

# به نام خدای رنگین کمان



## مستند فاز سوم پروژه‌ی درس نظریه‌ی زبان‌ها و ماشین‌ها

استاد: سرکار خانم دکتر اژه‌ای

اعضای گروه:

امیرعلی لطفی

محمد کاظم هرندی

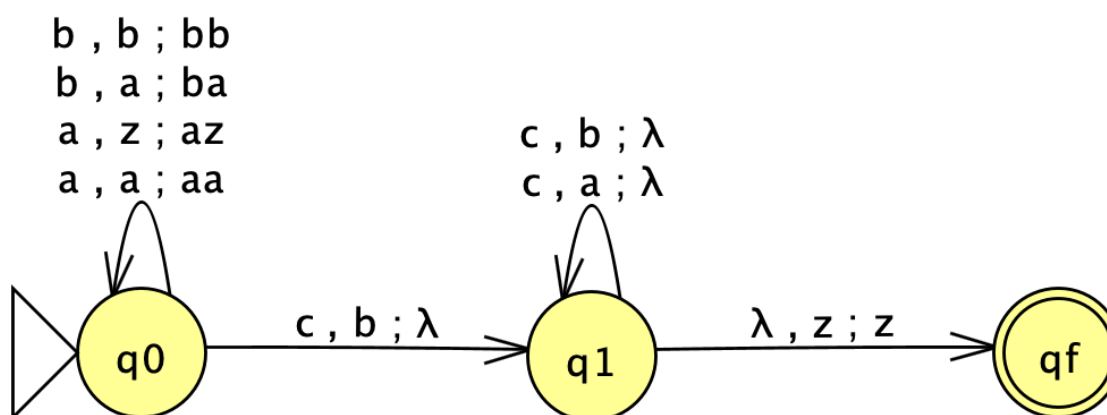
صابر سبزی

بهار ۱۴۰۲

در فاز گذشته با نحوه کار کردن با برنامه jFlap آشنا شدیم. در این فاز تنها با ساخت و نمایش مثال‌های دیگری از مفاهیم باقی‌مانده درس را کامل می‌کنیم.

## زبان‌های مستقل از متن - ساخت اتوماتای پشته‌ای

در اولین مثال یک اتوماتای پشته‌ای ساده ساخته شده است که شکل آن را در تصویر زیر مشاهده می‌کنید:



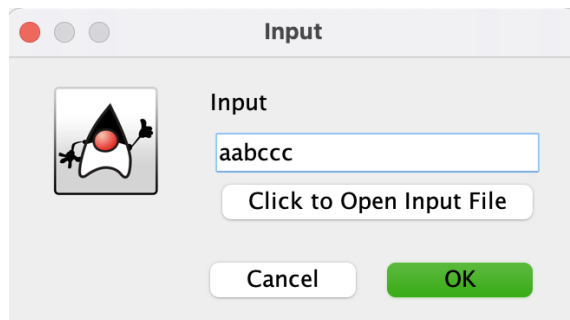
این اتوماتا از ۳ وضعیت تشکیل شده و یک وضعیت نهایی دارد. الفبای تشکیل دهنده آن حروف  $a, b, c$  هستند.

$$L = \{a^m b^n c^{(n+m)} \mid m > 0, n > 0\}$$

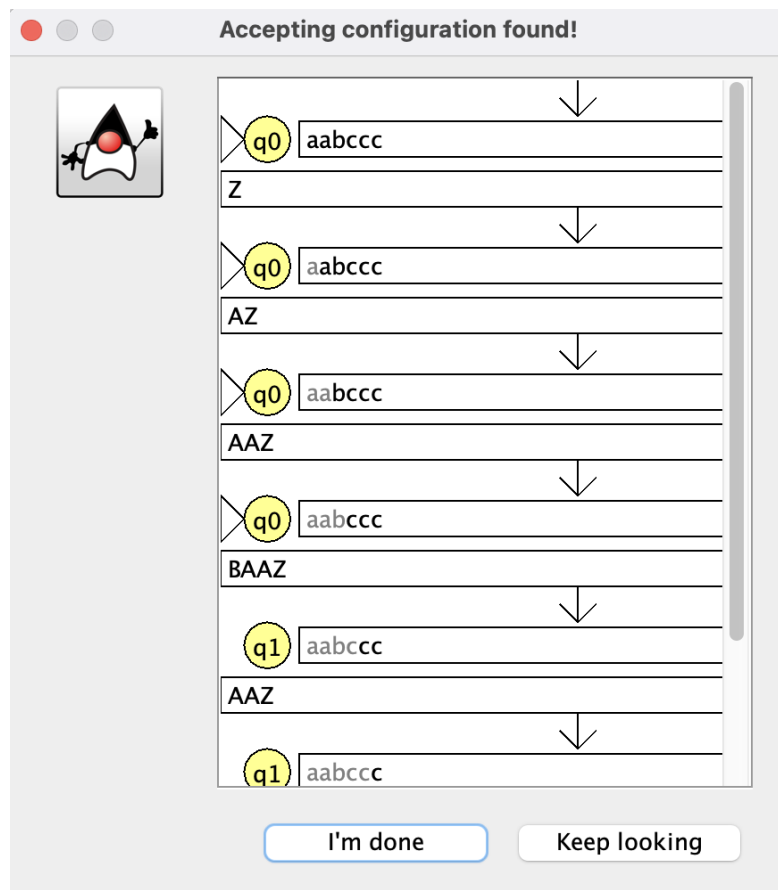
این زبانی است که اتوماتای پشته‌ای بالا پشتیبانی می‌کند.

حال در ادامه با زدن مثال برای تست رشته و پذیرش و عدم پذیرش آن شکل رو بیشتر و بهتر درک می‌کنیم.

Input	Result
aabbc	Reject
aaabbccccc	Accept



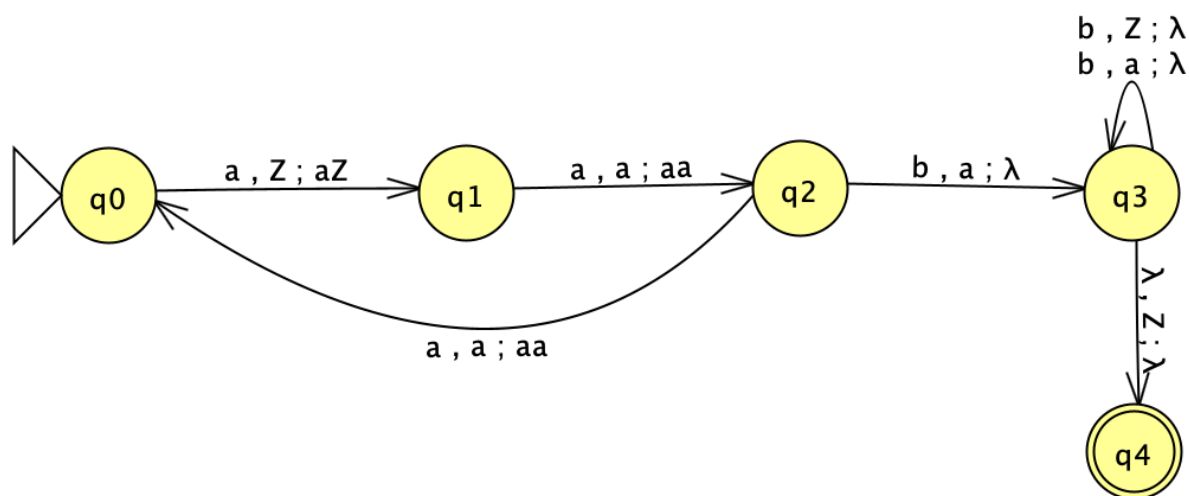
در مثال اول به دلیل کم بودن تعداد **c** رشته مورد پذیرش قرار نگرفت.  
اما در مورد دوم رشته مورد پذیرش قرار می‌گیرد. بعد از تست کردن رشته در برنامه  
JFlap می‌توانید روند پذیرش یا عدم پذیرش را ببینید :



## زبان‌های مستقل از متن - انتقال

### PDA به گرامر مستقل از متن

در ادامه می‌خواهیم یک اتوماتای پشته ای را به یک گرامر مستقل از متن تبدیل کنیم. در شکل زیر اتوماتای مورد نظر را مشاهده می‌کنیم:



اتوماتای پشته‌ای بالا از ۵ وضعیت تشکیل شده است که از زبان زیر پشتیبانی می‌کند :

$$L = ( a^{2n}b^n : a,b > 0 )$$

حال با استفاده از نرم‌افزار JFlap می‌خواهیم آن را به گرامر مستقل از متن تبدیل کنیم:

LHS		RHS
S	$\rightarrow$	aHB
C	$\rightarrow$	b
H	$\rightarrow$	aCA
A	$\rightarrow$	b
B	$\rightarrow$	$\lambda$

شکل بالا گرامر نهایی تولید شده را برای ما نمایش می‌دهد در ادامه مسیر و تکامل آن در تصاویر زیر آورده شده است.

(q3aq3)	$\rightarrow$	b
(q0Zq0)	$\rightarrow$	a(q1aq0)(q0Zq0)
(q0Zq0)	$\rightarrow$	a(q1aq1)(q1Zq0)
(q0Zq0)	$\rightarrow$	a(q1aq2)(q2Zq0)
(q0Zq0)	$\rightarrow$	a(q1aq3)(q3Zq0)
(q0Zq0)	$\rightarrow$	a(q1aq4)(q4Zq0)
(q0Zq1)	$\rightarrow$	a(q1aq0)(q0Zq1)
(q0Zq1)	$\rightarrow$	a(q1aq1)(q1Zq1)
(q0Zq1)	$\rightarrow$	a(q1aq2)(q2Zq1)
(q0Zq1)	$\rightarrow$	a(q1aq3)(q3Zq1)
(q0Zq1)	$\rightarrow$	a(q1aq4)(q4Zq1)
(q0Zq2)	$\rightarrow$	a(q1aq0)(q0Zq2)
(q0Zq2)	$\rightarrow$	a(q1aq1)(q1Zq2)
(q0Zq2)	$\rightarrow$	a(q1aq2)(q2Zq2)
(q0Zq2)	$\rightarrow$	a(q1aq3)(q3Zq2)
(q0Zq2)	$\rightarrow$	a(q1aq4)(q4Zq2)
(q0Zq3)	$\rightarrow$	a(q1aq0)(q0Zq3)
(q0Zq3)	$\rightarrow$	a(q1aq1)(q1Zq3)
(q0Zq3)	$\rightarrow$	a(q1aq2)(q2Zq3)
(q0Zq3)	$\rightarrow$	a(q1aq3)(q3Zq3)
(q0Zq3)	$\rightarrow$	a(q1aq4)(q4Zq3)
(q0Zq4)	$\rightarrow$	a(q1aq0)(q0Zq4)
(q0Zq4)	$\rightarrow$	a(q1aq1)(q1Zq4)
(q0Zq4)	$\rightarrow$	a(q1aq2)(q2Zq4)
(q0Zq4)	$\rightarrow$	a(q1aq3)(q3Zq4)

(q1aq0)	→	a(q2aq1)(q1aq0)
(q1aq0)	→	a(q2aq2)(q2aq0)
(q1aq0)	→	a(q2aq3)(q3aq0)
(q1aq0)	→	a(q2aq4)(q4aq0)
(q1aq1)	→	a(q2aq0)(q0aq1)
(q1aq1)	→	a(q2aq1)(q1aq1)
(q1aq1)	→	a(q2aq2)(q2aq1)
(q1aq1)	→	a(q2aq3)(q3aq1)
(q1aq1)	→	a(q2aq4)(q4aq1)
(q1aq2)	→	a(q2aq0)(q0aq2)
(q1aq2)	→	a(q2aq1)(q1aq2)
(q1aq2)	→	a(q2aq2)(q2aq2)
(q1aq2)	→	a(q2aq3)(q3aq2)
(q1aq2)	→	a(q2aq4)(q4aq2)
(q1aq3)	→	a(q2aq0)(q0aq3)
(q1aq3)	→	a(q2aq1)(q1aq3)
(q1aq3)	→	a(q2aq2)(q2aq3)
(q1aq3)	→	a(q2aq3)(q3aq3)
(q1aq3)	→	a(q2aq4)(q4aq3)
(q1aq4)	→	a(q2aq0)(q0aq4)
(q1aq4)	→	a(q2aq1)(q1aq4)
(q1aq4)	→	a(q2aq2)(q2aq4)
(q1aq4)	→	a(q2aq3)(q3aq4)
(q1aq4)	→	a(q2aq4)(q4aq4)
(q2aq3)	→	b
(q3Zq4)	→	λ
(q2aq0)	→	a(q0aq0)(q0aq0)
(q2aq0)	→	a(q0aq1)(q1aq0)
(q2aq0)	→	a(q0aq2)(q2aq0)

LHS		RHS
$(q3aq3)$	$\rightarrow$	$b$
$(q0Zq4)$	$\rightarrow$	$a(q1aq3)(q3Zq4)$
$(q1aq3)$	$\rightarrow$	$a(q2aq3)(q3aq3)$
$(q2aq3)$	$\rightarrow$	$b$
$(q3Zq4)$	$\rightarrow$	$\lambda$

$(q1aq4)$	$\rightarrow$	$a(q2aq4)(q4aq4)$
$(q2aq3)$	$\rightarrow$	$b$
$(q3Zq4)$	$\rightarrow$	$\lambda$
$(q2aq0)$	$\rightarrow$	$a(q0aq0)(q0aq0)$
$(q2aq0)$	$\rightarrow$	$a(q0aq1)(q1aq0)$
$(q2aq0)$	$\rightarrow$	$a(q0aq2)(q2aq0)$
$(q2aq0)$	$\rightarrow$	$a(q0aq3)(q3aq0)$
$(q2aq0)$	$\rightarrow$	$a(q0aq4)(q4aq0)$
$(q2aq1)$	$\rightarrow$	$a(q0aq0)(q0aq1)$
$(q2aq1)$	$\rightarrow$	$a(q0aq1)(q1aq1)$
$(q2aq1)$	$\rightarrow$	$a(q0aq2)(q2aq1)$
$(q2aq1)$	$\rightarrow$	$a(q0aq3)(q3aq1)$
$(q2aq1)$	$\rightarrow$	$a(q0aq4)(q4aq1)$
$(q2aq2)$	$\rightarrow$	$a(q0aq0)(q0aq2)$
$(q2aq2)$	$\rightarrow$	$a(q0aq1)(q1aq2)$
$(q2aq2)$	$\rightarrow$	$a(q0aq2)(q2aq2)$
$(q2aq2)$	$\rightarrow$	$a(q0aq3)(q3aq2)$
$(q2aq2)$	$\rightarrow$	$a(q0aq4)(q4aq2)$
$(q2aq3)$	$\rightarrow$	$a(q0aq0)(q0aq3)$
$(q2aq3)$	$\rightarrow$	$a(q0aq1)(q1aq3)$
$(q2aq3)$	$\rightarrow$	$a(q0aq2)(q2aq3)$
$(q2aq3)$	$\rightarrow$	$a(q0aq3)(q3aq3)$
$(q2aq3)$	$\rightarrow$	$a(q0aq4)(q4aq3)$
$(q2aq4)$	$\rightarrow$	$a(q0aq0)(q0aq4)$
$(q2aq4)$	$\rightarrow$	$a(q0aq1)(q1aq4)$
$(q2aq4)$	$\rightarrow$	$a(q0aq2)(q2aq4)$
$(q2aq4)$	$\rightarrow$	$a(q0aq3)(q3aq4)$
$(q2aq4)$	$\rightarrow$	$a(q0aq4)(q4aq4)$

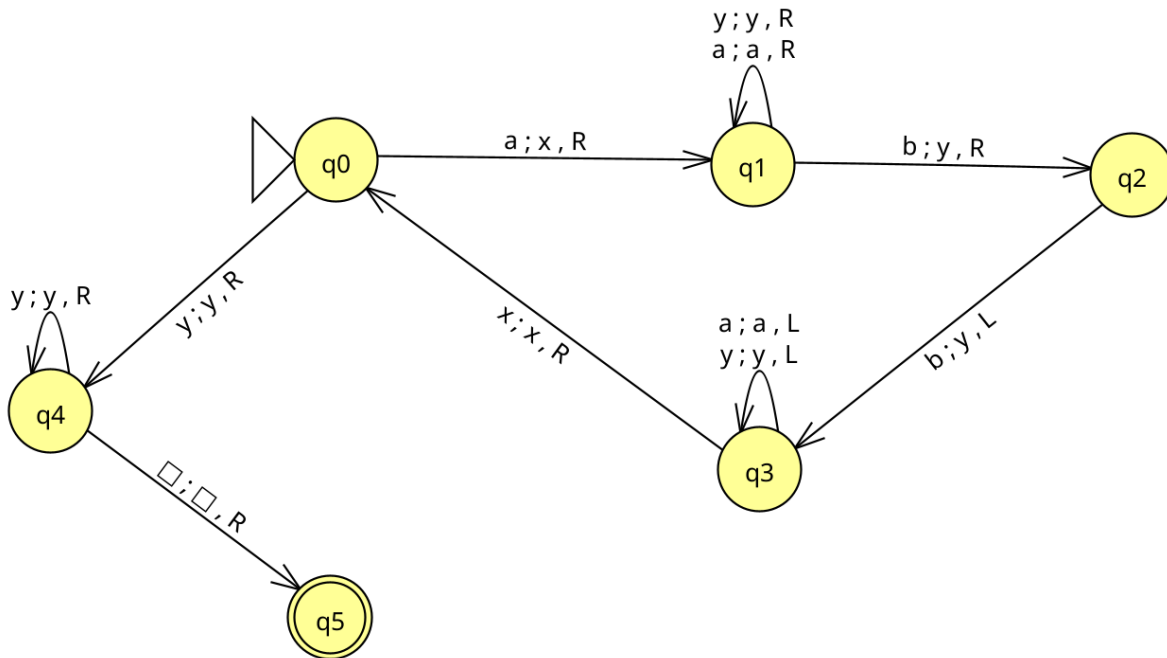
## زبان‌های شمارشی بازگشتی - تست

### ماشین تورینگ (تک نواره)

برای این قسمت زبان زیر در نظر گرفته شده است.

$$L = \{a^n b^{2n} \mid n > 0\}$$

این زبان را ماشین زیر قبول می‌کند:



این ماشین بدین صورت کار می‌کند که به ازای هر سمبل  $a$  که بر روی نوار مشاهده می‌کند،  $2$  بار  $b$  را باید مشاهده کند. این عملیات مشاهده کردن همراه با علامت زدن آن سمبل‌هاست به طوری که  $a$ های مشاهده شده را به  $x$  و  $b$ های مشاهده شده را به  $y$  تبدیل می‌کنیم.

برای تست ماشین ذکر شده، ورودی روبه‌رو را به آن می‌دهیم:

$|aabbabb \rightarrow x|abbbb \rightarrow xa|bbbb \rightarrow xay|bbb \rightarrow xa|yybb \rightarrow x|ayybb \rightarrow |xayybb$   
 $\rightarrow x|ayybb \rightarrow xx|yybb \rightarrow xxy|ybb \rightarrow xxyy|bb \rightarrow xxyyy|b \rightarrow xxyy|yy \rightarrow xxy|yyy$   
 $\rightarrow xx|yyyy \rightarrow x|xyyyy \rightarrow xxyyyy|$

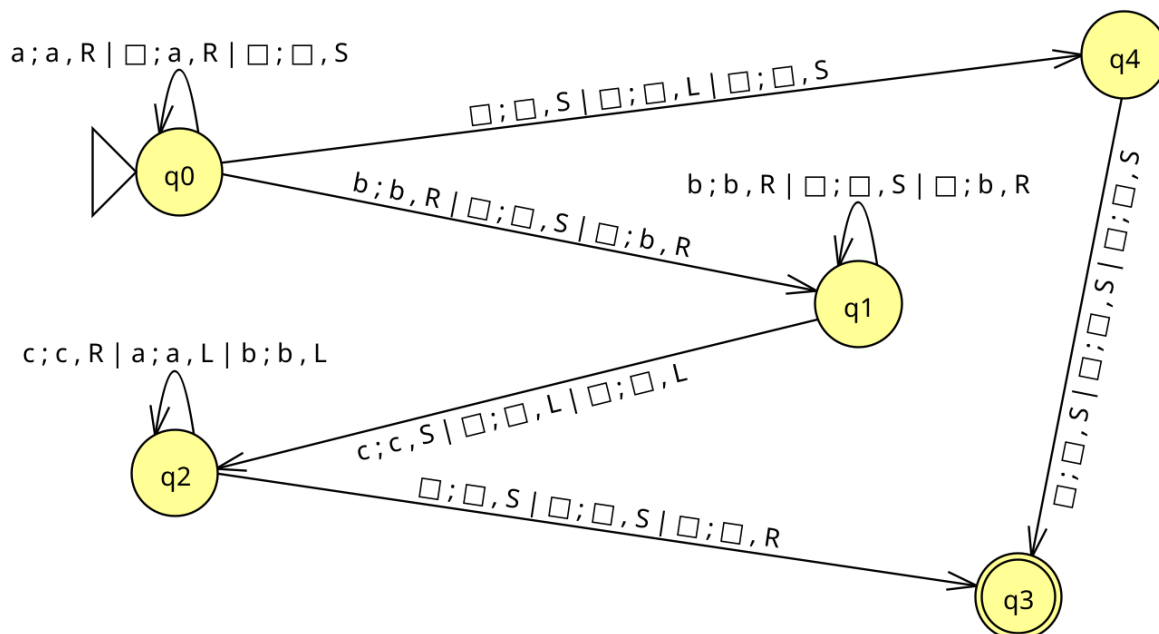
مشاهده می‌شود که رشته بالا در ماشین مذکور قبول می‌شود.



## ماشین تورینگ (چند نواره)

برای مثال این قسمت یک ماشین تورینگ ۳ نواره در نظر گرفته شده است که زبان زیر را می‌پذیرد:

$$L = \{a^n b^n c^n \mid a, b, c > 0\}$$



این ماشین بدین صورت کار می‌کند که هر نوار تعداد یک سمبل به خصوص را می‌شمارد. یعنی در اینجا نوار اول، تعداد  $c$ ها، نوار دوم تعداد  $a$ ها و نوار سوم تعداد  $b$ ها را می‌شمارد. اگر تعداد این سمبل‌ها با هم برابر نباشد، ماشین تورینگ در یکی از حالت‌های غیرپایانی متوقف می‌شود. برای نمونه چند مثال به این ماشین داده می‌شود:

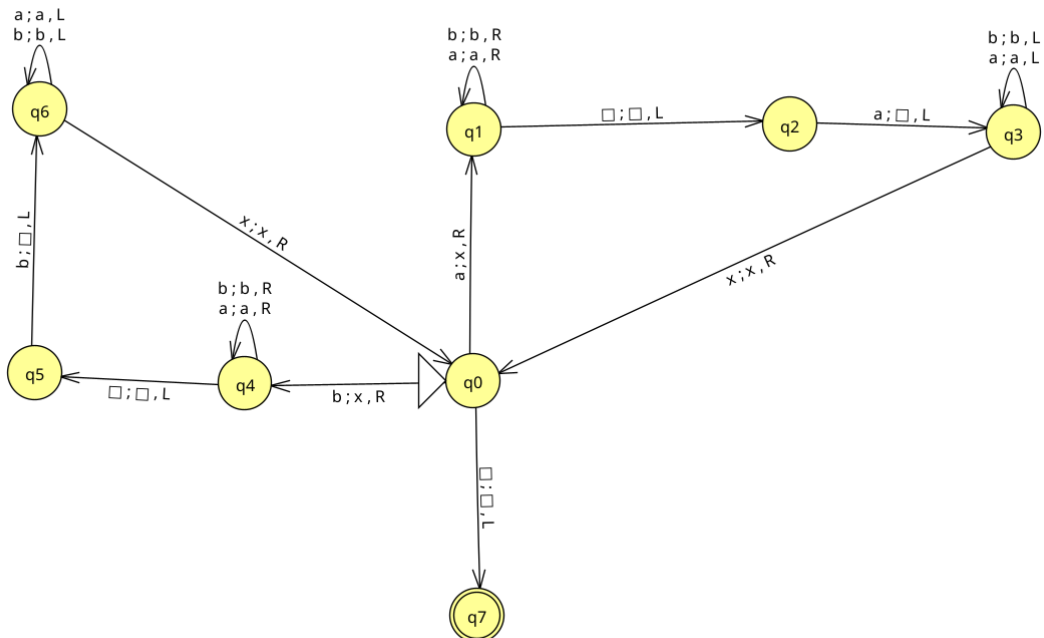
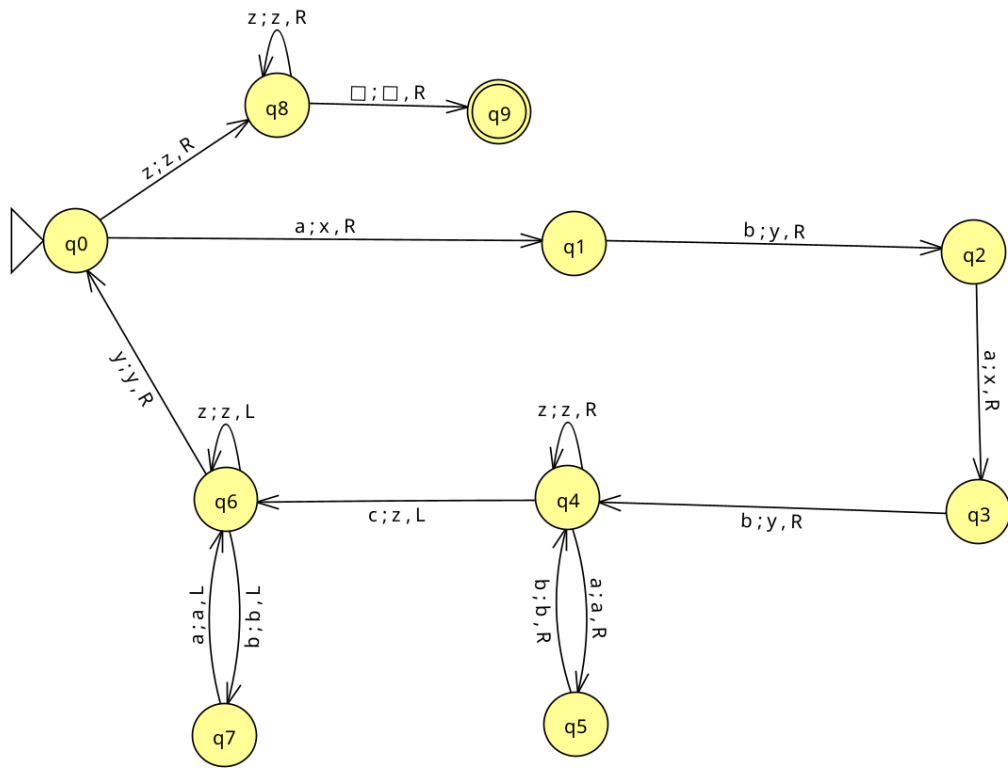
Input 1	Input 2	Input 3	Result
aabbcc			Accept
aaabbbccc			Accept
abc			Accept
aaaaa			Reject
bbbbaaccc			Reject

## ماشین تورینگ (بلوکی)

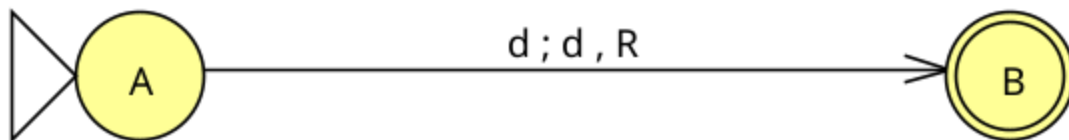
در این قسمت نیز به طراحی یک ماشین تورینگ بلوکی پرداخته می‌شود. این ماشین زبان زیر را می‌پذیرد:

$$L = \{(ab)^{2n} c^n d w w^R \mid n > 0, w \in \{a, b\}^*\}$$

برای ساختن این ماشین به ۲ بلوک نیاز داریم. بلوک اول قسمت اول و بلوک دوم قسمت دوم زبان را تشخیص می‌دهد. تصاویر این ماشین‌ها در پایین‌تر آورده شده است.  
بلوک اول قسمت  $(ab)^{2n} c^n$  را قبول می‌کند و بلوک دوم قسمت  $w w^R$ .



در نهایت این دوبلوک را با هم ذخیره کرده، سپس نتایج را به صورت بلوک در یک ماشین دیگر نمایش می‌دهد.



ارتباط دو بلوک نیز با این تابع انتقال نمایش داده شده است.