

1. Prove that there is one unique partition of states into equivalence classes in DFA, where in each class A , all states p in A are equivalent, but across any two classes A and B , any state p from A is not equivalent from any state q from B .

The uniqueness is basically saying the order of splitting does not matter.

Show that this immediately implies that the new DFA according to the partition is the smallest possible.

Solution:

(a)

Assume there are two different partitions for a DFA with more than 1 state. Then, there must exist two states p, q , they are in the same class in one partition [1], and in different classes in the other partition [2].

By [1], p and q are equivalent.

By [2], p and q are not equivalent.

Thus, the assumption does not hold.

(b)

On the one hand, assume the partition has k classes (C_1, C_2, \dots, C_k) ,

then we have $q_i \in C_i, i = 1..k$,

s.t. q_i and q_j are not equivalent, $i, j = 1..k, i \neq j$.

these k states in the original DFA cannot be merged, thus

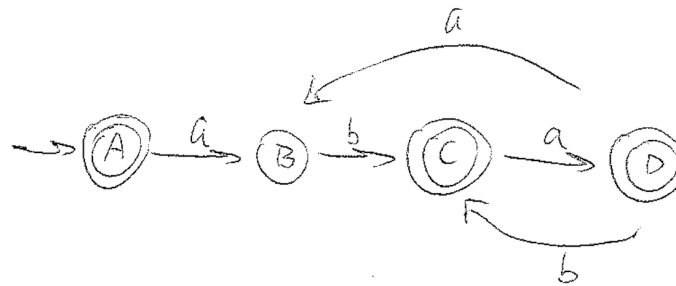
a DFA M is equivalent of the original DFA $\Rightarrow |Q_M| \geq k$.

On the other hand, a DFA can be constructed according to the partition which has exactly k states.

2. Now revisit problem 1 of HW 4. Which DFAs converted from NFAs can be minimized?

Solution:

(a) This DFA can not be minimized.

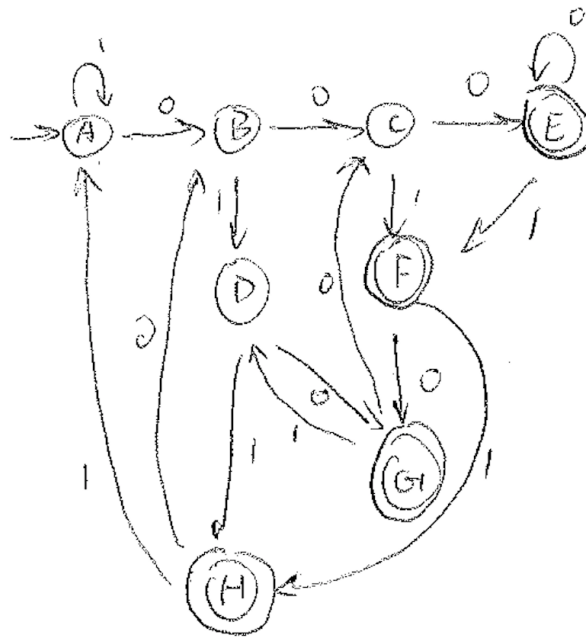


We start with two equivalent classes $F = \{A, C, D\}, Q - F = \{B\}$.
Consider states in F ,

	a	b
A	B	trap
C	D	trap
D	B	C

The three states in F should be split to three separate states.

(b) The DFA can not be minimized.



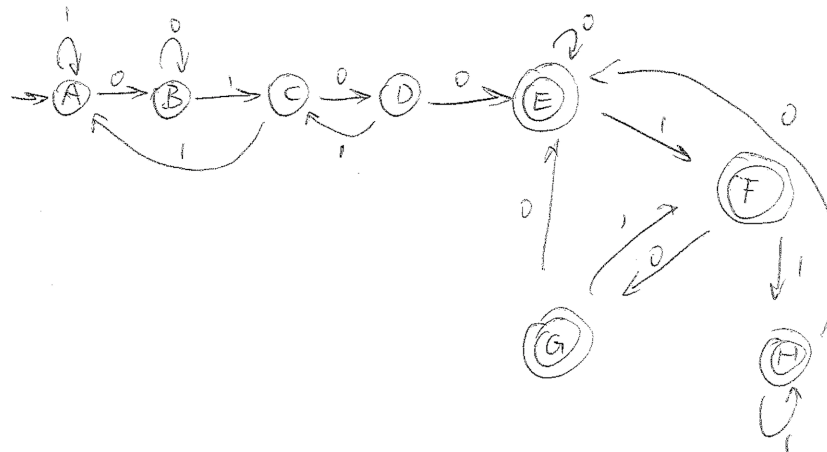
We start with equivalent classes $F = \{E, F, G, H\}$, $Q - F = \{A, B, C, D\}$.
 First consider the states in F :

	0	1
E	E	F
F	G	H
G	trap	D
H	B	A

The four states in F should be split to four separate states.
 Then consider the states in $Q - F$:

	0	1
A	B	A
B	C	D
C	E	F
D	G	H

The four states in $Q - F$ should be split to four separate states.
 (c) The DFA can be minimized.



We start with equivalent classes $F = \{E, F, G, H\}$ and $Q - F = \{A, B, C, D\}$.
First consider the states in F :

	0	1
E	E	F
F	G	H
G	E	F
H	E	H

The four states in F can be merged to one state.
The consider the states in $Q - F$:

	0	1
A	B	A
B	B	C
C	D	A
D	E	C

The four states in $Q - F$ should be split to four separate states.

3. Write the regular expressions for bitstrings; if impossible, explain.

(a) same number of 0s and 1s

Solution: Impossible.

Consider $w = 0^p 1^p$ which has the same number of 0s and 1s, and p is the pumping length.
Decompose $w = xyz$ where $|y| > 0$ and $|xy| \leq p$.

Since $|xy| \leq p$, xy only contains 0. Also since $|y| > 0$, y should be at least one 0. xy^2z does not have the same number of 0s and 1s, which contradicts the pumping lemma.

(b) all 0s are before all 1s

Solution: 0^*1^*

(c) contains at least two disjoint occurrences of 010

Solution: $\Sigma^*010\Sigma^+010\Sigma^*$

(d) every 1 must be followed by at least two 0s

Solution: $(0^*1000^*)^*$

(e) palindrome

Solution: Impossible.

Consider string $w = 0^p 10^p$ which is a palindrome and p is the pumping length. Decompose $w = xyz$ where $|y| > 0$ and $|xy| \leq p$. If the language of palindrome is regular, then from pumping lemma we have $xy^i z$ is also palindrome.

However, since $|xy| \leq p$, xy only contains 0. Also since $|y| > 0$, y should be at least one 0. But xy^2z can not be palindrome since we can not simply add more 0s in the first part of 0s without adding 0s in the second part, which contradicts.

(f) starts and ends with different letters

Solution: $(0\Sigma^*1) \cup (1\Sigma^*0)$

(g) divisible by 3

Solution: $(0 \cup 1(01^*0)^*1)^*$

(h) odd number of 0s

Solution: $1^*01^*(01^*01^*)^*$

(i) odd number of 0s and even number of 1s

Solution: $((11 \cup 10(00)^*01) \cup (0 \cup 10(00)^*1)(1(00)^*1)^*(0 \cup 1(00)^*01))^*(0 \cup 10(00)^*1)(1(00)^*1)^*$
<http://home.hiwaay.net/~gbacon/perl/nfa-001.pdf>

(j) the difference between the numbers of 0s and 1s is even

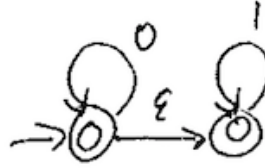
Solution: $(\Sigma\Sigma)^*$, i.e., when the length of the bitstring is even.

4. Convert the above REs to NFAs.

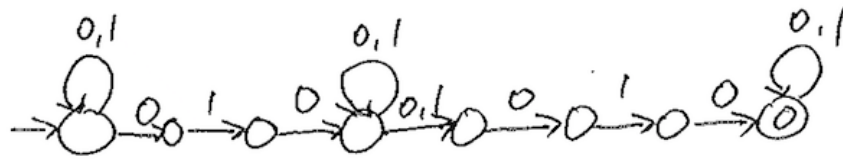
Solution:

(a) N/A.

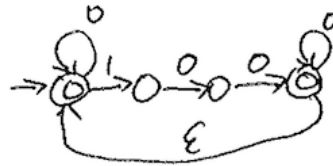
(b) NFA:



(c) NFA:

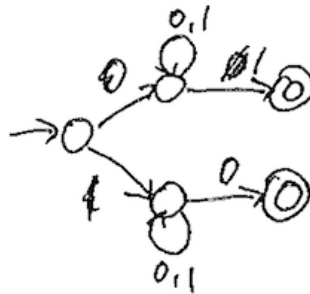


(d) NFA:

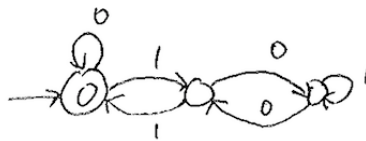


(e) N/A.

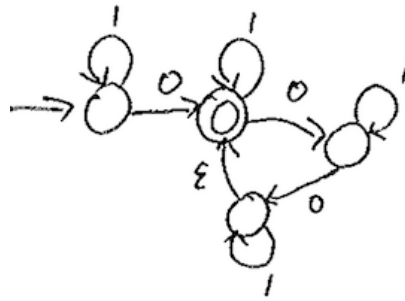
(f) NFA:



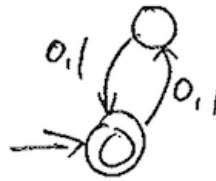
(g) NFA:



(h) NFA:



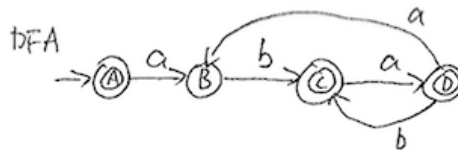
- (i) skip. too complicated.
- (j) NFA:



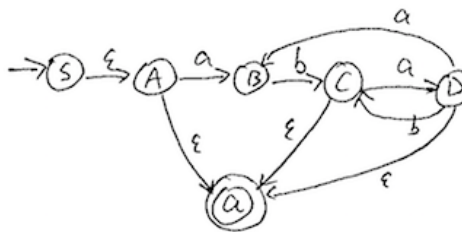
5. Convert the minimal DFAs from problem 1 to GNFA's and then REs.

Solution:

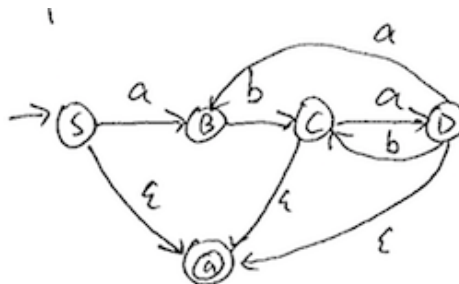
- (a) The initial DFA:



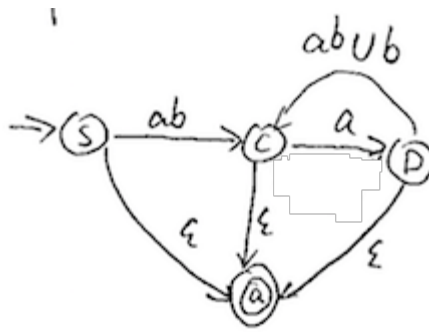
Step 1: add start state and accepting state:



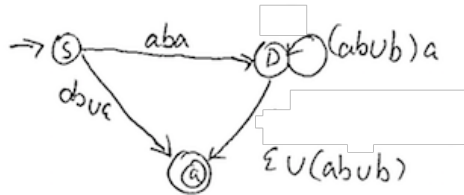
Step 2: remove state A:



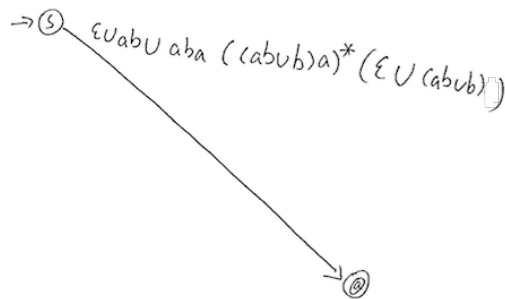
Step 3: remove state B:



Step 4: remove state C:



Step 5: remove state D:

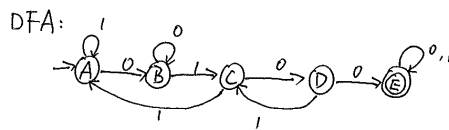


So the final RE is $ab \cup \epsilon \cup aba(((aba \cup ba)^* (\epsilon \cup (ab \cup b))))$

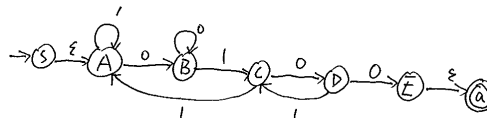
(b) skip. too complicated.

(c) We solve a simplified version of the problem here where the four accepting states are merged into one state E.

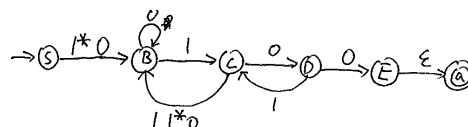
The simplified DFA:



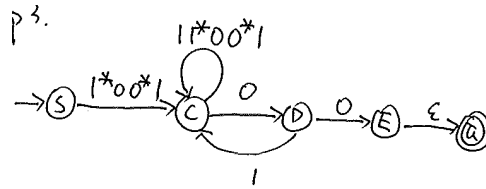
Step 1: add the new start and accepting state:



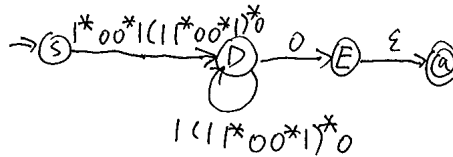
Step 2: remove state A:



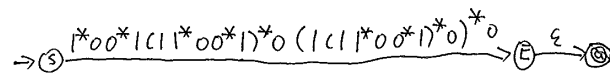
Step 3: remove state B :



Step 4: remove state C :



Step 5: remove state D :



We skip the last step of removing E and get the RE: $1^*00^*1(11^*00^*1)^*0(1(11^*00^*1)^*0)^*0$

6. Convert the NFAs from problem 4 back to REs. Do you get the same REs?

Solution:

- (a) N/A.
- (b) Same: 0^*1^*
- (c) Same: $\Sigma^*010\Sigma^+010\Sigma^*$
- (d) Not necessarily the same. Depends on the order of removing states.
- (e) N/A.
- (f) Same: $(0\Sigma^*1) \cup (1\Sigma^*0)$
- (g) Not the same: $(0 \cup (1(01^*0)^*1))^*$ (removing states from right to left.)
- (h) Not the same: $1^*0(01^*01^* \cup 1)^*$
- (i) N/A.
- (j) Same: $(\Sigma\Sigma)^*$