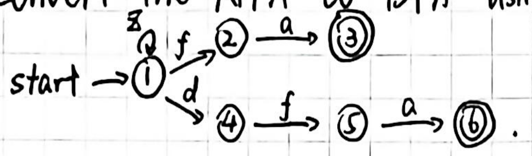
Week 5. By Deng Yufan.

Prob 1. Convert the NFA to DFA using lazy evaluation, where $\Sigma = \{a, b, \cdots z\}$.



As basis, we have $\{1\} \in Q_D$.

Since $\delta_D(\{1\}, f) = \{1,2\}$, $\delta_D(\{1\}, d) = \{1,4\}$, $\delta_D(\{1\}, w) = \{1\}$ for $w \in \Sigma \setminus \{f, d\}$,

we have $\{1,2\}, \{1,4\} \in Q_D$.

Since $\delta_D(\{1,2\}, a) = \{1,3\}$, and ~~else~~ other transition for $\{1,2\}$ are "old".

we have $\{1,3\} \in Q_D$.

Since $\delta_D(\{1,4\}, f) = \{1,2,5\}$, and other transition for $\{1,4\}$ are old, we have $\{1,2,5\} \in Q_D$.

Since, all transition for $\{1,3\}$ are old, we have nothing to add.

Since $\delta_D(\{1,2,5\}, a) = \{1,3,6\}$, and other transition for $\{1,2,5\}$ are old, we have $\{1,3,6\} \in Q_D$

Since all transition for $\{1,3,6\}$ are old, we have nothing to add.

So $Q_D = \{\overset{A=}{\{1\}}, \overset{B=}{\{1,2\}}, \overset{C=}{\{1,4\}}, \overset{D=}{\{1,3\}}, \overset{E=}{\{1,2,5\}}, \overset{F=}{\{1,3,6\}}\}$.

And we can construct

	a	d	f	$\Sigma \setminus \{a,d,f\}$
$\rightarrow$A	A	C	B	A
B	D	C	B	A
C	A	C	E	A
*D	A	C	B	A
E	F	C	E	A
*F	A	C	B	A

扫描全能王 创建

Prob 2. Start with $L_0 = \{0^n 1^n \mid n \geq 0\}$ is not a regular language, prove following language are not regular using closure properties.

a. $L = \{0^i 1^j \mid i \neq j\}$.

~~Let $\Sigma = \{0,1\}$~~. We have ~~$C_L = \{0^i$~~ $\boxed{A = \{0^i 1^j\} = L(0^* 1^*)}$ is regular.

And $C_L \cap A = \{0^i 1^j \mid i = j\} = \{0^n 1^n \mid n \geq 0\} = L_0$ is not regular.

So L is not regular.

b. $L = \{0^n 1^m 2^{n-m} \mid n \geq m \geq 0\}$.

We have $L \cap A = \{0^n 1^m \mid n - m = 0\} = \{0^n 1^n \mid n \geq 0\} = L_0$ is not regular.

So L is not regular.

Prob 3. Give an algorithm to tell whether a RL contains at least 100 strings.

For RE:

basis: $\emptyset$ has 0 string. ε has 1 string. a has 1 string.

induction: ~~If E has finite n strings, F has finite m strings.~~

E+F has n+m strings. (when E has n strings, F has m strings).

has infinite strings. (when E or F has infinite strings).

~~EF has nm strings.~~

~~E* has infinite strings.~~

EF has 0 string. (when E or F has 0 string)

has nm strings. (when E has n>0 strings.
F has m>0 strings).

has infinite strings. (when E or F has infinite strings.
and neither of them has 0 string).

E* has infinite strings. (when E doesn't has 0 string)

has 0 string.

Prob 3. Give an algorithm to tell whether a RL contains at least 100 strings.

Step 1. Represent the RL in NFA.

Step 2. For all state $q \in Q_N$, determine which of the states can be reached by q.

basis: q can be reached by q.

induction:

if q' can be reached by q, then every state in $\delta(q', a)$ can be reached by q, where a is a single char.

~~Step 3. Check for all transitions. If $q \in \delta(p, a)$, and p can be reached by q and q_0, and $\exists q' \in F$, q' can be reached by p~~

Step 3. Delete all states which can't be reached by q_0, or can't reach any of accepting states.

Step 4. Check for all transitions. If $q \in \delta(p, a)$, and p can be reached by q, then there is a reachable ring in NFA and the RL contains infinite strings.

Step 5. Otherwise, there is no ring in NFA. Convert the NFA to DFA, so there is no ring in DFA. (Easily proved using topo order).

Step 6. For all state $q \in Q_D$, calculate $f(q)$ representing ways of reaching q from q_0.

basis: $f(q_0) = 1$, and is finished; $f(q) = 0$, and is not finished for $q \neq q_0$.

induction: if p is finished, q is not finished, $\delta(p, a) = q$,

then $f(q) \leftarrow f(q) + f(p)$. every transition are considered exactly once.

if all transition towards p has been considered, p becomes finished.

end: all states are finished. since the DFA is DAG, the algorithm will end.

Step 7. Determine whether $\sum_{q \in F} f(q)$ is less than 100 or not.

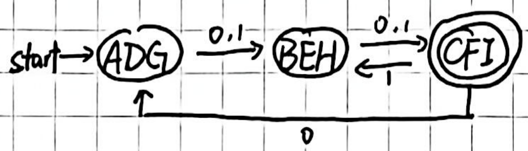
Prob 4. Consider the automaton:

	0	1
→A	B	E
B	C	F
*C	D	H
D	E	H
E	F	I
*F	G	B
G	H	B
H	I	C
*I	A	E

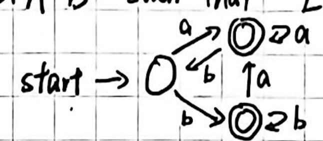
a. Draw the table of distinguishable pairs.

	A	B	C	D	E	F	G	H
B	x							
C	x	x						
D	·	x	x					
E	x	·	x	x				
F	x	x		x	x			
G	·	x	x		x	x		
H	x		x	x	·	x	x	
I	x	x		x	x		x	x

b. Construct the minimum-state equivalent DFA.



Prob 5. Let $h: \{0,1\} \to \{a,b\}^*$ that $h(0)=ab$, $h(1)=ba$. Consider DFA A as following, construct DFA B over $\{0,1\}$ such that $L(B) = h^{-1}(L(A))$.

A.


We have

B.