Homework 1 - 5

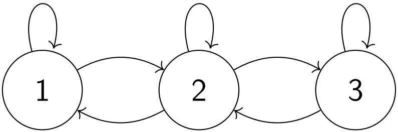
曾守瑜

Menu

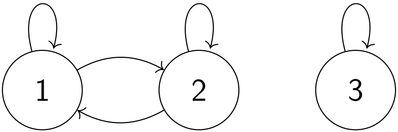
Hw1 Problem1

用有向圖來表達符合特定條件的 relation

Reflexive ∘ Symmetric ∘ Transitive ×



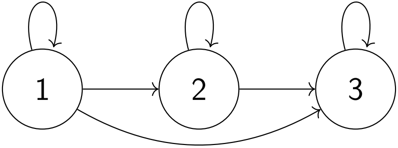
以下是錯誤範例



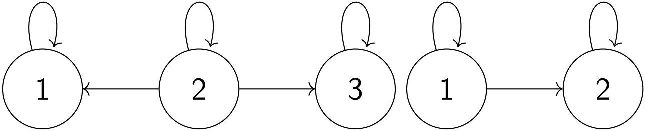
第二個 relation 並沒有違反 Transitive

Hw1 Problem1

Reflexive $\circ$ Symmetric $\times$ Transitive $\circ$

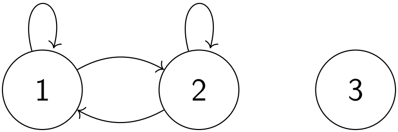


這張圖要注意箭頭方向，不要畫到最後變成大迴圈。
其他畫法：

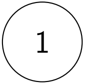


Hw1 Problem1

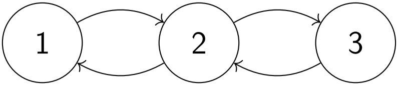
Reflexive × Symmetric ○ Transitive ○



最省事的畫法：



以下是錯誤範例



只把自迴圈拔掉，會違反 Transitive (1 到 2，2 又到 1，那 1 應該有自迴圈)

Hw1 Problem2

給定一個 relation，判斷是不是 equivalence relation
是則證，不是則給反例

第一小題題目沒寫清楚，題目的意圖是要把公因數為 1（互質）排除在外
也就是 4R6, 6R9, 而 4R9 不成立

第二小題是對的：
Reflexive：很明顯自己除以某數的餘數一定一樣
Symmetric：a 與 b 有相同餘數，b 與 a 當然也會有相同餘數
Transitive：a 與 b 有相同餘數，b 與 c 也有，則 a b c 都能寫成 qX+r 的形式，看得出 a 與 c 也會有相同餘數

Hw1 Problem3

重新定義 relation 並以此基礎延伸定義 function，使 function 為 relation 的特例，並且定義要涵蓋 arity 與 f(a)=b 的表示法

從頭定義可以很自由，但題目的要求全數必須達成
你可以將 relation 定義成一個 input-output 的關係，且 output 為布林值
也可以將 relation 定義成集合，裡頭包含許多的 tuples
你可以把 n-ary function 定義成一種 n+1-ary relation，也可以定義成一種 binary relation，只要寫得合理就有分

Hw1 Problem3

一個 relation R 是若干集合的 Cartesian product 的子集合
若 $R \subseteq A_1 \times ... \times A_k$，則 R 為 k-ary relation
若 $A_1 = ... = A_k = A$，則稱 R 是 (k-ary) relation on A
2-ary relation 也被稱為 binary relation

一個 function f 是滿足下列條件的一種 binary relation
$f \subseteq (A_1 \times ... \times A_k) \times B$，換句話說，在這個 relation 當中，每個 tuple 的第一項元素 t 也是一個 k-ary tuple，所有 t 都屬於同一群集合的 Cartesian product
且對於所有 t，如果 (t, a) 與 (t, b) 都屬於 function f，則 a=b
當 t 是 k-ary tuple，我們稱此 function 是 k-ary function
我們使用 f(a) = b 代表 ((a), b) 屬於 function(binary relation) f 當中
$f(a_1, ..., a_n) = b$ 則代表 $((a_1, ..., a_n), b) \in f$

Hw1 Problem4

對一個 simple finite graph（沒有自迴圈）而言，是否總會有兩個點他們的 degree 相同？
因為兩個點之間不會有重複的 edge，也沒有自迴圈，所以一個點頂多被 n-1 個 edge 連到，degree 有 0 到 n-1 這 n 種可能
simple graph 沒有限制 graph 本身是 connected 與否，無法直接排除 degree 為 0 的可能，所以無法直接用鴿籠原理

n 個點要分 n 種可能不重複，就只能是 0 到 n-1 各自分到一個點
被分到 n-1 的點，因為這個點沒有自迴圈，所以必然被其他的 n-1 個點連到了
但這 n-1 個點包含了 degree 為 0 的點，矛盾發生了，完美。

Hw1 Problem5

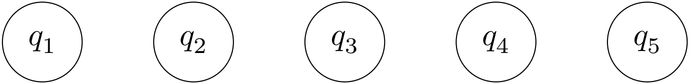
世界上的馬都變成白馬了，你能找出原因嗎？
若是三匹馬以上，H_1 與 H_2 的交集是有別的馬的
藉由 Inductive Hypothesis，這些別的馬與被移除的第一匹馬同色，
也和被移除的第二匹馬同色，所以這群馬同色
但是
在兩匹馬的時候，H_1 與 H_2 的交集是空的
從而無法真正確定兩匹馬顏色一樣
既然二就爆掉了，那麼三以上的證明也就爆了

Hw2 Problem1

$\{q_1, q_2, q_3, q_4, q_5\}$
$\{a, b\}$
q_1
$\{q_5\}$

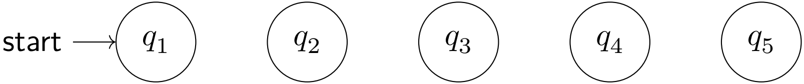
Hw2 Problem1

$\{q_1, q_2, q_3, q_4, q_5\}$
$\{a, b\}$
q_1
$\{q_5\}$



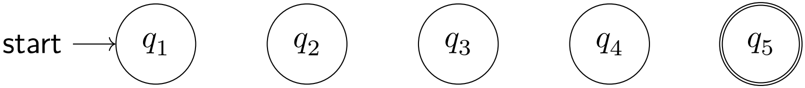
Hw2 Problem1

$\{q_1, q_2, q_3, q_4, q_5\}$
$\{a, b\}$
q_1
$\{q_5\}$



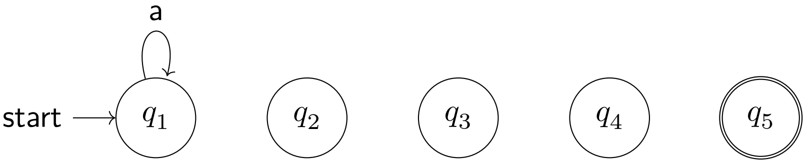
Hw2 Problem1

$\{q_1, q_2, q_3, q_4, q_5\}$
$\{a, b\}$
q_1
$\{q_5\}$



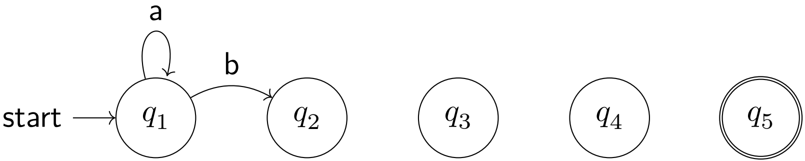
Hw2 Problem1

$\{q_1, q_2, q_3, q_4, q_5\}$
$\{a, b\}$
q_1
$\{q_5\}$
$\delta(q_1, a) = q_1$



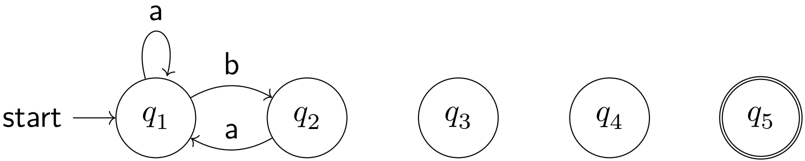
Hw2 Problem1

$\{q_1, q_2, q_3, q_4, q_5\}$
$\{a, b\}$
q_1
$\{q_5\}$
$\delta(q_1, b) = q_2$



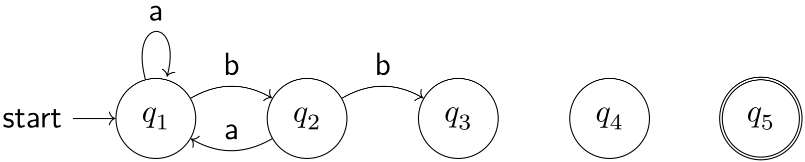
Hw2 Problem1

$\{q_1, q_2, q_3, q_4, q_5\}$
$\{a, b\}$
q_1
$\{q_5\}$
$\delta(q_2, a) = q_1$



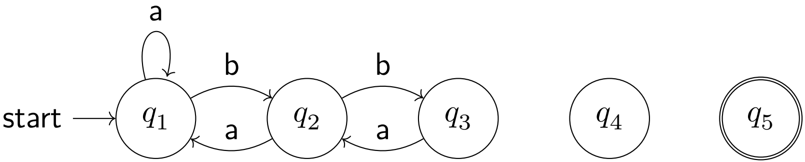
Hw2 Problem1

$\{q_1, q_2, q_3, q_4, q_5\}$
$\{a, b\}$
q_1
$\{q_5\}$
$\delta(q_2, b) = q_3$



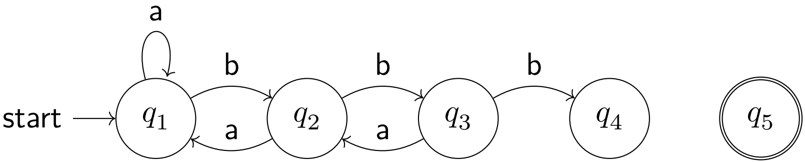
Hw2 Problem1

$\{q_1, q_2, q_3, q_4, q_5\}$
$\{a, b\}$
q_1
$\{q_5\}$
$\delta(q_3, a) = q_2$



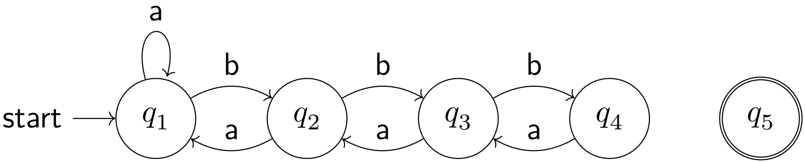
Hw2 Problem1

$\{q_1, q_2, q_3, q_4, q_5\}$
$\{a, b\}$
q_1
$\{q_5\}$
$\delta(q_3, b) = q_4$



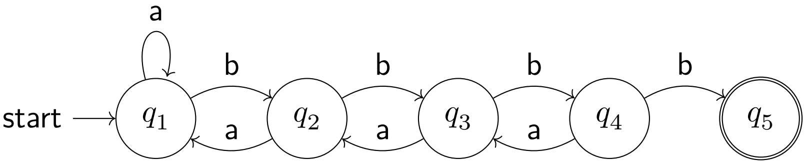
Hw2 Problem1

$\{q_1, q_2, q_3, q_4, q_5\}$
$\{a, b\}$
q_1
$\{q_5\}$
$\delta(q_4, a) = q_3$



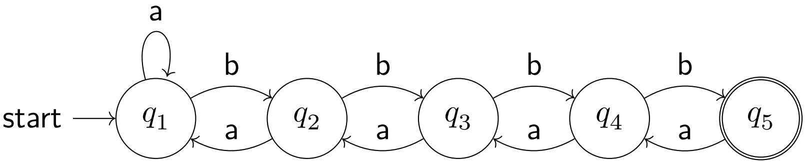
Hw2 Problem1

$\{q_1, q_2, q_3, q_4, q_5\}$
$\{a, b\}$
q_1
$\{q_5\}$
$\delta(q_4, b) = q_5$



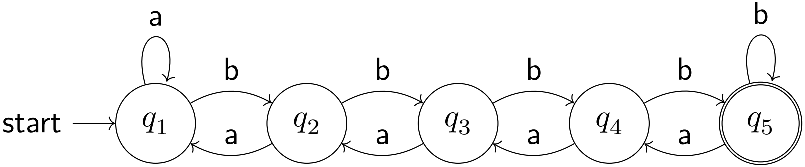
Hw2 Problem1

$\{q_1, q_2, q_3, q_4, q_5\}$
$\{a, b\}$
q_1
$\{q_5\}$
$\delta(q_5, a) = q_4$



Hw2 Problem1

$\{q_1, q_2, q_3, q_4, q_5\}$
$\{a, b\}$
q_1
$\{q_5\}$
$\delta(q_5, b) = q_5$



Hw2 Problem1

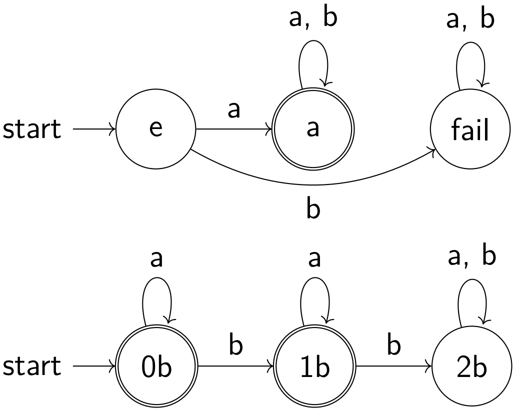
描述這台自動機 recognize 的語言

將一次 b 視為 +1，一次 a 視為-1，初始值為 0，從字串左邊到右邊進行 +1 或-1
值域控制在 0-4 之間，0 減下去仍為 0，4 加上去仍是 4，使最終結果為 +4 的字串

助教我沒有嚴格要求一定要寫對（不過考試會改比較嚴）
常錯的點：以 b 結尾且 b 的數量比 a 多 4
反例：aaaaaaaaabbbb

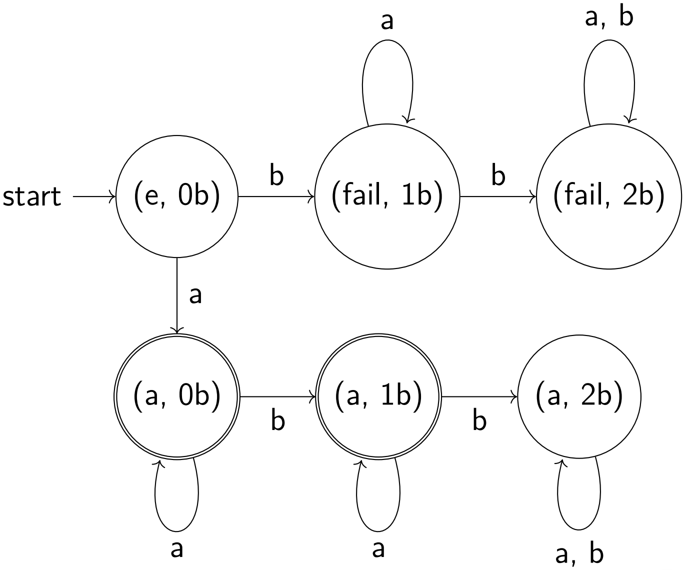
Hw2 Problem2

第一小題：開頭為 a，且最多一個 b



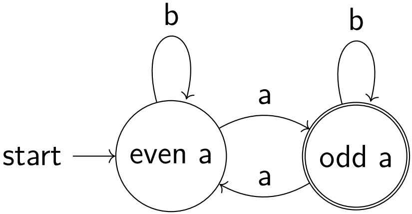
Hw2 Problem2

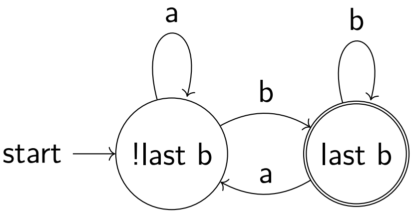
第一小題：開頭為 a，且最多一個 b



Hw2 Problem2

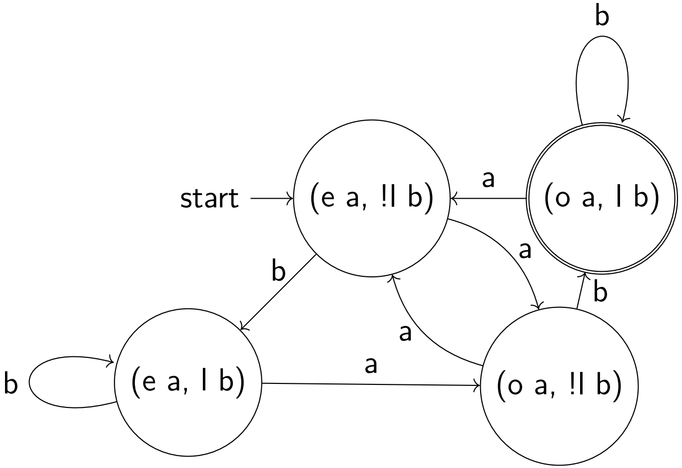
第二小題：有奇數個 a，且以 b 結尾





Hw2 Problem2

第二小題：有奇數個 a，且以 b 結尾

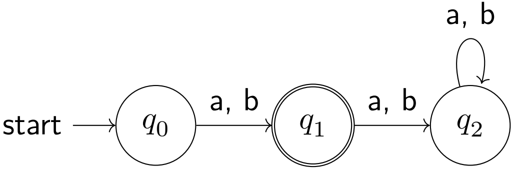


Hw2 Problem2

注意題目要你建構的是 DFA
還有請遵照題目步驟進行
雖然你可能覺得這題直接建構也很簡單

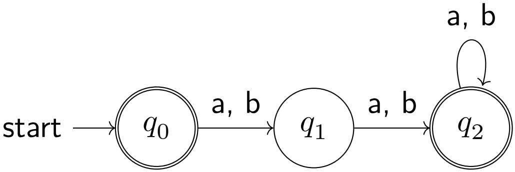
Hw2 Problem3

第一小題：只接受字串 a 與 b



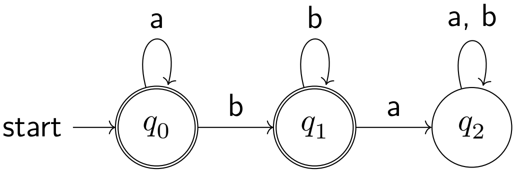
Hw2 Problem3

第一小題：只拒絕字串 a 與 b



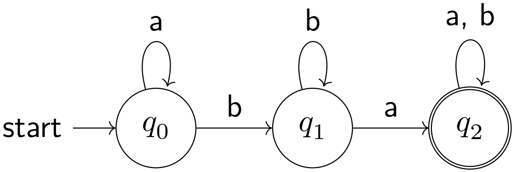
Hw2 Problem3

第二小題：字串屬於 a*b*



Hw2 Problem3

第二小題：字串不屬於 a*b*



Hw2 Problem4

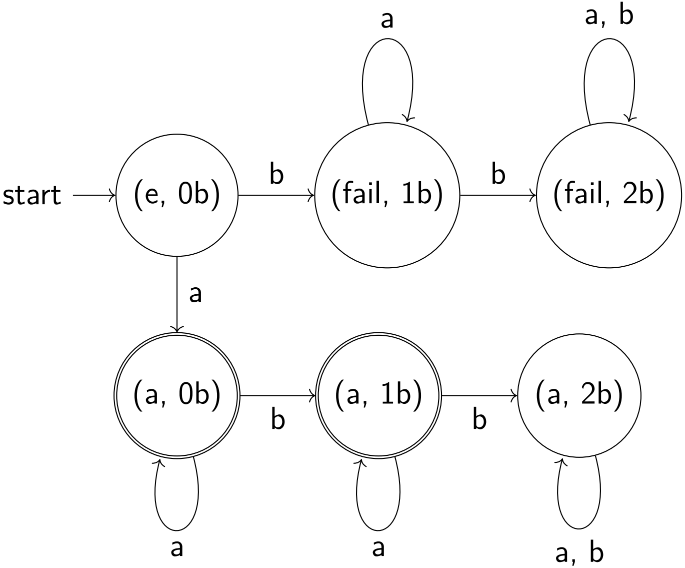
DFA 解法 (比較難)
若 A 是 regular 的，那麼就能找到一台 DFA M recognizes A
試著建構一個 DFA M' recognizes A^R
以 M 為基礎：
原本是從 initial state 走到 accepting state，現在要倒著走，從 accepting state 走回去
所以得將所有箭頭轉向
並將 initial 變成 accepting，accepting 變成 initial
可是我們只會有一個 initial state，怎麼辦？

Hw2 Problem4

選定一個 initial state
找出所有從 initial state 走到 accepting state 的路徑
節點與邊的數量都是有限的，simple path 與 cycle 的數量也是有限的
這些 simple paths 與 cycles 的組合是有限的
所以可以畫出有限多個 DFA
把這些 DFA 全都 Union 在一起即為所求

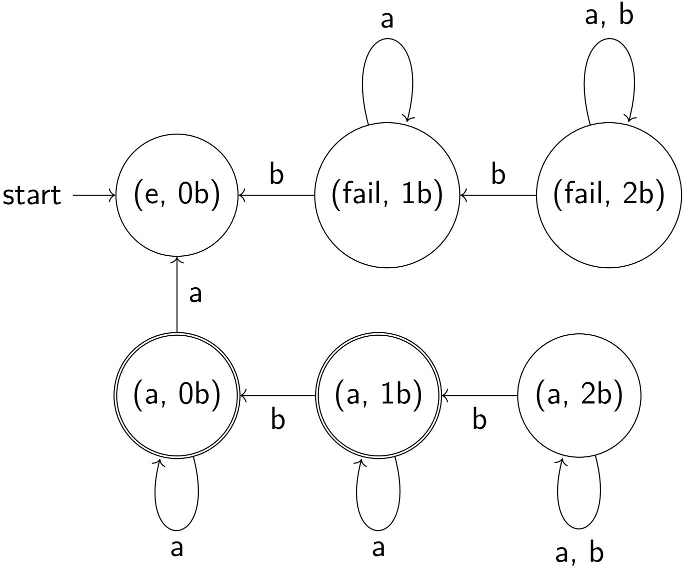
Hw2 Problem4

以這台自動機為例子



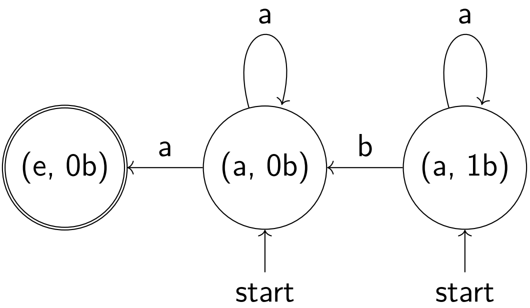
Hw2 Problem4

將箭頭反過來



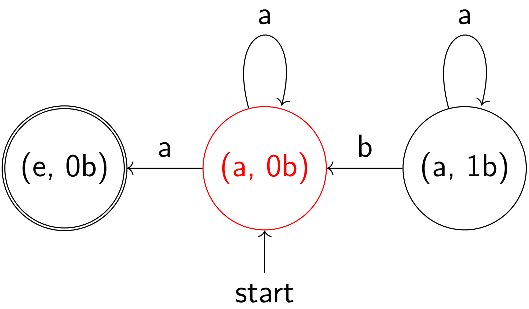
Hw2 Problem4

將 initial 與 accepting 反轉，並刪去不會到達的節點



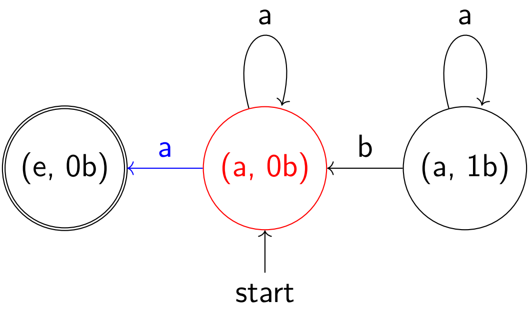
Hw2 Problem4

先選定 (a, 0b) 當起始



Hw2 Problem4

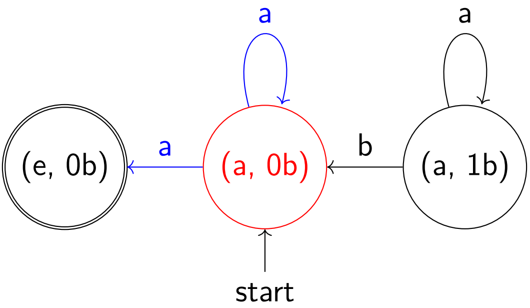
能到達 (e, 0b) 的簡單路徑只有一條



a

Hw2 Problem4

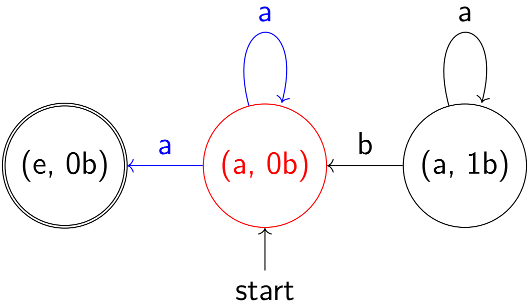
路徑上有一個迴圈



a^*a

Hw2 Problem4

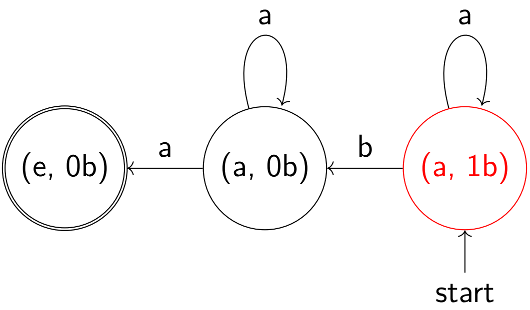
迴圈能用 star 表示；不同迴圈的選擇能用 Union 表示；在路徑中往前走能用 concat 表示。你畫得出這個小語言的 DFA



a^*a

Hw2 Problem4

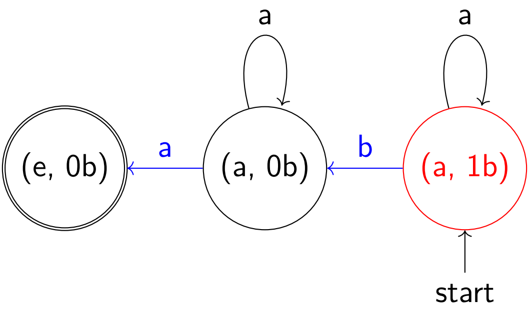
再選定 (a, 1b) 當起始



a^*a

Hw2 Problem4

能到達 (e, 0b) 的簡單路徑也只有一條

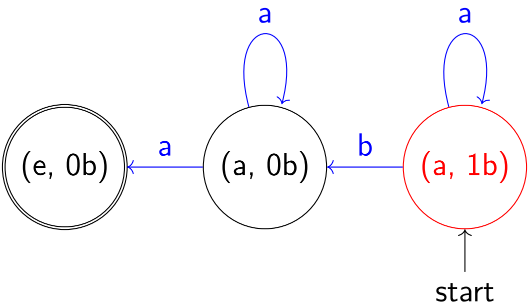


a^*a

$b \quad a$

Hw2 Problem4

路徑上有兩個迴圈

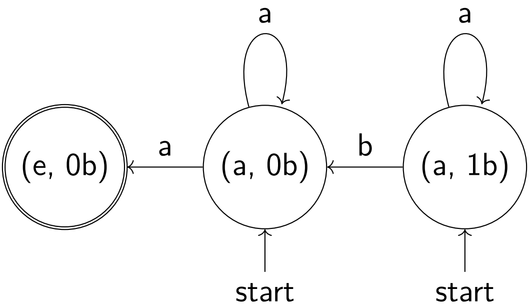


a^*a

a^*ba^*a

Hw2 Problem4

找完所有路徑後，將這些 DFA 給 Union 起來



a^*a

a^*ba^*a

Hw2 Problem4

NFA 解法 (比較容易)
建構一個 M' recognizes A^R
一樣將箭頭轉向、initial 與 accepting 互換...... 噢
我們只會有一個 initial state，所以我們的做法是創造一個 initial state，並用 ϵ 連到所有原本的 accepting state

注意我們建構出來的 M' 是 NFA

Hw2 Problem5

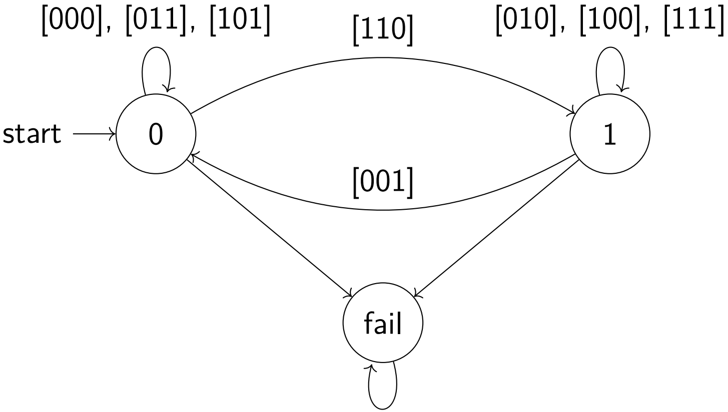
簡單來說，B 接受的字串將字元從左到右排列，看起來會是二進位的直式加法
直式加法很明顯從右邊算起來簡單，所以從 reverse 下手

首先我們要關注運算的正確性，不正確的話就直接掛掉
其次也要關注進位，卡著進位的狀態是不能結束的

Hw2 Problem5

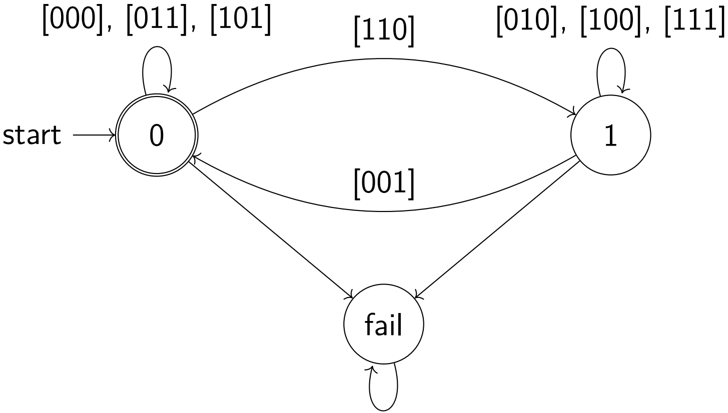
正確性

進位



Hw2 Problem5

正確性
進位



Hw2 Problem5

$$Q = \{q_0, q_1, q_f\}$$
$$F = \{q_0\}$$
$$\delta(q_i, \begin{bmatrix} a \\ b \\ c \end{bmatrix}) =$$
$$\begin{cases} q_{(i+a+b)/2} & if\ i = 0\ or\ 1\ and\ c = (i + a + b)\ mod\ 2 \\ q_f & otherwise \end{cases}$$

Hw2 Problem6

改寫 Union 的證明使之能用在 $A_1 A_2$ 的 alphabet 不同的狀況

Hw2 Problem6

- $M_1 = (Q_1, \Sigma_1, \delta_1, q_1, F_1)$
- $M_2 = (Q_2, \Sigma_2, \delta_2, q_2, F_2)$
- $M = (Q, \Sigma, \delta, q, F)$

Hw2 Problem6

- $M_1 = (Q_1, \Sigma_1, \delta_1, q_1, F_1)$
- $M_2 = (Q_2, \Sigma_2, \delta_2, q_2, F_2)$
- $M = (Q, \Sigma, \delta, q, F)$
- $Q = (Q_1 \cup \{q_f\}) \times (Q_2 \cup \{q_f\})$

Hw2 Problem6

- $M_1 = (Q_1, \Sigma_1, \delta_1, q_1, F_1)$
- $M_2 = (Q_2, \Sigma_2, \delta_2, q_2, F_2)$
- $M = (Q, \Sigma, \delta, q, F)$
- $Q = (Q_1 \cup \{q_f\}) \times (Q_2 \cup \{q_f\})$
- $\Sigma = \Sigma_1 \cup \Sigma_2$

Hw2 Problem6

- $M_1 = (Q_1, \Sigma_1, \delta_1, q_1, F_1)$
- $M_2 = (Q_2, \Sigma_2, \delta_2, q_2, F_2)$
- $M = (Q, \Sigma, \delta, q, F)$
- $Q = (Q_1 \cup \{q_f\}) \times (Q_2 \cup \{q_f\})$
- $\Sigma = \Sigma_1 \cup \Sigma_2$
- $q = (q_1, q_2)$

Hw2 Problem6

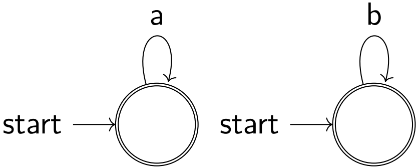
- $M_1 = (Q_1, \Sigma_1, \delta_1, q_1, F_1)$
- $M_2 = (Q_2, \Sigma_2, \delta_2, q_2, F_2)$
- $M = (Q, \Sigma, \delta, q, F)$
- $Q = (Q_1 \cup \{q_f\}) \times (Q_2 \cup \{q_f\})$
- $\Sigma = \Sigma_1 \cup \Sigma_2$
- $q = (q_1, q_2)$
- $F = \{(r_1, r_2) | r_1 \in F_1 \vee r_2 \in F_2\}$

Hw2 Problem6

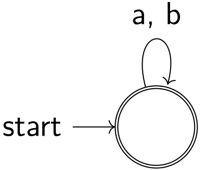
- $M_1 = (Q_1, \Sigma_1, \delta_1, q_1, F_1)$
- $M_2 = (Q_2, \Sigma_2, \delta_2, q_2, F_2)$
- $M = (Q, \Sigma, \delta, q, F)$
- $Q = (Q_1 \cup \{q_f\}) \times (Q_2 \cup \{q_f\})$
- $\Sigma = \Sigma_1 \cup \Sigma_2$
- $q = (q_1, q_2)$
- $F = \{(r_1, r_2) | r_1 \in F_1 \vee r_2 \in F_2\}$
- $\delta((r_1, r_2), a) =$
$$\begin{cases} (\delta_1(r_1, a), \delta_2(r_2, a)) & r_1 \neq q_f \wedge r_2 \neq q_f \wedge a \in \Sigma_1 \cap \Sigma_2 \\ (\delta_1(r_1, a), q_f) & r_1 \neq q_f \wedge a \in \Sigma_1 \wedge (r_2 = q_f \vee a \notin \Sigma_2) \\ (q_f, \delta_2(r_2, a)) & r_2 \neq q_f \wedge a \in \Sigma_2 \wedge (r_1 = q_f \vee a \notin \Sigma_1) \\ (q_f, q_f) & otherwise \end{cases}$$

Hw2 Problem6

為何需要 Fail？考慮這個情境
兩台 DFA



第一台自動機 recognize a^*，第二台則是 b^*
如果沒有 fail 狀態...



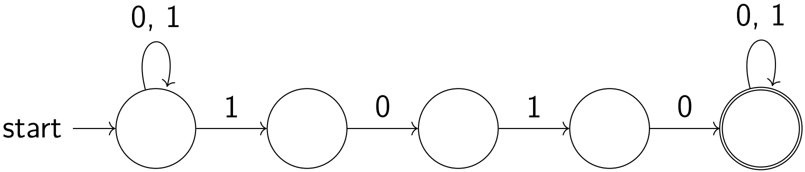
這台自動機接受的是 $(a|b)^*$
字串「ab」會被新機接受，但卻不會被前兩台接受，因為第一台不吃 b，第二台不吃 a

Hw3 Problem1

請用題目要求的狀態數量畫出指定的 NFA

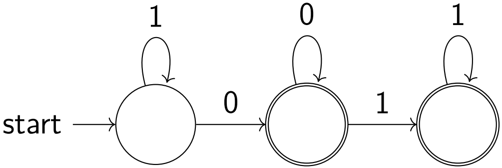
Hw3 Problem1

第一小題：一個包含子字串 1010 的字串，5 個狀態



Hw3 Problem1

第二小題：$1^*0^+1^*$，3 個狀態

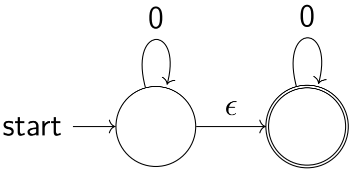


對這張圖的理解：
走到第二個狀態的字串 $1^*0^+\epsilon$
走到第三個狀態的字串 $1^*0^+1^+$
而 $1^*0^+1^*$ 就是上面兩者的聯集

Hw3 Problem2

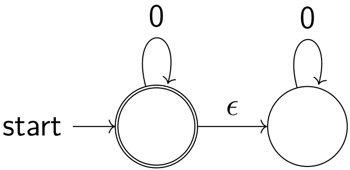
這題有兩個部分
給出一個例子說明把 NFA 的 accepting 調換過來不會產生 complement
以及 NFA 辨識的語言的 complement 是否能被另外一個 NFA 辨識
也就是「能被 NFA 辨識」這件事 closed under complement

Hw3 Problem2



接受 0^*

Hw3 Problem2



還是接受 0^*

Hw3 Problem2

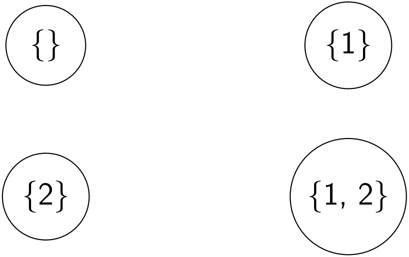
NFA 可以轉換成等價的 DFA
已知 DFA 的 accepting 調換之後就能辨識其語言的 complement
又 DFA 就是一種 NFA，所以是 closed under complement

Hw3 Problem3

利用 Th 1.39 的方法 (subset construction) 建構出等價的 DFA

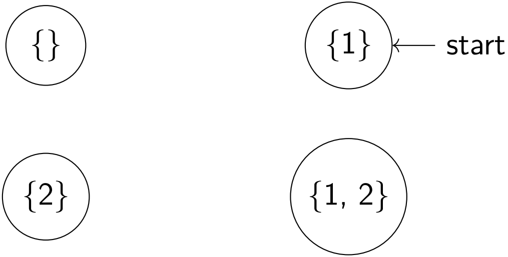
Hw3 Problem3

第一小題



Hw3 Problem3

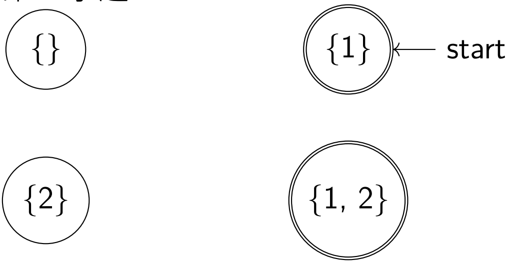
第一小題



$$q_0' = E(\{1\}) = \{1\}$$

Hw3 Problem3

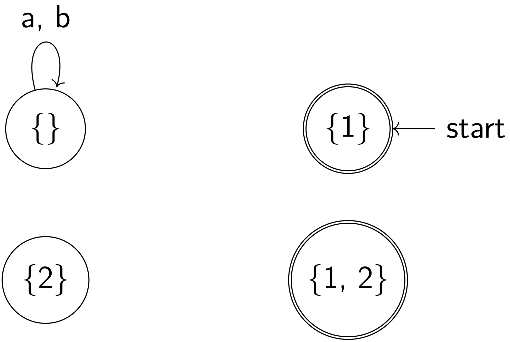
第一小題



$$F' = \{\{1\}, \{1, 2\}\}$$

Hw3 Problem3

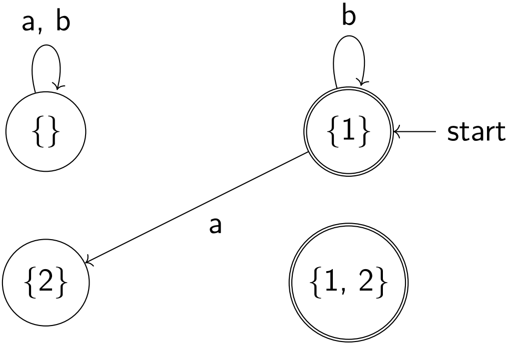
第一小題



$$\delta'(\{\}, a) = \{\}$$
$$\delta'(\{\}, b) = \{\}$$

Hw3 Problem3

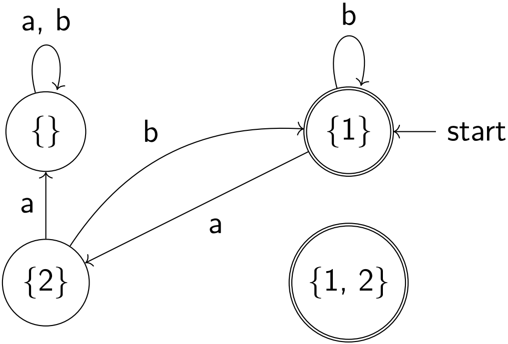
第一小題



$$\delta'(\{1\}, a) = \{2\}$$
$$\delta'(\{1\}, b) = \{1\}$$

Hw3 Problem3

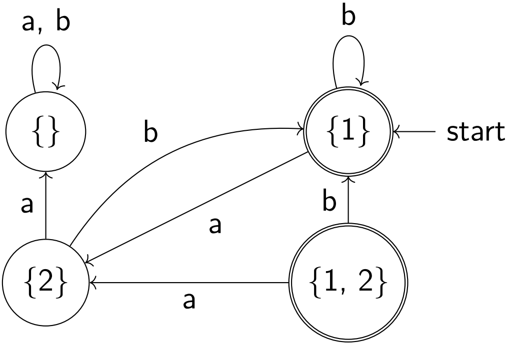
第一小題



$\delta'(\{2\}, a) = \{\}$
$\delta'(\{2\}, b) = \{1\}$

Hw3 Problem3

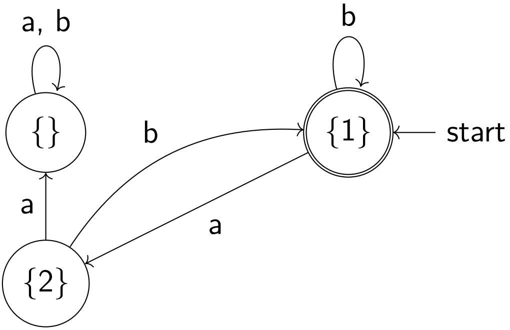
第一小題



$$\delta'(\{1,2\},a) = \{2\}$$
$$\delta'(\{1,2\},b) = \{1\}$$

Hw3 Problem3

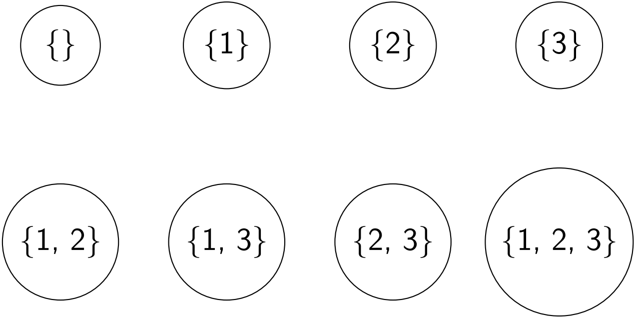
第一小題



刪去無法到達的狀態

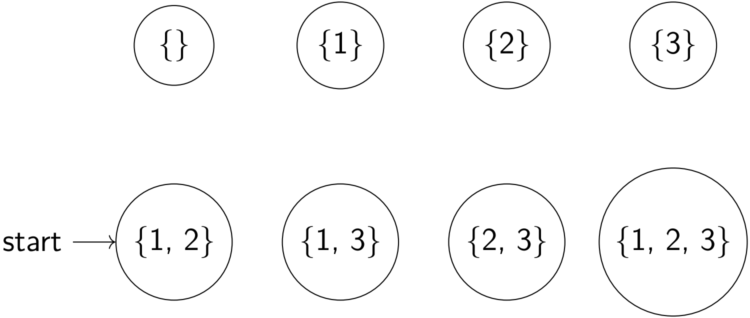
Hw3 Problem3

第二小題



Hw3 Problem3

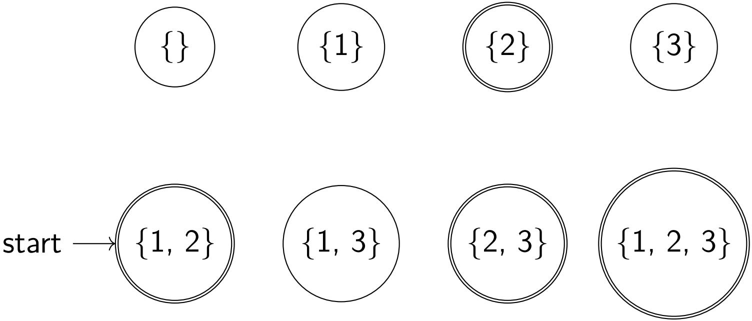
第二小題



$q_0' = E(\{1\}) = \{1, 2\}$

Hw3 Problem3

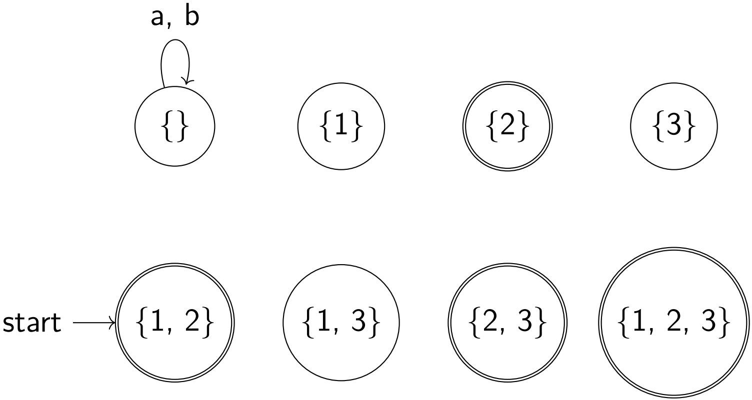
第二小題



$$F' = \{\{2\}, \{1,2\}, \{2,3\}, \{1,2,3\}\}$$

Hw3 Problem3

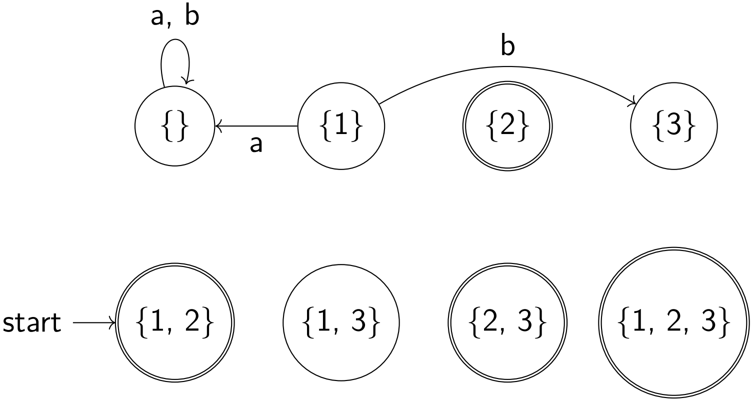
第二小題



$$\delta'(\{\}, a) = \{\}$$
$$\delta'(\{\}, b) = \{\}$$

Hw3 Problem3

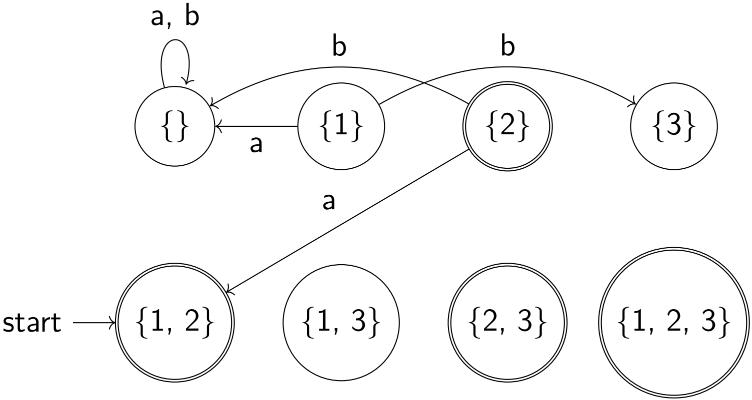
第二小題



$\delta'(\{1\}, a) = \{\}$
$\delta'(\{1\}, b) = \{3\}$
注意我們並沒有把 {1} 偷偷展開成 {1, 2}

Hw3 Problem3

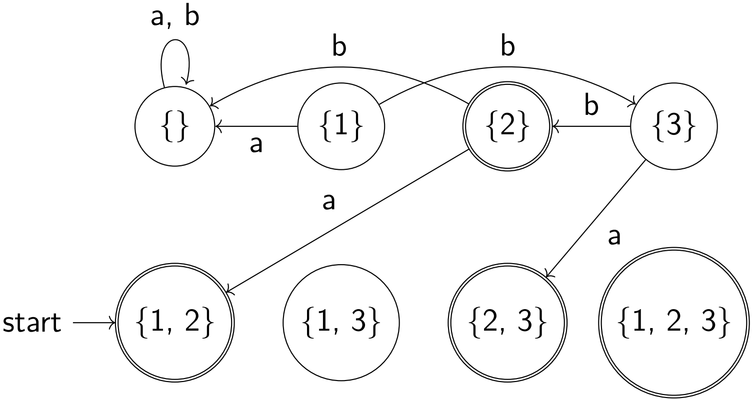
第二小題



$\delta'(\{2\}, a) = \{1, 2\}$
$\delta'(\{2\}, b) = \{\}$

Hw3 Problem3

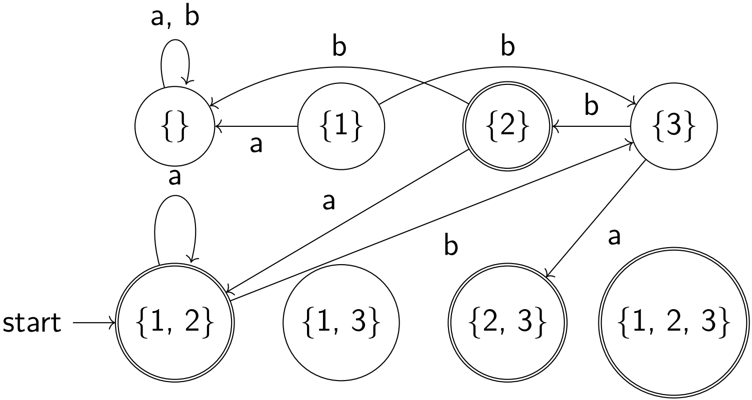
第二小題



$\delta'(\{3\}, a) = \{2, 3\}$
$\delta'(\{3\}, b) = \{2\}$

Hw3 Problem3

第二小題

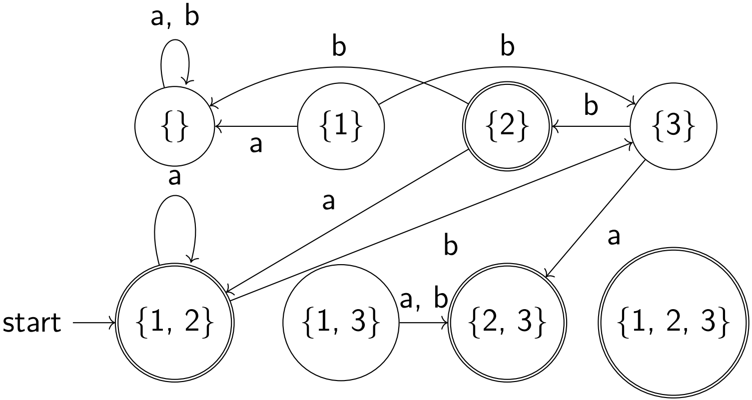


$\delta'(\{1,2\},a) = \{1,2\}$

$\delta'(\{1,2\},b) = \{3\}$

Hw3 Problem3

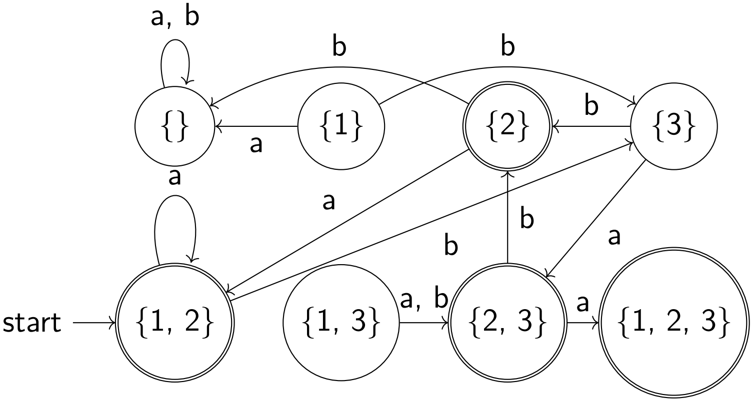
第二小題



$$\delta'(\{1,3\},a) = \{2,3\}$$
$$\delta'(\{1,3\},b) = \{2,3\}$$

Hw3 Problem3

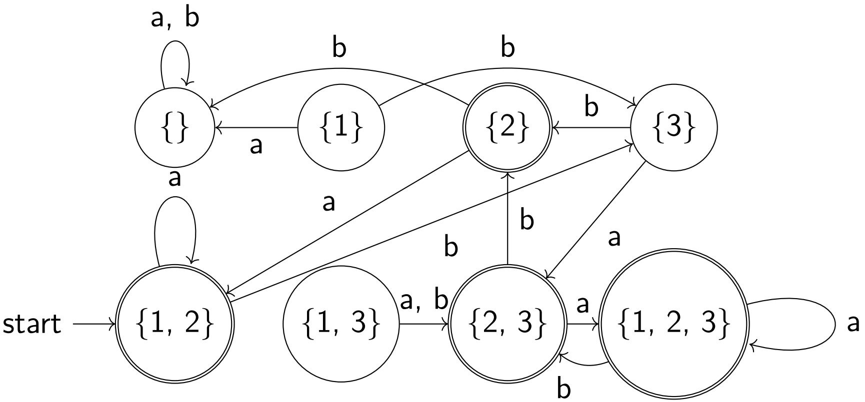
第二小題



$\delta'(\{2,3\},a) = \{1,2,3\}$
$\delta'(\{2,3\},b) = \{2\}$

Hw3 Problem3

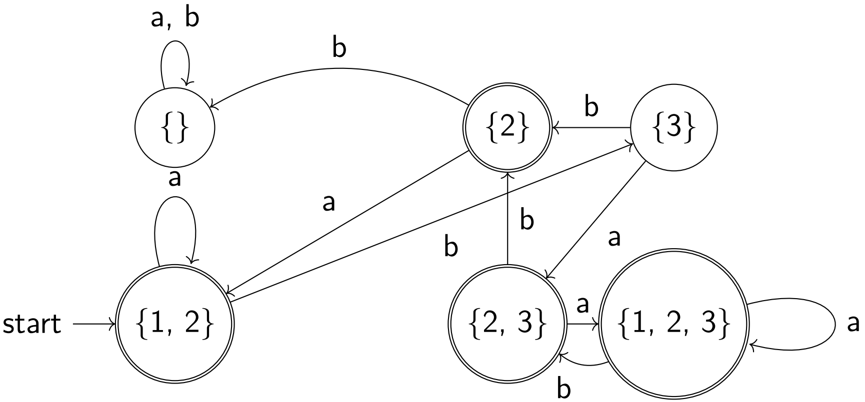
第二小題



$$\delta'(\{1,2,3\},a) = \{1,2,3\}$$
$$\delta'(\{1,2,3\},b) = \{2,3\}$$

Hw3 Problem3

第二小題

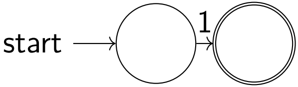


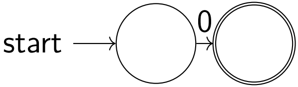
刪去無法到達的狀態

{1} 與 {1, 3} 這兩個節點一定無法到達 (Why?)

Hw3 Problem4

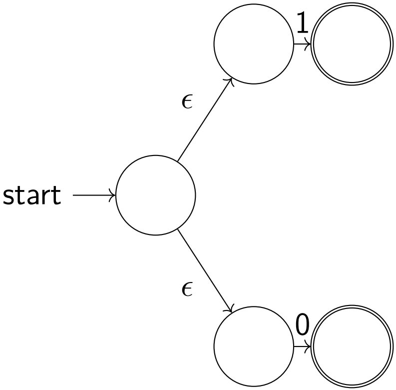
$0 \quad 1$





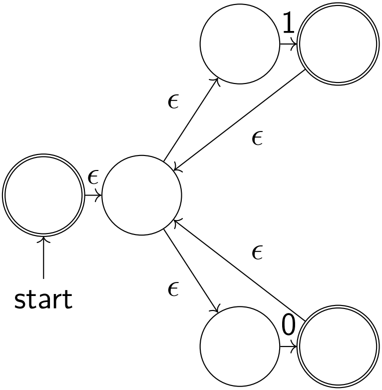
Hw3 Problem4

$0 \cup 1$



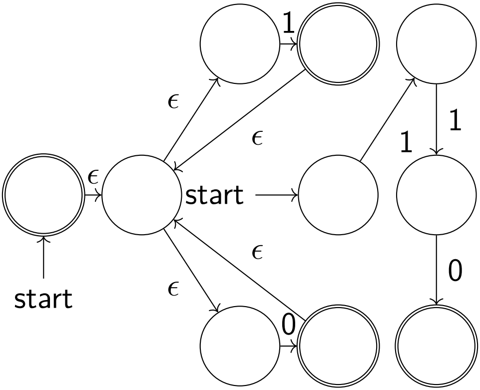
Hw3 Problem4

$(0 \cup 1)^*$



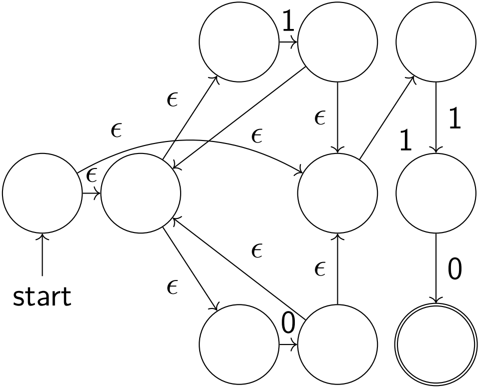
Hw3 Problem4

$(0 \cup 1)^*110$



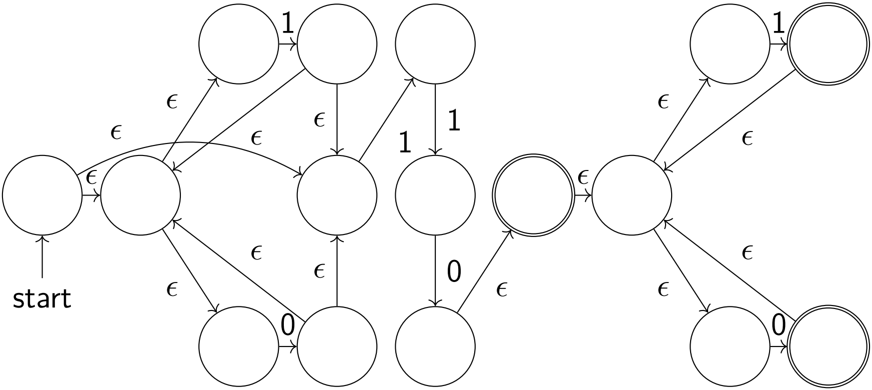
Hw3 Problem4

$(0 \cup 1)^*110$



Hw3 Problem4

$(0 \cup 1)^{*}110(0 \cup 1)^{*}$



Hw3 Problem5

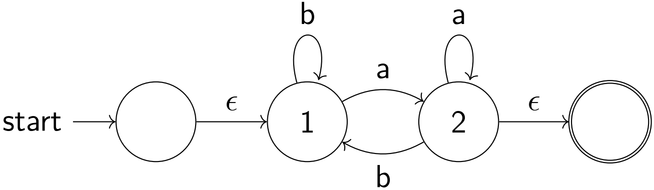
第一小題：字串的奇數位置一定是 1
考慮整體長度為奇數的情境：$(1(0 \cup 1))^*1$
考慮整體長度為偶數的情境：$(1(0 \cup 1))^*$
組合起來：$(1(0 \cup 1))^*(\epsilon \cup 1)$

Hw3 Problem5

第二小題：字串不包含子字串 011
只要出現 0，後面就不能出現 2 個以上的 1
暗示第一次遇見 0 之前可以有一堆 1 (開頭為 1^*)
後面則可以看成有一堆 0，隨便在其右方加上一個 1
有被添加 1 的 0，把它看做一個整體 01，沒有的則是 0
於是右邊可以變成 $(0 \cup 01)^*$
合起來就是 $1^*(0 \cup 01)^*$

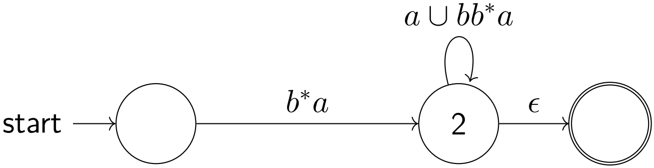
Hw3 Problem6

第一小題



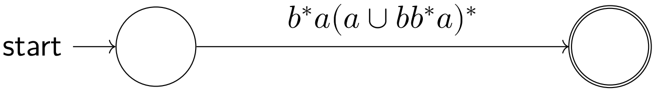
Hw3 Problem6

第一小題



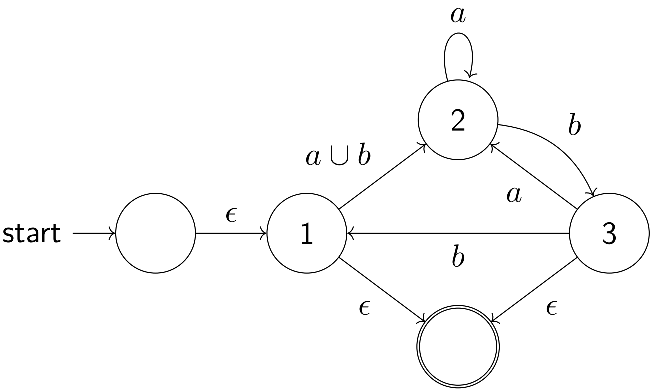
Hw3 Problem6

第一小題



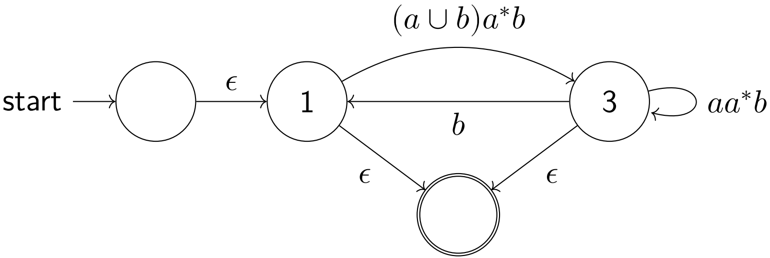
Hw3 Problem6

第二小題



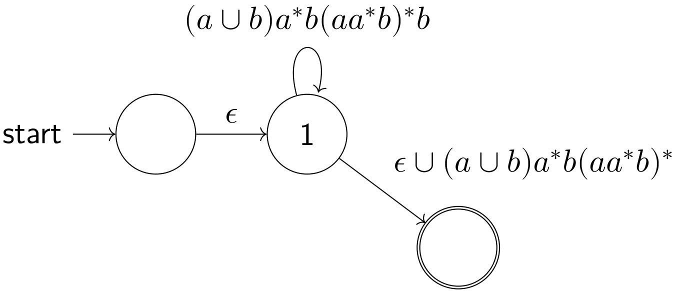
Hw3 Problem6

第二小題



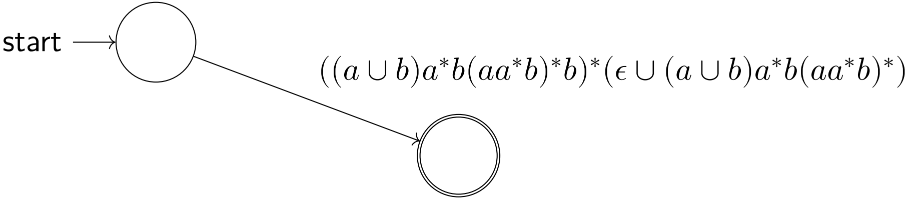
Hw3 Problem6

第二小題



Hw3 Problem6

第二小題



Hw3 Problem6

依照去除 state 的順序不同，答案會有不同的長相

1-2-3 $\epsilon \cup ((a \cup b)a^*b((b(a \cup b) \cup a)a^*b)^*(\epsilon \cup b))$

1-3-2 $\epsilon \cup ((a \cup b)(a \cup (b((b(a \cup b)) \cup a)))^*(b(\epsilon \cup b)))$

2-1-3 $\epsilon \cup (((a \cup b)a^*b)((aa^*b) \cup (b(a \cup b)a^*b))^*(\epsilon \cup b))$

2-3-1 $((a \cup b)a^*b(aa^*b)^*b)^*(\epsilon \cup (a \cup b)a^*b(aa^*b)^*)$

3-1-2 $\epsilon \cup ((a \cup b)((a \cup ba) \cup (bb(a \cup b)))^*(b \cup bb))$

3-2-1 $((a \cup b)(a \cup ba)^*bb)^*(\epsilon \cup (a \cup b)(a \cup ba)^*b)$

Hw3 Problem7

q_1
q_1 吃 1 輸出 0 跑到 q_1
q_1 吃 0 輸出 0 跑到 q_1
q_1 吃 2 輸出 1 跑到 q_2
q_2 吃 0 輸出 0 跑到 q_1
q_1 吃 1 輸出 0 跑到 q_1
q_1 吃 2 輸出 1 跑到 q_2
輸出 001001

Hw3 Problem7

q_1
q_1 吃 b 輸出 1 跑到 q_3
q_3 吃 a 輸出 0 跑到 q_1
q_1 吃 b 輸出 1 跑到 q_3
q_3 吃 b 輸出 1 跑到 q_2
q_2 吃 b 輸出 0 跑到 q_1
q_1 吃 a 輸出 1 跑到 q_2
輸出 101101

Hw3 Problem8

An FST T is a 5-tuple $(Q, \Sigma, \Gamma, \delta, q_0)$
Q is a finite set of states
Σ is a finite set of input symbols
Γ is a finite set of output symbols
$\delta : Q \times \Sigma \to Q \times \Gamma$ is the transition function
$q_0 \in Q$ is the start state

Let $w = w_1 w_2 ... w_n$ be a string over Σ and $x = x_1 x_2 ... x_n$ a string over Γ
We say T produces output x on input w with the sequence of states $r_0, r_1, ..., r_n$ when

- $r_0 = q_0$
- $\delta(r_i, w_{i+1}) = (r_{i+1}, x_{i+1})$ for $i = 0, 1, ..., i-1$

Hw3 Problem8

最容易錯的點：
沒注意到題目有說要寫出 FST 的運作過程

Hw4 Problem1

若 A 與 B 皆為 regular language，那一定找得到 DFA M_A, M_B
shuffle 的過程，吃幾個（0 個以上）$w_A \in \Sigma^*$ 的字元，又吃幾個 $w_B \in \Sigma^*$ 的字元
相當於讓 M_A 吃 w_A 走幾步之後讓 M_B 吃 w_B 走幾步，往復直到 M_A 與 M_B 都到各自的 accepting state

Hw4 Problem1

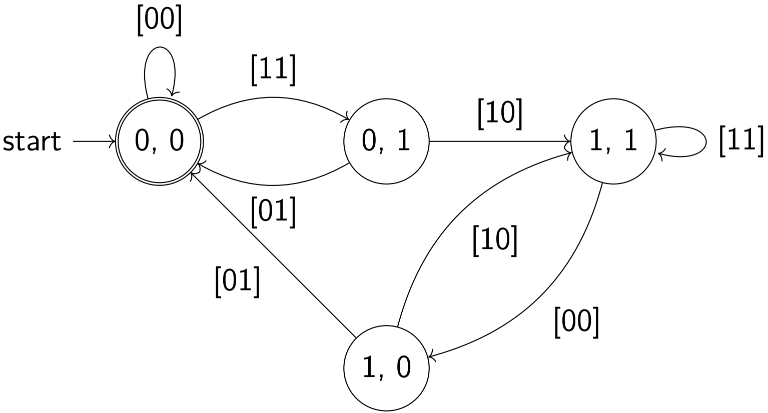
$M_i = (Q_i, \Sigma, \delta_i, q_i, F_i)$ for $i \in \{A, B\}$
建構一個 NFA $M = (Q, \Sigma, \delta, q_0, F)$
$Q = Q_A \times Q_B \times \{A, B\}$
$q_0 = (q_A, q_B, A)$
$F = F_A \times F_B \times \{B\}$

$$\delta((r_1, r_2, X), a) = \begin{cases} (\delta_A(r_1, a), r_2, A) & X = A \\ (r_1, \delta_B(r_2, a), B) & X = B \\ (r_1, r_2, A) & X = B \wedge a = \epsilon \\ (r_1, r_2, B) & X = A \wedge a = \epsilon \end{cases}$$

Hw4 Problem2

利用類似作業 2 第五題的建構法
從右往左比較容易
三倍就是 1+2 倍，二進制的 2 倍就是左移一格
所以要考慮的東西：1 倍數本身、從右邊帶來的數（2 倍數）、進位

Hw4 Problem2



狀態 (a, b) 代表此時有進位 a、右邊的數字是 b
輸入 [x y]，y = (a + b + x) % 2
跳轉到 (c, d) 狀態，c = (a + b + x) / 2, d = x

Hw4 Problem3

假設 pumping length 為 p
我們關注字串 $(\begin{bmatrix}0\\1\end{bmatrix})^p(\begin{bmatrix}1\\0\end{bmatrix})^p$
試圖將其分成 xyz，由於 $|xy| \leq p$，y 必然只由 $\begin{bmatrix}0\\1\end{bmatrix}$ 組成
而 $xy^2z \notin E$ (第一 row 的 0 與第二 row 的 0 數量不一樣，一定不會是 reverse)
證明完畢

Hw4 Problem4

C_n 是不是 regular
對一個 n 而言，可以透過餘數是不是 0 來判別
舉例，二進制數字 X 與 Xy 之間滿足 $Xy \equiv 2 \times X + y (\text{mod } n)$
若已知 X 的餘數為 R
則數字 Xy 的餘數為 $(2 \times R + y)$ mod n

Hw4 Problem4

建構 DFA M_n
$Q = \{q_i \mid 0 \le i \le n-1\}$
$\Sigma = \{0, 1\}$
$\delta(q_i, a) = q_{(2\times i+a) \bmod n}$
起始狀態為 q_0
$F = \{q_0\}$

C_n 包不包含 ϵ ?
Naturally $\epsilon \notin C_n$
不過以這題的結果而言都一樣：
如果包含，那這台自動機辨識的就是 C_n
如果不包含，則 $C_n = L(M_n) - \{\epsilon\} = \overline{\overline{L(M_n)} \cup \{\epsilon\}}$，依然是 regular 的
由此可知 C_n 為 regular language

Hw4 Problem5

令 A 為所有回文字串組成的 language，題目所求的即為 $\overline{A}$
已知 regular language 的 complement 也會是 regular
若 $\overline{A}$ 為 regular，那麼 A 也會是 regular
也就是說，若 A 不是 regular，則可知道 $\overline{A}$ 也不是 regular

Hw4 Problem5

證明 A 不是 regular
取 $w = 0^p10^p$，$w \in A$，p 為 pumping length
由於 $|xy| \leq p$，y 一定是第一個 0^p 裡頭的非空子字串
令 $|y| = n > 0$，$xy^2z = 0^{p+n}10^p \notin A$
得證 A 不是 regular，而 $\overline{A}$ 也不是

Hw4 Problem6

點子：讓 k 個 state 變一樣，那就先讓其中兩個變一樣，再讓三個變一樣...

由於 M 是一個 synchronizable DFA
對 M 的兩個 state (q_a, q_b) 而言

$$(q_a, q_b) \xrightarrow{s} (h, h)$$

那麼必然會有一個字串 s_{q_a,q_b} 使

$$(q_a, q_b) \xrightarrow{s_{q_a,q_b}} (r, r)$$

s_{q_a,q_b} 是個能讓這兩個 state (q_a, q_b) 同步走到同一個 state 的字串
那麼 s_{q_a,q_b} 的長度可以多短呢？

Hw4 Problem6

由 k 個 state 組成的 pair 最多就 k^2 種
若是 s_{q_a,q_b} 的長度超過 k^2，則

$$(q_a, q_b) \xrightarrow{*} (q_c, q_d) \xrightarrow{\text{substring of } s_{q_a,q_b}} (q_c, q_d) \xrightarrow{*} (r, r)$$

過程中一定會路過重複的 pair (q_c, q_d)
那麼把這個 substring 從 s_{q_a,q_b} 去掉之後仍有相同效果
所以一定存在某個 s_{q_a,q_b} 其長度不長於 k^2

Hw4 Problem6

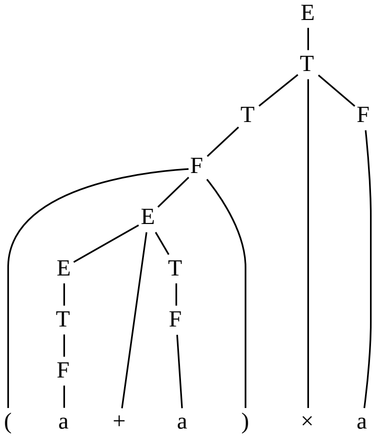
現在回到 k 個 state $(q_1, ..., q_k)$，讓它們同步進行
首先找到使 (q_1, q_2) 跑到同一個 state 的字串 s_{q_1,q_2}
使這 k 個 state 變成 $(q_1', q_2', ..., q_k')$
由於 $q_1' = q_2'$，現在這 k 個 state 至多只會有 $k-1$ 種不同樣子
接著找 (q_2', q_3') 對應的 $s_{q_2',q_3'}$，使得前三個 state 相等，k 個 state 最多只剩下 $k-2$ 種不同樣子

以此推類，在這個過程進行了 $k-1$ 次之後，這 k 個 state 就會只有 1 種樣子
也就是我們找到了題目要求的字串，這個字串的長度最多是 $(k-1) \times k^2 \leq k^3$ for all $k \in \mathbb{N}$

Hw5 Problem1

第一小題：

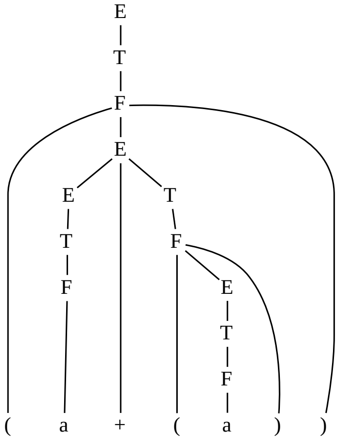
$E \Rightarrow T \Rightarrow T \times F \Rightarrow F \times F \Rightarrow (E) \times F \Rightarrow (E + T) \times F \Rightarrow$
$(T + T) \times F \Rightarrow (F + T) \times F \Rightarrow (a + T) \times F \Rightarrow (a + F) \times F \Rightarrow$
$(a + a) \times F \Rightarrow (a + a) \times a$



Hw5 Problem1

第一小題：

$$E \Rightarrow T \Rightarrow F \Rightarrow (E) \Rightarrow (E+T) \Rightarrow (F+T) \Rightarrow (a+T) \Rightarrow$$
$$(a+F) \Rightarrow (a+(E)) \Rightarrow (a+(T)) \Rightarrow (a+(F)) \Rightarrow (a+(a))$$



Hw5 Problem2

第一小題

$S \to CA$

$A \to CCA \mid \epsilon$

$C \to 0 \mid 1$

C 生成長度為 1 的字串，A 生成長度為偶數的字串，而初始符號 S 則在 A 的前頭添加一個字

Hw5 Problem2

第二小題
$S \to 0S0 \mid 1S1 \mid C \mid \epsilon$
$C \to 0 \mid 1$

為何不直接 CSC 呢？因為兩個 C 可能會生出不同字，就不會是回文了

Hw5 Problem3

造出一個 CFG 生成 $x_1\#x_2\#...\#x_k$ 使得其中兩個子字串 x_i, x_j 互為 reverse

假設 $i \leq j$
那麼 x_i 與 x_j 的聯繫可以看成這樣：

$$\underbrace{\cdots\cdots}_{L}\overbrace{c_1c_2\cdots c_n}^{x_i}\underbrace{\cdots\cdots}_{T}\overbrace{c_n\cdots c_2c_1}^{x_j}\underbrace{\cdots\cdots}_{R}$$

此時我們假定 x_i, x_j 都已經生成完畢
左端、中間與右端分別有三個 CFG 的變數

Hw5 Problem3

如果 $i = 1$，那麼 L 會生成空字串
若不是，那麼應該生成 $X\#$，其中 X 會生成一個沒有限制的，以 $\#$ 分隔的字串（實際上就等價於 $\{a, b, \#\}^*$）
R 同理

比較關鍵的是 T 能夠生成什麼？
T 可以生成 $\#X\#$，當 $j - i \geq 2$
T 可以生成 $\#$，當 $j - i = 1$
T 也能生成 ϵ 與一個字元，當 $i = j$（回文）

Hw5 Problem3

$$S \to LS'R$$
$$S' \to aS'a \mid bS'b \mid T$$
$$L \to X\# \mid \epsilon$$
$$R \to \#X \mid \epsilon$$
$$T \to \#X\# \mid \# \mid C \mid \epsilon$$
$$X \to CX \mid \#X \mid \epsilon$$
$$C \to a \mid b$$

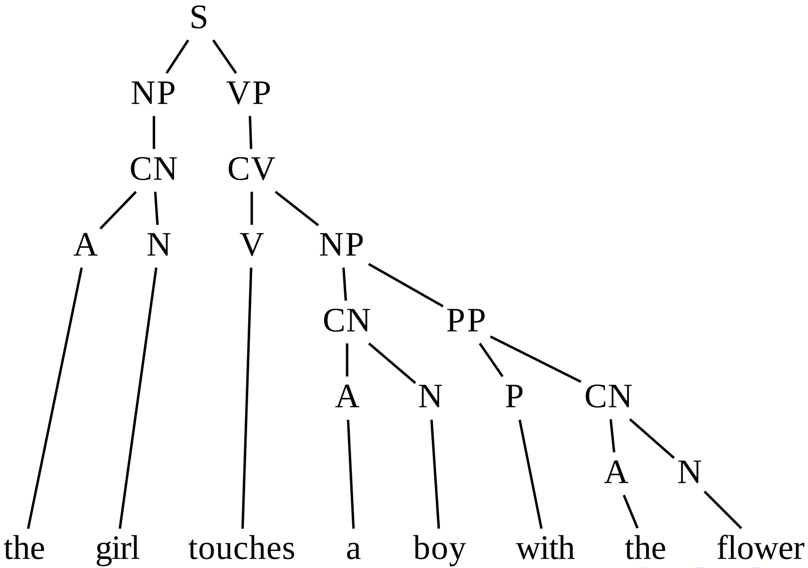
Hw5 Problem4

第一種：女孩碰有花的男孩

S $\Rightarrow$ NP VP $\Rightarrow$ CN VP $\Rightarrow$ A N VP $\Rightarrow$ the N VP $\Rightarrow$ the girl VP $\Rightarrow$
the girl CV $\Rightarrow$ the girl V NP $\Rightarrow$ the girl touches NP $\Rightarrow$
the girl touches CN PP $\Rightarrow$ the girl touches A N PP $\Rightarrow$
the girl touches a N PP $\Rightarrow$ the girl touches a boy PP $\Rightarrow$
the girl touches a boy P CN $\Rightarrow$ the girl touches a boy with CN $\Rightarrow$
the girl touches a boy with A N $\Rightarrow$
the girl touches a boy with the N $\Rightarrow$
the girl touches a boy with the flower

Hw5 Problem4

第一種：女孩碰有花的男孩



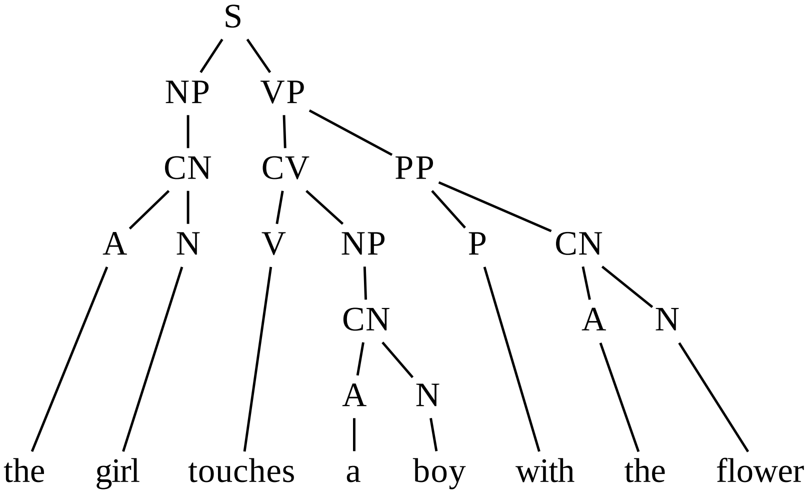
Hw5 Problem4

第二種：女孩用花來碰男孩

$\mathrm{S} \Rightarrow \mathrm{NP\ VP} \Rightarrow \mathrm{CN\ VP} \Rightarrow \mathrm{A\ N\ VP} \Rightarrow$ the $\mathrm{N\ VP} \Rightarrow$ the girl $\mathrm{VP} \Rightarrow$
the girl $\mathrm{CV\ PP} \Rightarrow$ the girl $\mathrm{V\ NP\ PP} \Rightarrow$ the girl touches $\mathrm{NP\ PP} \Rightarrow$
the girl touches $\mathrm{CN\ PP} \Rightarrow$ the girl touches $\mathrm{A\ N\ PP} \Rightarrow$
the girl touches a $\mathrm{N\ PP} \Rightarrow$ the girl touches a boy $\mathrm{PP} \Rightarrow$
the girl touches a boy $\mathrm{P\ CN} \Rightarrow$ the girl touches a boy with $\mathrm{CN} \Rightarrow$
the girl touches a boy with $\mathrm{A\ N} \Rightarrow$
the girl touches a boy with the $\mathrm{N} \Rightarrow$
the girl touches a boy with the flower

Hw5 Problem4

第二種：女孩用花來碰男孩



Hw5 Problem5

設計一個 CFG 生成 $a^ib^jc^k$ 其中 $i = j \vee j = k$
我們可以走兩條路線：$i = j$ 路線或 $j = k$ 路線
對 $i = j$ 路線而言，左半邊要有等量生成的 a 與 b，右邊的 c 則是任意生成
$j = k$ 路線也是以此類推

Hw5 Problem5

$S \to UC \mid AV$
$U \to aUb \mid \epsilon$
$V \to bVc \mid \epsilon$
$A \to aA \mid \epsilon$
$C \to cC \mid \epsilon$

解釋這個 CFG 是否為 ambiguous
考慮字串 abc，可以有兩條路線：
$S \Rightarrow UC \Rightarrow aUbC \Rightarrow abC \Rightarrow abcC \Rightarrow abc$
$S \Rightarrow AV \Rightarrow aAV \Rightarrow aV \Rightarrow abVc \Rightarrow abc$

Hw5 Problem6

$$A \rightarrow BAB \mid B \mid \epsilon$$
$$B \rightarrow 01 \mid \epsilon$$

Hw5 Problem6

第一程序：增加新的 start symbol
加上 $S_0 \rightarrow A$
$S_0 \rightarrow A$
$A \rightarrow BAB \mid B \mid \epsilon$
$B \rightarrow 01 \mid \epsilon$

Hw5 Problem6

第二程序：去除 ϵ

去除 $B \to \epsilon$

$S_0 \to A$

$A \to BAB \mid B \mid \epsilon \mid BA \mid AB \mid A$

$B \to 01$

Hw5 Problem6

第二程序：去除 ϵ

去除 $A \to \epsilon$

$S_0 \to A \mid \epsilon$

$A \to BAB \mid B \mid BA \mid AB \mid A \mid BB$

$B \to 01$

Hw5 Problem6

第三程序：去除 unit rule
去除 $A \rightarrow A$
$S_0 \rightarrow A \mid \epsilon$
$A \rightarrow BAB \mid B \mid BA \mid AB \mid BB$
$B \rightarrow 01$

Hw5 Problem6

第三程序：去除 unit rule

去除 $A \to B$

$S_0 \to A \mid \epsilon$

$A \to BAB \mid BA \mid AB \mid BB \mid 01$

$B \to 01$

Hw5 Problem6

第三程序：去除 unit rule

去除 $S \to A$

$S_0 \to BAB \mid BA \mid AB \mid BB \mid 01 \mid \epsilon$

$A \to BAB \mid BA \mid AB \mid BB \mid 01$

$B \to 01$

Hw5 Problem6

第四程序：分割其它 rule

去除 $S_0 \to BAB$ 與 $A \to BAB$

$S_0 \to BC_1 \mid BA \mid AB \mid BB \mid 01 \mid \epsilon$

$A \to BC_2 \mid BA \mid AB \mid BB \mid 01$

$B \to 01$

$C_1 \to AB$

$C_2 \to AB$

Hw5 Problem6

第四程序：分割其它 rule

去除 $S \to 01$、$A \to 01$ 與 $B \to 01$

$S_0 \to BC_1 \mid BA \mid AB \mid BB \mid O_1I_1 \mid \epsilon$

$A \to BC_2 \mid BA \mid AB \mid BB \mid O_2I_2$

$B \to O_3I_3$

$C_1 \to AB$

$C_2 \to AB$

$O_1 \to 0$

$O_2 \to 0$

$O_3 \to 0$

$I_1 \to 1$

$I_2 \to 1$

$I_3 \to 1$