

علیرضا کریمی 810101492

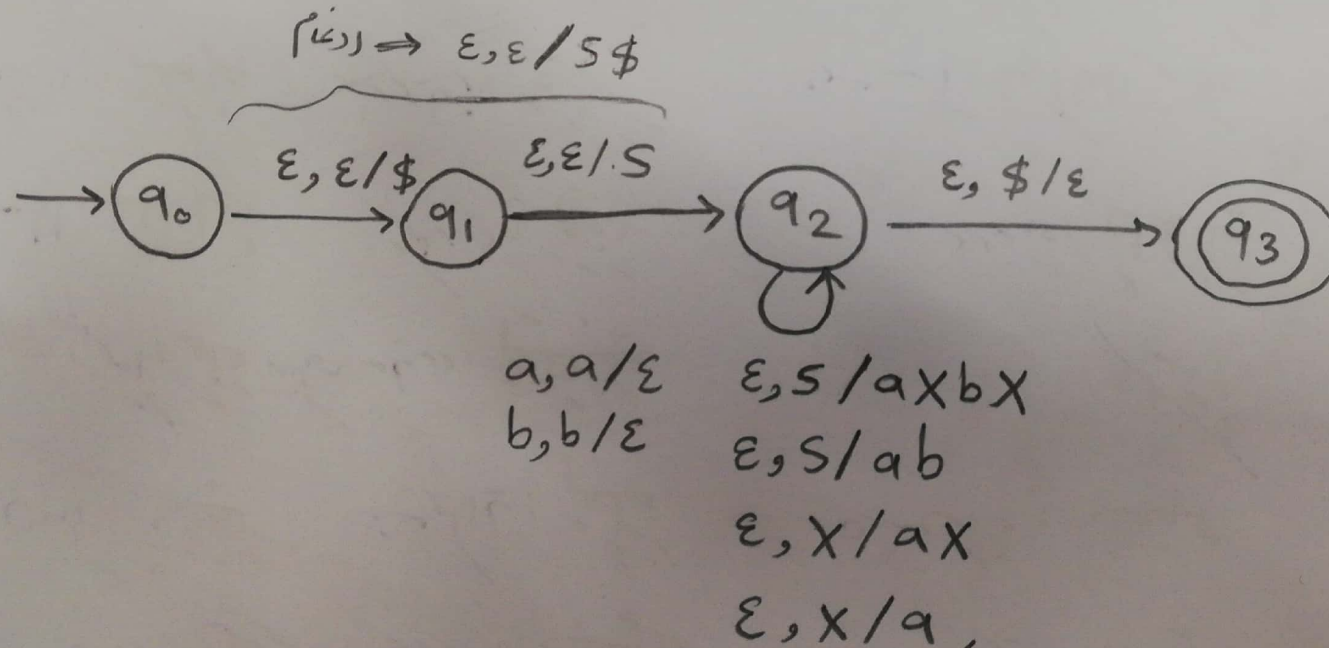
۱ توکن

(۱) از گرامر تبدیل CFG به PDA استفاده می‌کنیم. گرامر زبان که رشته هر به شکل a^*b^* را قبول می‌کند.

$$S \rightarrow aXbX \mid ab$$

$$X \rightarrow aX \mid a$$

به این شکل است:



لحظه بدون تبدیل CFG به PDA هم می‌توان آن را رسم کرد.

(2)

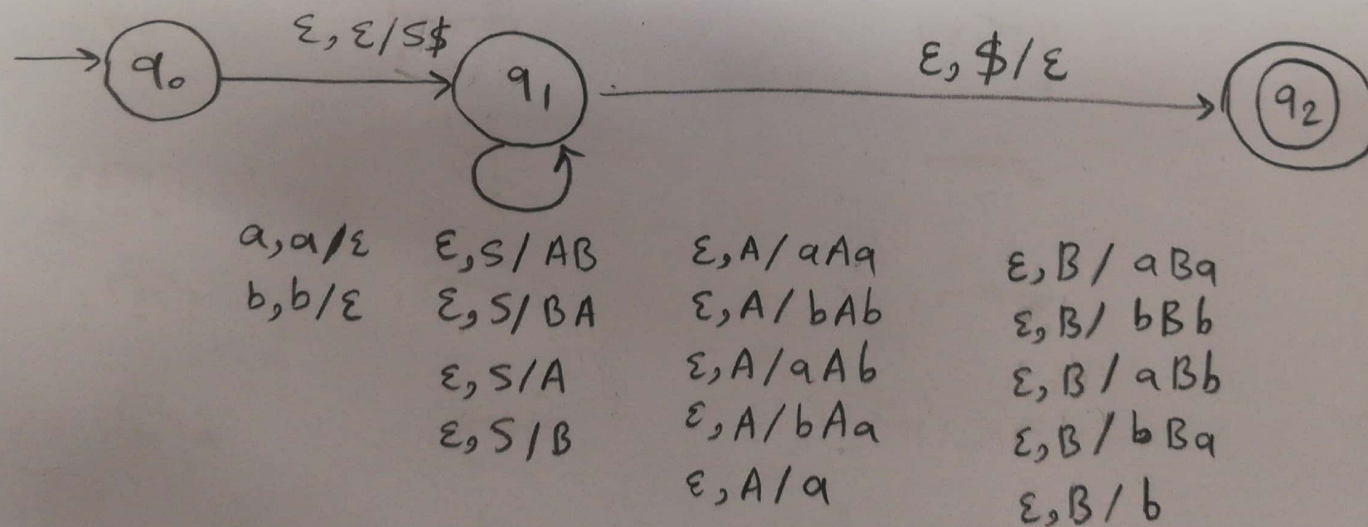
الف) در واقع ما به دنبال مکمل زبان $\{w \mid ww^R \in L\}$ هستیم. برابر این کار باید به دنبال رشته‌ای با طول فرد باشیم. به دنبال رشته‌ای زوج که دو نصف آن یکسان نباشند. پس داریم: گرامر \bar{L} به این شکل است:

برابر رشته با طول فرد \rightarrow همه برابر طول زوج
 $S \rightarrow AB \mid BA \mid A \mid B$

$A \rightarrow aAa \mid aAb \mid bAa \mid bAb \mid a$

$B \rightarrow aBa \mid aBb \mid bBa \mid bBb \mid b$

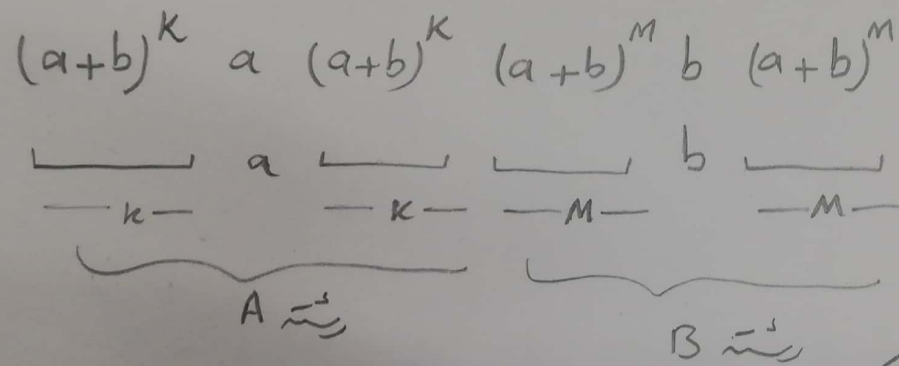
PDA به این شکل می‌شود



توضیح:

دقت کنید که اگر طول رشته ما فرد باشد، قطعاً به صورت ww نیست. پس اگر $S \rightarrow A$ و $S \rightarrow B$ داشته باشیم، چون طول رشته در تولید شده توسط A و B فرد است، این حالت پوشش داده می شود. حال، حالتی را در نظر بگیرید که $S \rightarrow AB$ و $S \rightarrow BA$ بدون کاسن از یکت، فرض کنید که $S \rightarrow AB$. در این صورت رشته ما به

این شکل خواهد شد:



$$m = k + z$$

$$z \in \mathbb{Z}$$

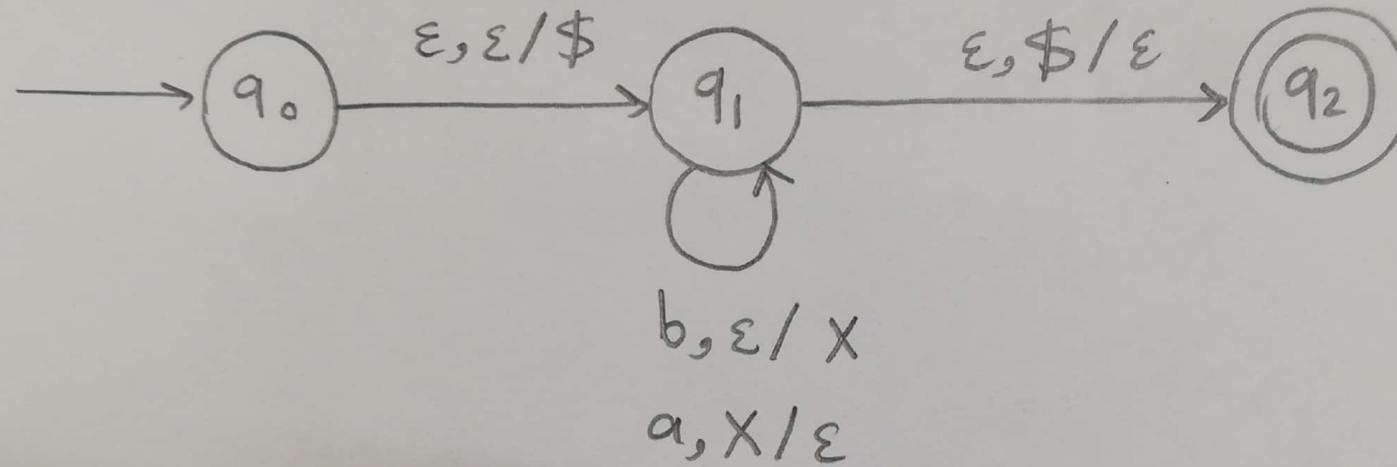
حال رشته حاصل را از میان به دو قسمت تقسیم می کنیم. طول هر قسمت برابر است با:

$$\frac{2k+1 + 2m+1}{2} = \frac{2k+1 + 2(k+z)+1}{2} = 2k+z+1$$

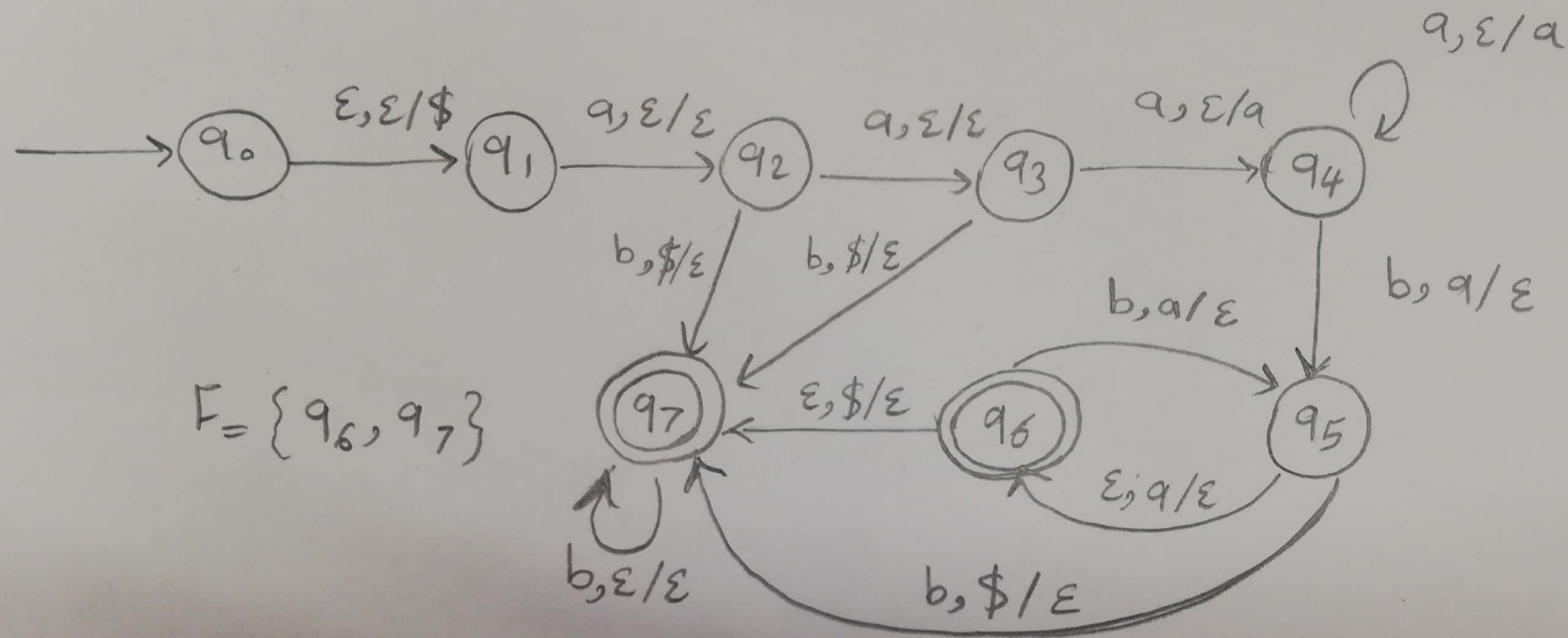
$$\Rightarrow (a+b)^k a (a+b)^{k+z} (a+b)^k b (a+b)^{k+z}$$

همانگونه می بینید، حرف a در قسمت اول در جایگاه $k+1$ ام و حرف b نیز در قسمت دوم در جایگاه $k+1$ ام قرار گرفته است. پس چون هر دو حرف تفاوت نسبت به هر قسمت در جایگاه یکسانی هستند، پس ww در این حالت نیز خواهیم داشت.

ب.



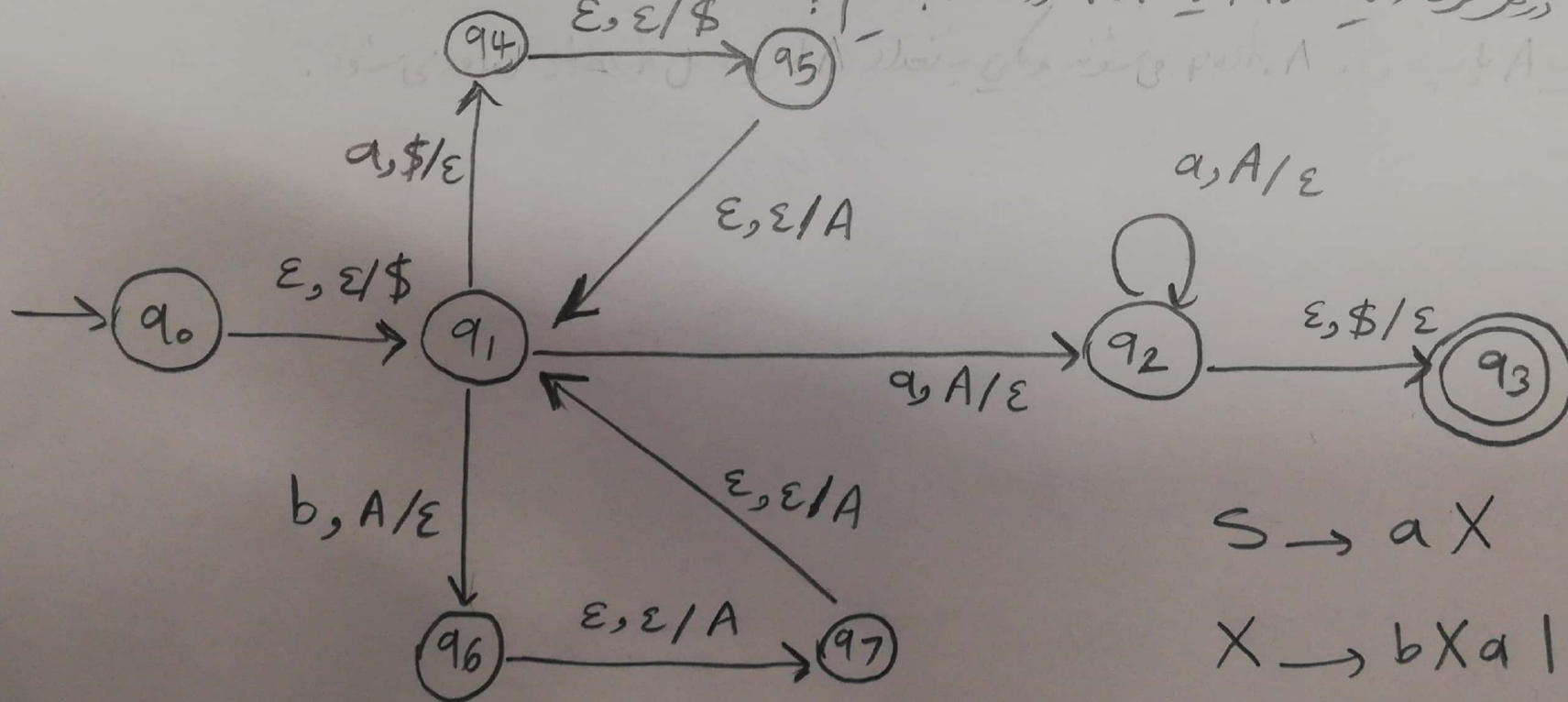
چون تعداد b در مراحل قبلی از a بیشتر است، آن را $push$ می‌کنیم و a را pop می‌کنیم. اگر تعداد a و b برابر شد
تعداد $push$ و pop برابر می‌شود و $\$$ اشیاء است که می‌ماند



توضیح: اگر تعداد a برابر 1 یا 2 باشد، هر مقدار مثبت b می توانیم داشته باشیم (برای همین از q_2 و q_3 یک $\$/\epsilon$ داریم و زنجیر به استیت accept) از $n \geq 3$ به بعد در استک push می کنیم. بعد تقسیم می گیریم که اگر b زوج، به ازای هر b ، دو تا a push می کنیم. اگر داشته باشیم $b^m a^{2m+1}$ ، در q_5 گیر می افتم.

(3) برابر تبدیل کردن PDA به Simplified PDA ، 3 مرحله را انجام می دهیم :
 مرحله اول (یکبار کردن استیت accept (که already میونه استیت accept داریم)

مرحله دوم (خالی کردن stack قبل از accept کردن . لازمی که داریم $\epsilon, \$ / \epsilon$ به استیت آخر نشان می دهیم
 (سنگ خالی شده است)
 مرحله سوم (در هر مرحله یا Push یا Pop داشته باشیم :



$S \rightarrow aX$

$X \rightarrow bXa \mid a$

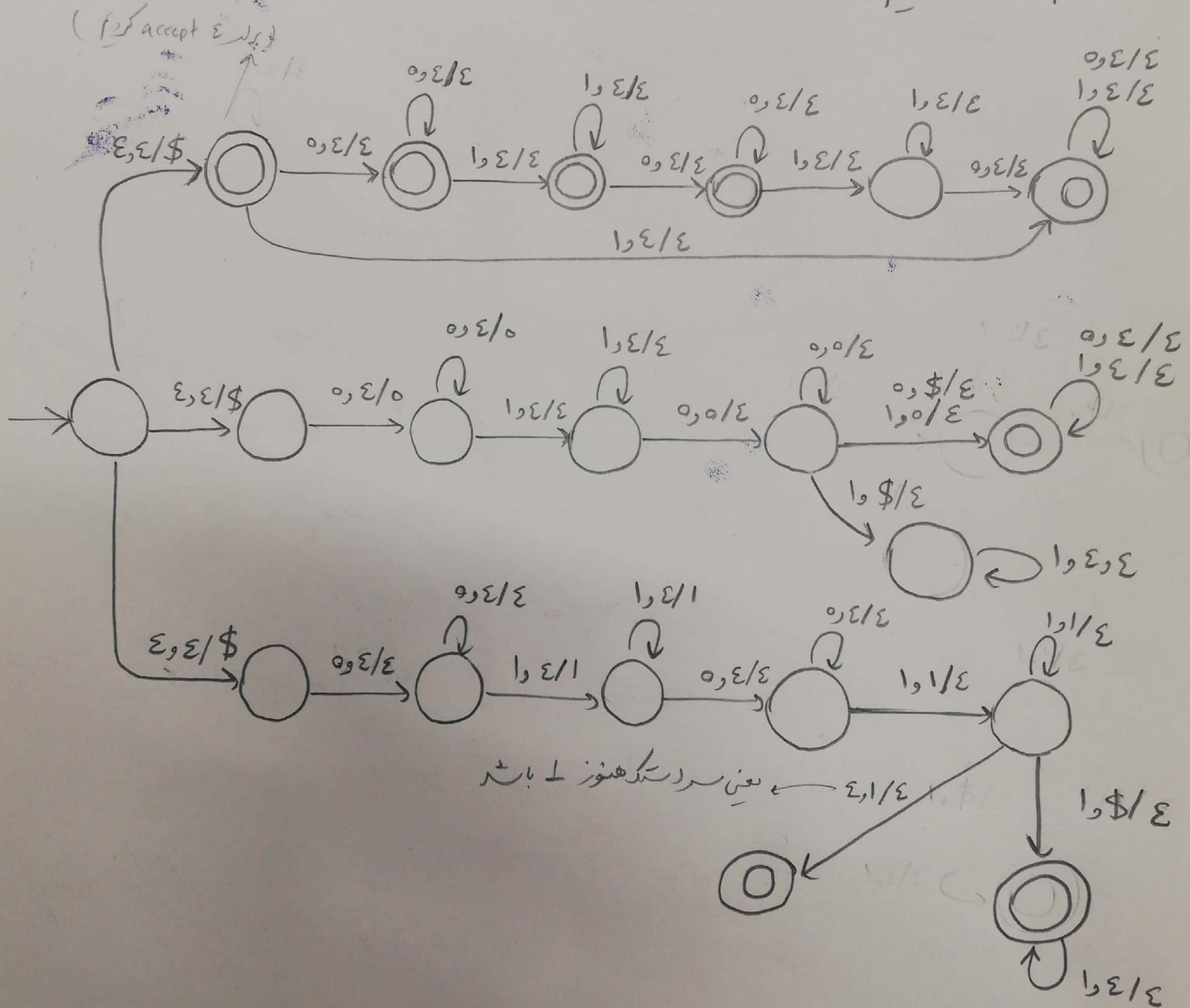
$ab^n a^{n+1} \mid n \geq 0$

(4) خود زبان L، CFG نیست اما قسم آن هست. قسم آن 3 حالت دارد:

حالت اول) کلاً به فرمت $0^k 1^l 0^k$ نباشد. $k, l, j \geq 0$

حالت دوم) رشته باشیم $0^n 1^m 0^n$ به طوری که $a \neq b$ (توان در صفر متفاوت اند)

حالت سوم) رشته باشیم $0^m 1^c 0^m$ به طوری که $c \neq d$ (توان در 1 متفاوت اند)

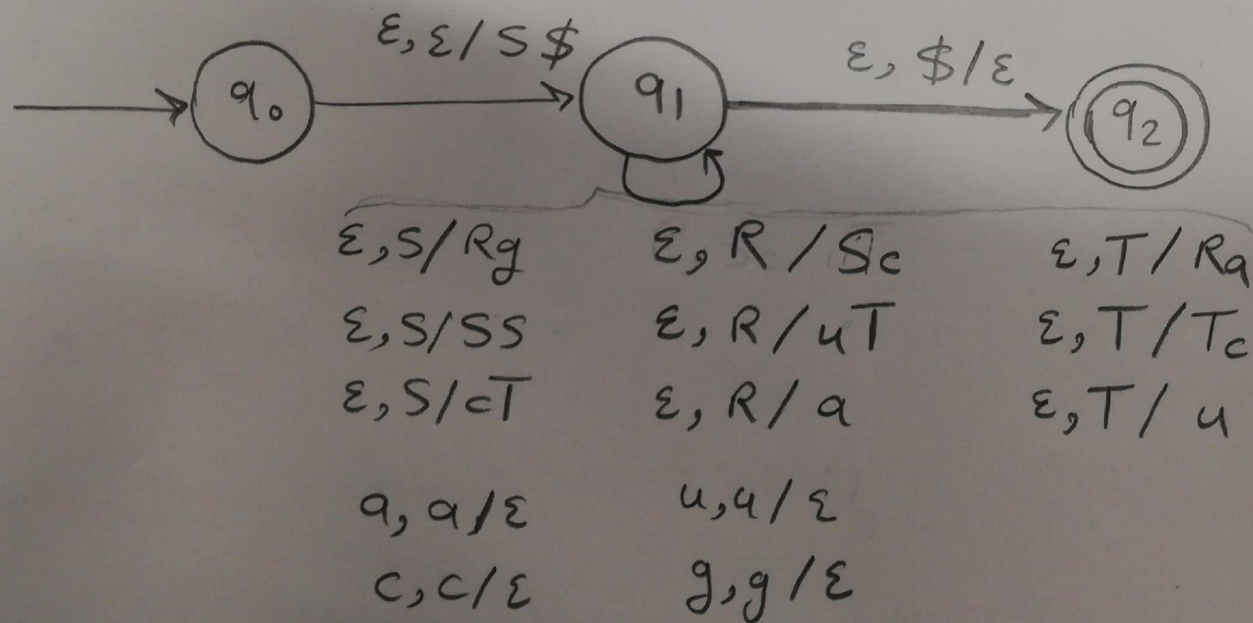


(5)

$$10^n 1^n + 11^{2n} 0^n \mid n > 0 \quad (\text{الف})$$

$$a a^n b b^m \mid n' \leq m \leq 2n \quad (\text{ب})$$

(6)



(7) قدرت هر دو یکسان است؛ چون می توانیم یک state جدید به ازای state جدید به تمام استیت ها بر
 اولیه موجود در PDAI، یک ϵ/ϵ و ϵ بزنیم. حال چون یک استیت شروع داریم، پس PDA داریم.
 لذا اینی که هر PDAI را می توان به این شکل به PDA تبدیل کرد، PDAI، و ویژگی متمایز کننده و قدرت

و بر بر خاص نسبت به PDA، ندارد.
 PDA

