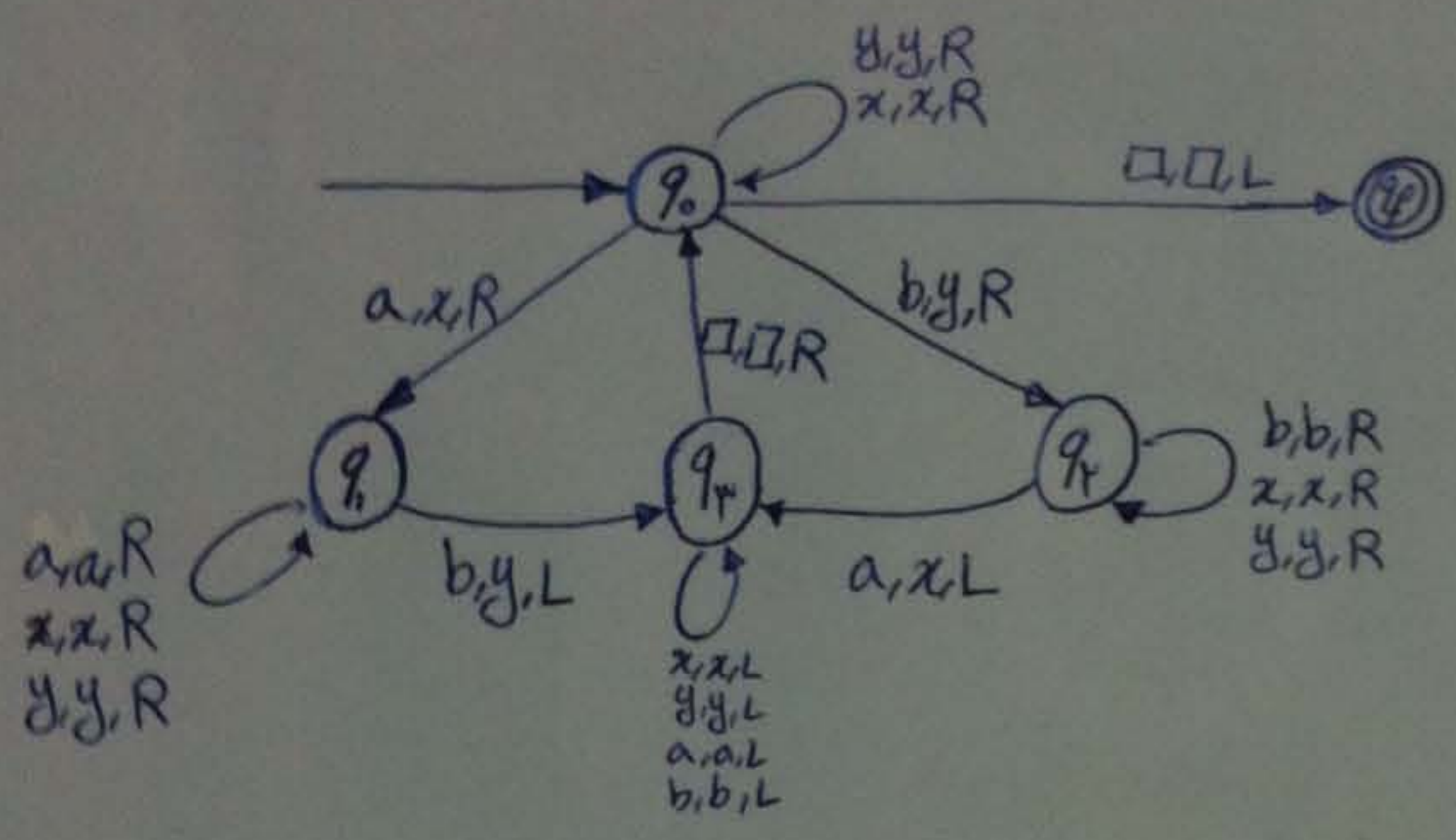


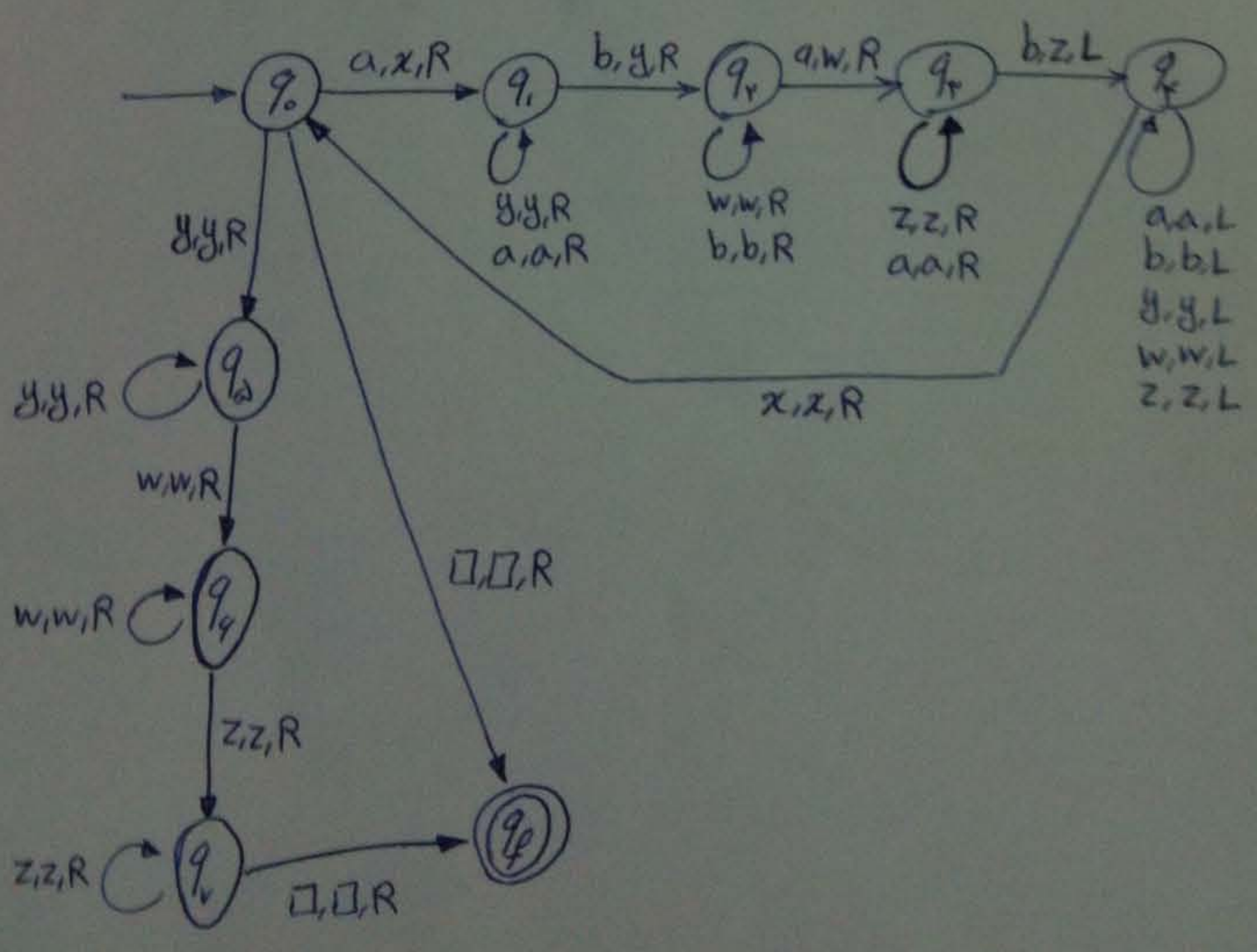
الف)  $L_1 = \{w : n_a(w) = n_b(w)\}$

①



undefined states → fail.  
infinite loop

ب)  $L_2 = \{a^n b^n a^n b^n : n \geq 0\}$

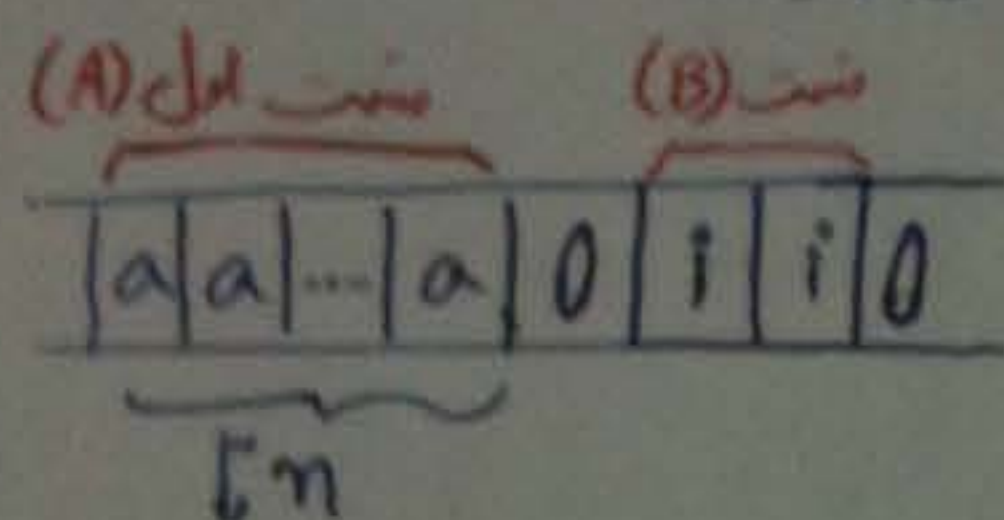




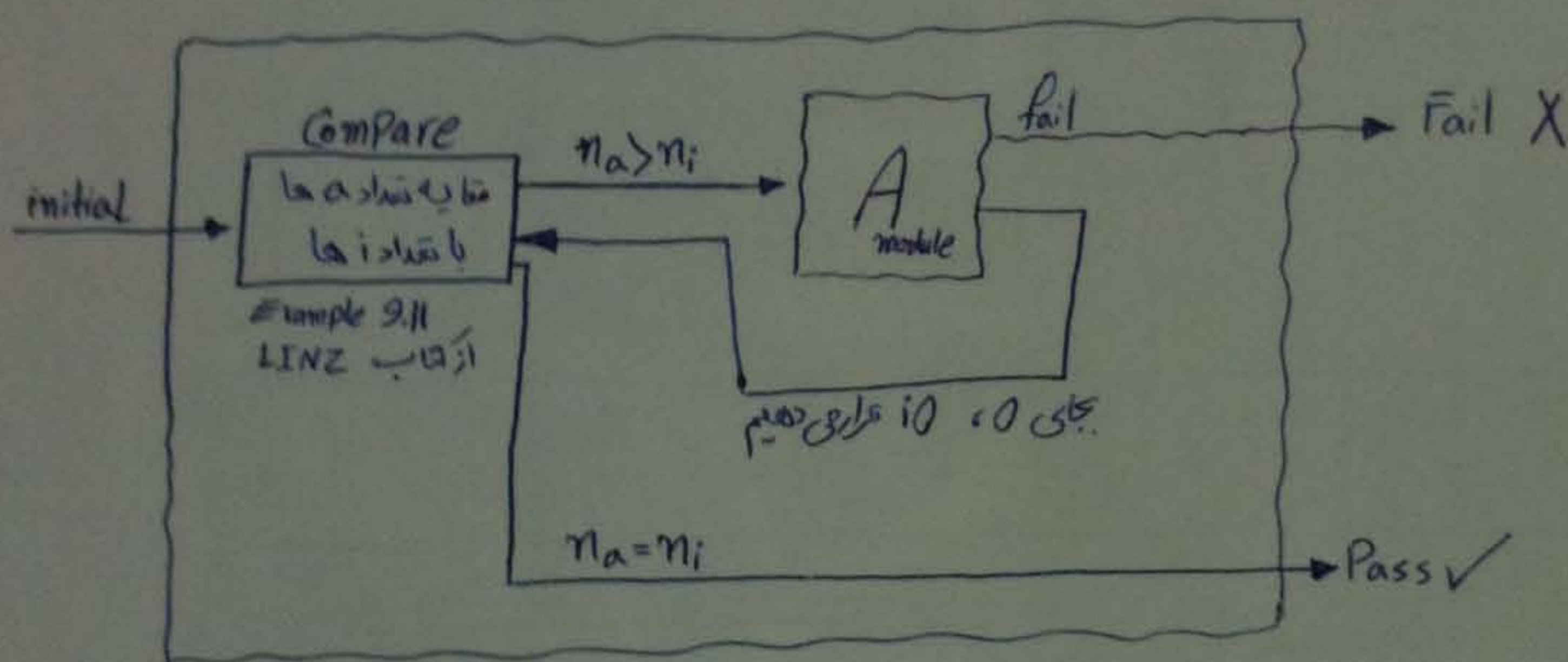
$L_p = \{a^n : n \text{ is a prime number}\}$

ادامه ۱

می دانیم  $n$  اول است اگر بر اعداد  $i$ ،  $2 \leq i < n$  بخش پذیر نباشد



بدین منظور، فوار، را بصورت زیر initial می کنیم:



توضیح A: در این مرحله کارهای زیر را تکرار می کنیم:

- هر  $i$  را به  $i'$  تبدیل می کنیم، و بصورت متناظر یک  $a$  را  $x$  می کنیم
- هر  $i'$  را به  $i$  تبدیل می کنیم، و بصورت متناظر یک  $a$  را  $x$  می کنیم

• در پایان اگر تمام  $a$  ها  $x$  شدند (با 0 بعدش چک می کنیم) ولی تنها  $i$  یا تنها  $i'$

نداشتیم باشیم یعنی  $n \% i \neq 0$  و کار بدست می آید (به تعداد آنها یک واحد اضافه می شود)

• در غیر اینصورت، وضعیت Fail رخ می دهد و  $n$  عدد Prime نیست

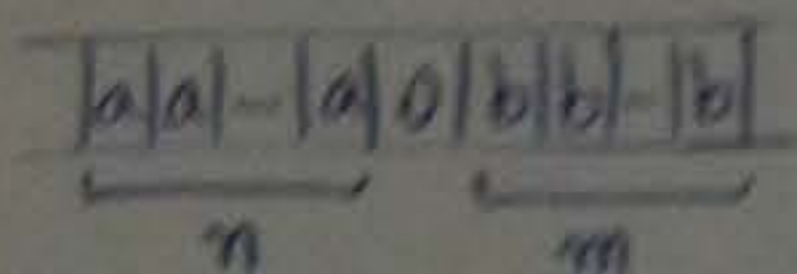


$$L_1 = \{a^n b^m : m = n^2, n \geq 1\}$$

(۲) الف

۱- بجای اولین  $b$ ،  $0$  قرار می دهیم

۲- بعد آخرین  $b$  یک  $b$  اضافه می کنیم و در ردی اولین  $a$  قرار می دهیم



۳- مراحل زیر را تا زمانی که قبل از  $0$ ، تنها یک  $a$  باقی بماند.

(۳-۱)  $a$  ای که در ردی آن قرار دارد را  $k$  می کنیم

(۳-۲)  $0$  را تا چپ ترین  $k$  عقب می بریم

(۳-۳) مرحله زیر را تا  $x$  کردن آخرین  $a$  انجام می دهیم

-  $k$  دیدیم  $L$  می کنیم و یک  $a$  به چپ ترین جای ممکن اضافه می کنیم  
و اگر  $a$  دیدیم  $x$  می کنیم و یک  $a$  به چپ ترین جای ممکن اضافه می کنیم

(۳-۴)  $0$  را تا ردی  $0$  جلو می بریم

(۳-۵) مرحله زیر را تا رسیدن به  $a$  تکرار می کنیم

-  $x$  ها را  $a$  و  $L$  ها را  $k$  می کنیم

(۳-۶)  $0$  را ردی اولین  $a$  بعد از  $k$  جلو می بریم.

۴- هوی  $k$  ها را  $a$  می کنیم و در ردی چپ ترین  $a$  می بریم

۵- مراحل زیر را تا  $x$  شدن خونه ی قبل  $0$  تکرار می کنیم

(۵-۱)  $a$  را  $x$  می کنیم و بدینال اولین  $a$  می گردیم و آن را  $y$  می کنیم (اگر  $y$  باشد  $undefined$ )

(۵-۲)  $0$  را ردی چپ ترین  $a$  می بریم

مقایسه تعداد  $a$  ها و  $b$  ها.

(fail)



۴

$$L_2 = \{ \underbrace{w}_A w^R \underbrace{w}_B : w \in \{a,b\}^+ \}$$

۲. ب) [ایده اولیه :  
خام چرایی]

در این مثال باید از ماشین تورینگ غیر قطعی استفاده کرد. که نسبت در هر مرحله اولین حرف  $w_A$  که  $x$  است را  $x$  می‌کنیم، پس اولین حرف  $w^R$  را در صورت تساوی با  $w_A$   $x$  می‌کنیم و پس بدینال اولین حرف  $w_B$  می‌گردیم و در صورت تساوی با حرف قبلی (در  $w^R, w_A$ ) آن را  $x$  می‌کنیم پس این مرحله را تا پایان رشته، ادامه می‌دهیم.

پ)

$$L_3 = \{ a^i b^j c^k : i \times j = k \text{ and } i, j, k \geq 1 \}$$

در این مثال به ازای هر  $a$  باید  $j$  تا  $c$  داشته باشیم.

- مراحل زیر را تا  $x$  شدن هوی  $a$  تکرار می‌کنیم

۱- اولین  $a$  را  $x$  می‌کنیم

۲- از سایر  $a$  می‌گذریم تا به اولین  $b$  برسیم.

۳- مراحل زیر را تا  $y$  شدن هوی  $b$  ها تکرار می‌کنیم

I- اولین  $b$  را  $y$  کن

II- هدر را آنقدر ~~بست~~ <sup>بست</sup> میرتا اولین  $c$ ، ابتدا کپی را آن را با  $z$  جایگزین کن

III- هدر را تا رسیدن به چپ‌ترین  $b$  عقب ببر

۴- هوی  $y$  ها را  $b$  کن

۵- هدر را تا است چپ‌ترین  $a$  عقب ببر



۵

۳

ناتمامی: مسعود جلیلی

الف) 
$$\left\{ \begin{array}{l} S \rightarrow abaa \mid aAbaa \\ aAb \rightarrow aaABbb \mid ab \\ Bb \rightarrow bB \\ bBaa \rightarrow baaaa \end{array} \right.$$

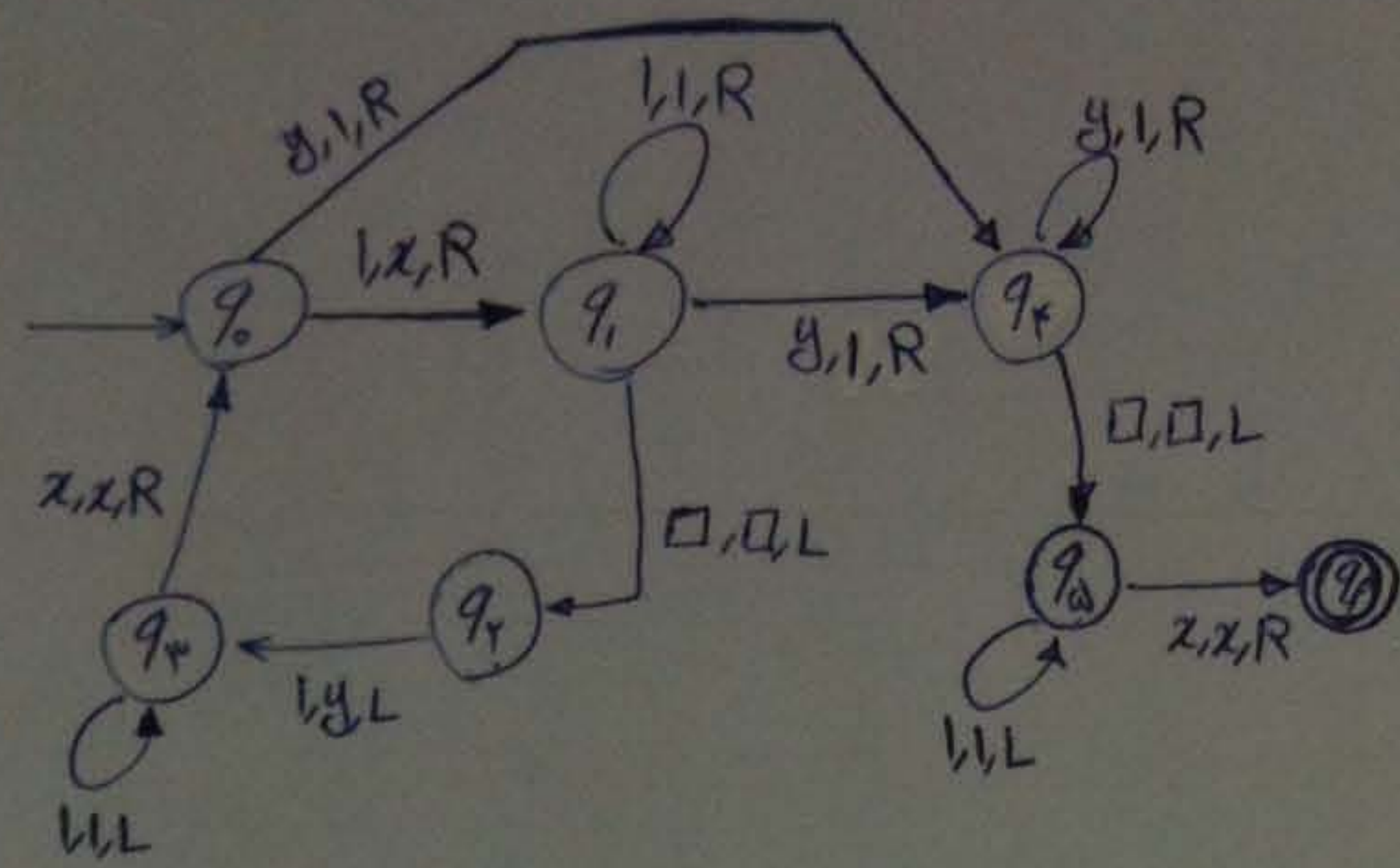
ب) 
$$\left\{ \begin{array}{l} S \rightarrow \#aka\$ \mid \#bkb\$ \\ K \rightarrow aKA \mid bKB \mid \lambda \\ Aa \rightarrow aA \\ Ab \rightarrow bA \\ A\$ \rightarrow a\$ \\ Ba \rightarrow aB \\ Bb \rightarrow bB \\ B\$ \rightarrow b\$ \\ \#a \rightarrow a\# \\ \#b \rightarrow b\# \\ \#\$ \rightarrow \lambda \end{array} \right.$$

ج)

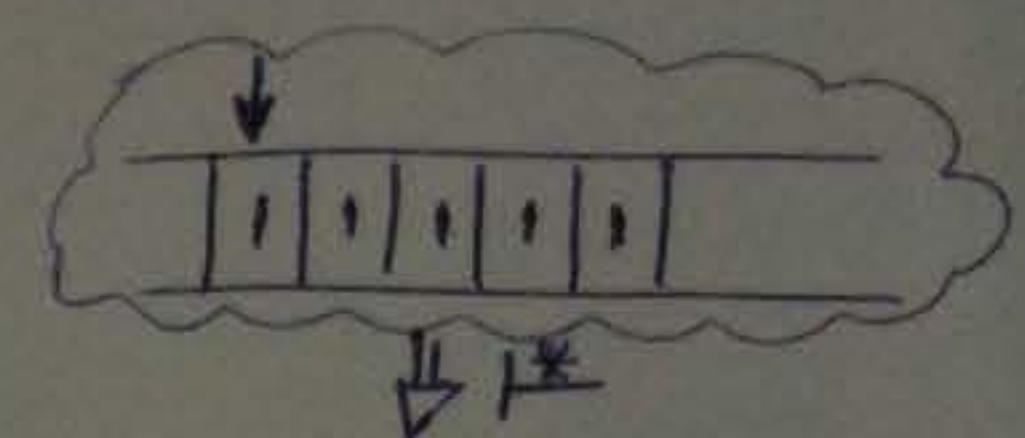


9

(الف)



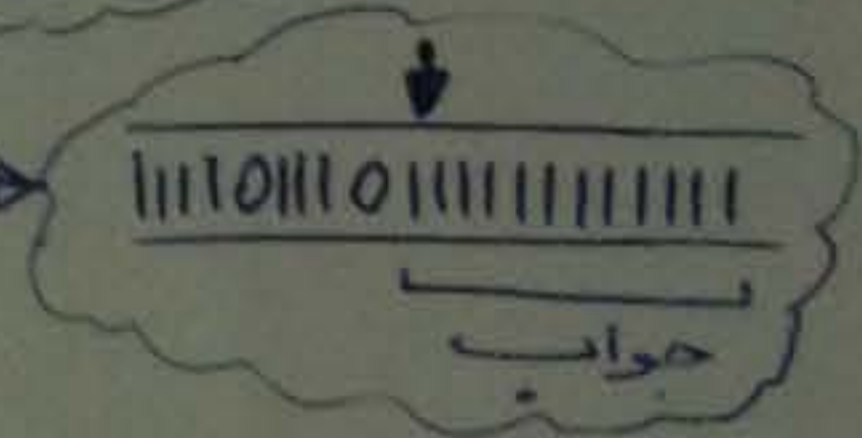
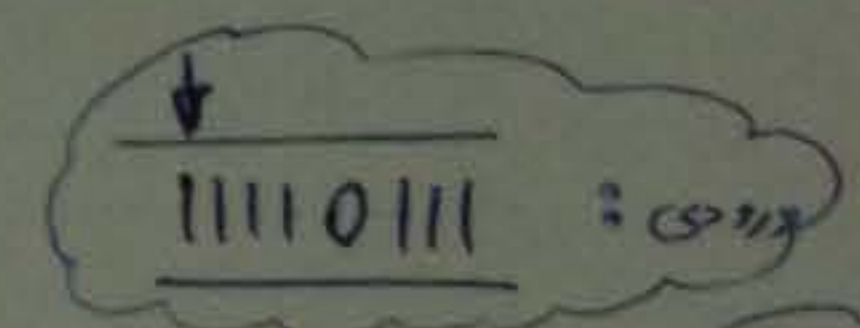
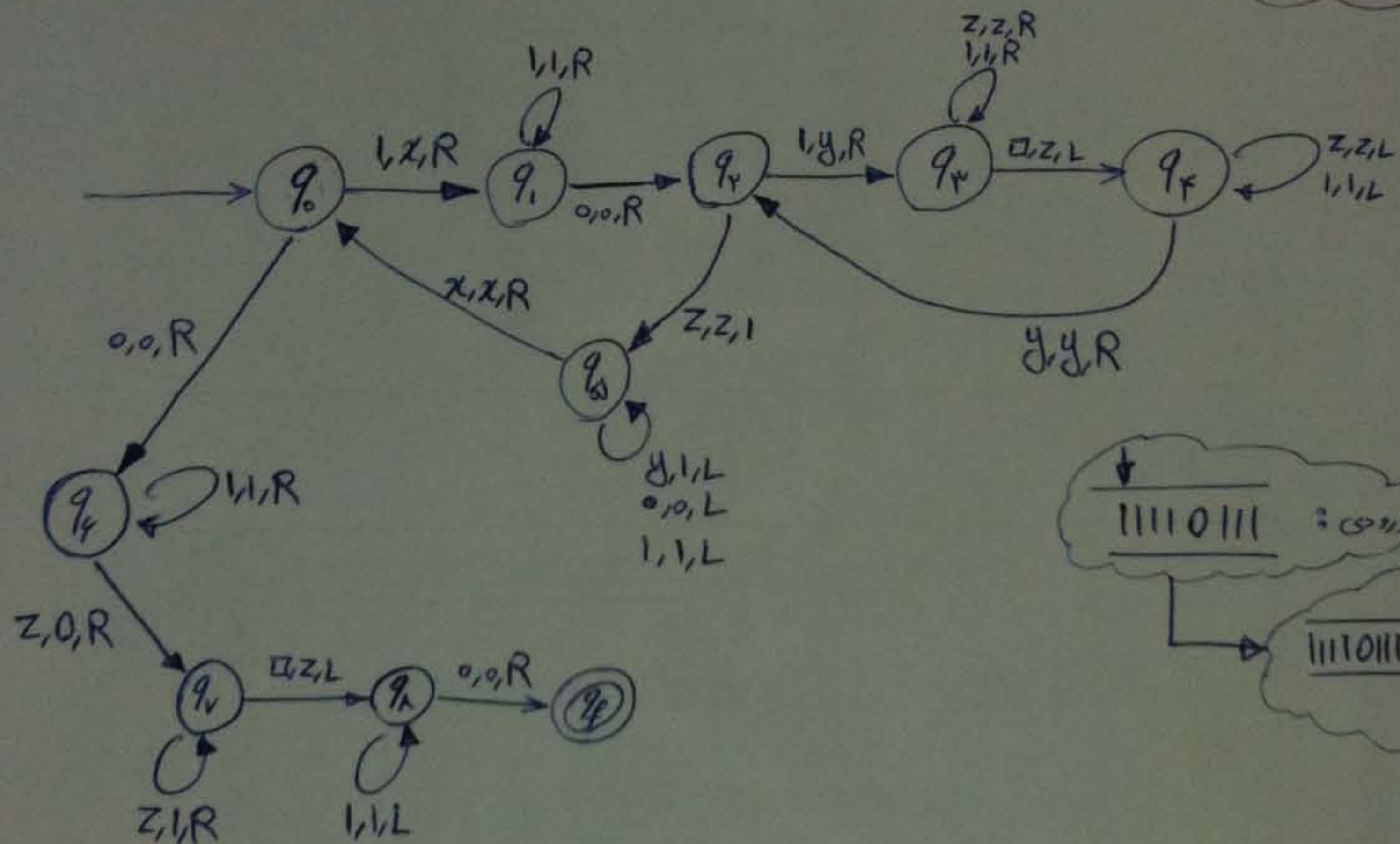
(3)   
 :   
 with stay option



final state:   

x	z	x	1	1
---	---	---	---	---

(ب)



(ج)

