



ساختارهای گسسته

نیم سال دوم ۹۶-۹۷

مدرس: حمید ضرابی زاده

تمرین سری هشتم

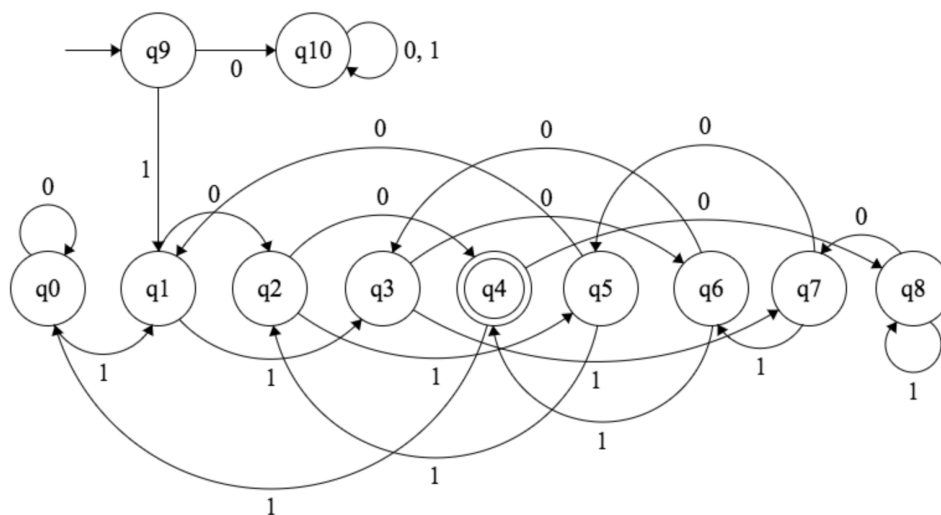
مدل سازی محاسبات

زمان تحویل: ندارد

مسئله ۱. مجذوربازی

یک ماشین حالت متناهی رسم کنید که تمام رشته های دودویی که باقی مانده ی تقسیم عدد متناظر آن ها به ۹ برابر ۴ است را بپذیرد. فرض کنید بیت های عدد به ترتیب از پرارزش به کم ارزش به ماشین داده می شوند. رشته هایی که با صفر شروع می شوند، مورد قبول نیستند.

حل. در شکل زیر حالت qi برای $i < 9$ معادل با باقی مانده ی i به پیمانه ۹ است.



▷

مسئله ۲. همچنان منظم

اگر A زبانی منظم باشد، ثابت کنید به ازای هر زبان B که لزوماً منظم نیست، $C = \{w \mid \exists x : wx \in A, x \in B\}$ منظم است.

حل. زبان A منظم است، پس برای آن DFA وجود دارد، آن را در نظر می گیریم. با استفاده از این DFA و صرفاً با تغییر حالات پایانی آن، برای زبان C یک DFA می سازیم. به ازای هر یک از حالات DFA زبان A مانند q ، اگر رشته ای وجود داشته باشد که عضو زبان B باشد و نیز با شروع از حالت q و ورود آن رشته، به یک حالت پایانی در DFA زبان A برسیم، حالت q یک حالت پایانی در DFA زبان C می شود. بدین ترتیب DFA ای برای زبان C ساخته شده بنابراین این زبان منظم است.

▷

مسئله ۳. عبارت منظم

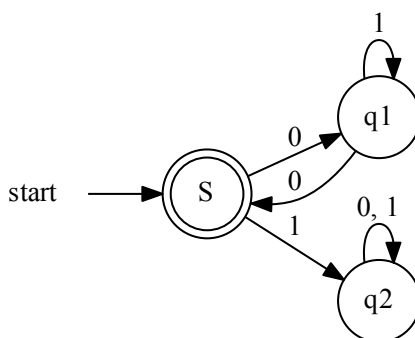
برای عبارت های منظم زیر ماشین حالت متناهی قطعی (DFA) رسم کنید.

الف) $(0^*1^*0)^*$

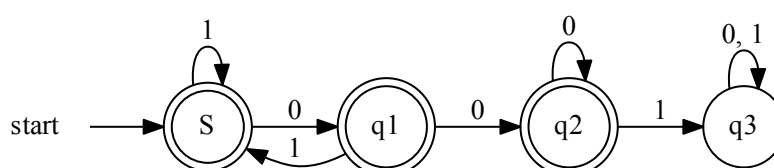
ب) $(0^*1)^*0^*$

حل.

الف)



ب)



▷

مسئله ۴. بی‌نظمی محرز

ثابت کنید زبان $A = \{a^{n!} \mid n \geq 0\}$ منظم نیست.

حل. فرض کنید زبان A منظم است، پس DFA ای با k حالت برای آن وجود دارد. بنابراین، DFA رشته‌ی $a^{(1^0 \times k)!}$ را قبول می‌کند. از آنجا که طول این رشته بزرگ‌تر از k است، طبق لم پمپاژ، $j \leq k$ وجود دارد که DFA رشته‌ی $a^{(1^0 \times k)! + j}$ را نیز می‌پذیرد. اما این رشته، عضو زبان A نیست. بنابراین ثابت می‌شود که زبان A منظم نیست. ▷

مسئله ۵. شک

فرض کنید A یک زبان نامنظم است. در این صورت آیا A^* نیز نامنظم است؟ توضیح دهید.

حل. خیر. مثلاً زبان $A = \{a^{n!} \mid n \geq 0\}$ نامنظم است. ولی در اینجا A^* برابر با a^* می‌شود که منظم می‌باشد. ▷

مسئله‌ی ۶. قدرت

ثابت کنید ماشین تورینگ از ماشین حالت‌متناهی قوی‌تر است. یعنی هر زبانی که با ماشین حالت متناهی قابل تشخیص است با ماشین تورینگ نیز قابل تشخیص است.

حل. باید ثابت کنیم که برای هر زبانی مانند A که DFA ای برای آن وجود دارد، ماشین تورینگ نیز وجود دارد. از روی DFA ماشین تورینگ را به این صورت می‌سازیم که به ازای هر انتقال بین دو حالت در DFA ، یک انتقال بین دو حالت ماشین تورینگ داریم که در این انتقال نوار را تغییر نداده و کلاhek را یک خانه به راست منتقل می‌کنیم. به این ترتیب ماشین تورینگ ذکر شده دقیقاً زبان DFA اولیه را قبول می‌کند. ▽

مسئله‌ی ۷. تصمیم

فرض کنید زبان L و متمم آن (یعنی $L' = \Sigma^* - L$) هر دو توسط ماشین تورینگ قابل تشخیص باشند. ثابت کنید ماشین تورینگی وجود دارد که زبان L را تصمیم می‌گیرد.

حل. فرض کنید ماشین تورینگ T زبان L ، و ماشین تورینگ T' زبان L' را تشخیص می‌دهد. اکنون با استفاده از این دو ماشین، ماشین تورینگ R را به گونه‌ای می‌سازیم که زبان L را تصمیم بگیرد. به طور کلی R به این صورت کار می‌کند که اجرای دو ماشین T و T' را بر روی رشته‌ی ورودی به صورت هم‌رند شبیه‌سازی می‌کند و با توجه به این که این ماشین‌ها رشته‌ی ورودی را می‌پذیرند یا نمی‌پذیرند، تصمیم می‌گیرد که رشته‌ی ورودی در زبان L هست یا نه. ماشین R در آغاز رشته‌ی ورودی w را بر روی نوار خود دارد. این ماشین ابتدا یک گام از ماشین T را بر روی ورودی w اجرا می‌کند. سه حالت ممکن است رخ دهد:

۱. ماشین T به حالت پذیرش برود: در این صورت، رشته‌ی w در زبان L است و بنابراین، ماشین R هم به حالت پذیرش می‌رود.

۲. ماشین T به حالت رد برود: در این صورت، رشته‌ی w در زبان L نیست و بنابراین، ماشین R هم به حالت رد می‌رود.

۳. هیچ‌کدام از دو حالت قبلی روی ندهند: در این صورت، ماشین R اطلاعات مربوط به اجرای T بر روی رشته‌ی w شامل حالت کنونی ماشین T و رشته‌ی حاصل از اجرای T بر روی w را بر روی نوار خود ذخیره می‌کند (پیش از همه‌ی داده‌های موجود کنونی بر روی نوار) تا بعداً بتواند اجرای T را دوباره ادامه دهد.

سپس، در صورتی که ماشین R به حالت پذیرش یا رد نرفته بود، یک گام از اجرای ماشین T' بر روی رشته‌ی w را شبیه‌سازی می‌کند. باز هم سه حالت می‌تواند روی دهد:

۱. ماشین T' به حالت پذیرش برود: در این صورت، $w \in L'$ پس $w \notin L$. بنابراین، ماشین R به حالت رد می‌رود.

۲. ماشین T' به حالت رد برود: در این صورت، $w \notin L'$ پس $w \in L$. بنابراین، ماشین R به حالت پذیرش می‌رود.

۳. هیچ‌کدام از دو حالت قبلی روی ندهند: در این صورت، ماشین R اطلاعات مربوط به اجرای T' بر روی رشته‌ی w شامل حالت کنونی ماشین T' و رشته‌ی حاصل از اجرای T' بر روی w را بر روی نوار خود ذخیره می‌کند (پیش از همه‌ی داده‌های موجود کنونی بر روی نوار) تا بعداً بتواند اجرای T' را دوباره ادامه دهد.

پس از اجرای یک گام از T' ، اگر هنوز ماشین R به حالت پذیرش یا رد نرفته بود، ماشین R با خواندن اطلاعات مربوط به آخرین گام اجرا شده از ماشین T ، گام بعدی از اجرای این ماشین را شبیه‌سازی می‌کند. باز هم اگر ماشین

R به حالت پذیرش یا رد نرفته بود، گام بعدی از اجرای ماشین T' را شبیه‌سازی می‌کند. این مراحل تا جایی که R به حالت پذیرش یا رد برود، اجرا می‌شوند. با توجه به حالت‌بندی‌های بالا، روشن است که ماشین R رشته‌ی w را می‌پذیرد، اگر و تنها اگر $w \in L$. اکنون توجه کنید که اگر $w \in L$ ، ماشین T حتماً پس از تعدادی گام متوقف می‌شود و به حالت پذیرش می‌رود. همچنین اگر $w \notin L$ آن‌گاه $w \in L'$ و بنابراین، ماشین T' پس از تعدادی گام متوقف شده و به حالت پذیرش می‌رود. در هر دو حالت، ماشین R متوقف می‌شود و به یکی از حالت‌های پذیرش یا رد می‌رود. پس ماشین R زبان L را تصمیم می‌گیرد.

▷