular languages are closed under the operation twice, construct an NFA from the original where the only change needed is that all transitions require a pair of the same characters.

Given that the `twice` operation doubles the length and introduces the pairs of same characters, M2 simply needs to consume the same characters but doubled in every transition.