

به نام خدا



نظریه زبان‌ها و ماشین‌ها- بهار ۱۴۰۱

تمرین شماره ۹

دستیار آموزشی این مجموعه: سامان اسلامی نظری

SamanEslmai78@gmail.com



تاریخ تحویل: ۲۳ آذر (صفحه درس)

۱. ابتدا اثبات می‌کنیم قدرت ماشین تورینگ که هر خانه‌اش را حداکثر دوبار می‌توان تغییر داد، قدرتی برابر با ماشین تورینگ استاندارد دارد.

نوشتن هر کاراکتر روی نوار طبق معمول و نسبت به مکان فعلی نشانگر انجام میشود. هرگاه نیاز به تغییر یک خانه بود، ابتدا تمام کاراکترهای فعلی را در بخش خالی سمت راست نوار می‌نویسیم (انتهای رشته فعلی را می‌توان با سمبلی مانند \$ مشخص کرد). برای اینکار کاراکتر به کاراکتر عملیات کپی را انجام می‌دهیم و هر کاراکتر کپی شده را علامت می‌زنیم. علاوه بر آن موقعیت اصلی نشانگر را نیز از قبل روی نوار مشخص می‌کنیم. در حین کپی کردن کاراکترها قانون تعویض کاراکتر خواسته شده را نیز انجام می‌دهیم و بجای کاراکتر قدیمی که روی نوار نوشته شده بود، کاراکتر جدید را در بخش راستی نوار می‌نویسیم. بدین صورت اثبات می‌شود که قدرت این ماشین تورینگ با ماشین تورینگ استاندارد یکسان است. برای ماشین تورینگ مذکور در صورت سوال (ماشین تورینگ که حداکثر یکبار می‌توان هر خانه‌اش را تغییر داد) نیز به همین روش عمل می‌کنیم. برای هر کاراکتر دو خانه در نظر می‌گیریم؛ یکی برای نوشتن کاراکتر اصلی و دومی را ابتدا خالی می‌گذاریم؛ هرگاه نیاز به تغییر خانه اصلی شد، جایگزین آن را در این خانه می‌نویسیم. بدین صورت این قابلیت بوجود می‌آید که هر خانه را حداکثر دو بار بتوان تغییر داد. ادامه اثبات از طریق الگوریتم مذکور در بند اول کامل می‌شود.

۲. برای اینکار دو راه حل ارائه می‌دهیم:

a. راه حل اول: برای اثبات این موضوع ابتدا عملیات کپی کردن را، طبق قابلیت‌های ماشین مذکور، تعریف می‌کنیم:

برای کپی کردن از ابتدای نوار تا محل نشانگر، ابتدا محل فعلی را علامت گذاری می‌کنیم و بعد به ابتدای نوار بازمی‌گردیم. سپس کاراکتر فعلی (که در اینجا اولین کاراکتر کپی نشده است) را علامت می‌زنیم و آن را در اولین خانه خالی سمت راست نوار می‌نویسیم. دوباره به ابتدای نوار باز می‌گردیم و مراحل قبلی را تکرار می‌کنیم. حال برای عملیات اصلی ابتدا تمام کاراکترها را تا قبل از محل اولیه نشانگر کپی می‌کنیم. سپس آخرین کاراکتر کپی شده را علامت می‌زنیم (محل اصلی این کاراکتر سمت چپ خانه فعلی نشانگر است. دقت شود در اینجا کپی کاراکتر را، که در حال حاضر آخرین کاراکتر در سمت راست نوار است، علامت می‌زنیم) و سپس باقی کاراکترها را در انتهای نوار کپی می‌کنیم. حال در سمت راست نوار دقیقاً همان کاراکترهای نوار را خواهیم داشت با این تفاوت

که محل خانه سمت چپ نشانگر نیز مشخص شده است. اکنون تنها کافی است با یک حرکت به ابتدای نوار و پیمایش به راست به خانه سمت چپ نوار برسیم. واضح و مبرهن است که ادامه فعالیت‌های ماشین روی کاراکترهای کپی انجام خواهد شد. علاوه بر آن نیازی به ذکر این موضوع نیست که کاراکترهای کپی شده جدید توسط یک سمبول مانند \$ از کاراکترهای قدیمی جدا می‌شوند. بدین صورت عملیات حرکت به چپ نیز در این ماشین تعریف شد. بنابراین این ماشین قدرتی برابر ماشین تورینگ استاندارد دارد.

b. راه حل دوم: با مراحل زیر این ماشین را پیاده سازی می‌کنیم:

۱. ابتدا خانه فعلی را علامت‌گذاری می‌کنیم و به اول نوار می‌رویم.

۲. اول نوار را علامت می‌گذاریم و دوباره به اول نوار می‌رویم.

۳. تا زمانی که به خانه علامت‌گذاری شده نرسیده‌ایم، راست می‌رویم. پس از این که به خانه علامت‌گذاری شده رسیدیم، باز هم به راست می‌رویم.

۴. اگر خانه فعلی علامت‌گذاری شده بود، یعنی خانه قبلی خانه مورد نظر است. پس علامت خانه فعلی را برداشته و به اول نوار می‌رویم. پس از آن با پیمایش به راست به خانه علامت‌گذاری شده می‌رسیم.

۵. اگر خانه فعلی علامت‌گذاری نشده بود، آن را علامت‌گذاری می‌کنیم و به اول نوار می‌رویم؛ تا اولین خانه علامت‌دار راست می‌رویم و علامت آن را پاک می‌کنیم و از ابتدای نوار شروع می‌کنیم.

۳. برای اثبات این برابری ابتدا ماشین تورینگ استاندارد را توسط ماشین صفی مذکور شبیه‌سازی می‌کنیم:

برای نشان دادن ابتدای نوار ماشین تورینگ از \$ استفاده می‌کنیم. بنابراین صف ماشین ما بصورت زیر خواهد بود:

ته صف $\leftarrow ax\$y$ → سر صف

صف ما در هر لحظه مقادیر ماشین تورینگ را دارد و سر صف نشان‌دهنده جایی که نشانگر (head) ماشین تورینگ روی آن قرار دارد است. $a \in \Sigma$ عنصری است که نشانگر ماشین تورینگ روی آن قرار دارد. $x \in \Sigma^*$ مقادیر موجود در سمت راست نشانگر روی نوار و $y \in \Sigma^*$ نیز مقادیر موجود در سمت چپ نشانگر روی نوار است. حال یک عمل بنام شیف راست چرخه‌ای برای صف تعریف می‌کنیم و در ادامه از آن برای تکمیل اثبات استفاده می‌کنیم:

فرض کنید محتویات صف بصورت $ab\$c$ می‌باشد. در مرحله اول ابتدا # را enqueue می‌کنیم. سپس به ازای هر عنصر w که آن را در مرحله قبلی enqueue کرده‌ایم، عنصر x را dequeue کرده و بجای آن (w, x) را enqueue می‌کنیم (یعنی هر عنصر را با عنصر چپش در صف جفت می‌کنیم و در صف قرار می‌دهیم. بجز عنصر اول صف که با # جفت می‌شود). برای حفظ کردن آخرین عنصر dequeue شده از استیت کمکی استفاده می‌کنیم. این کار را آنقدر تکرار می‌کنیم تا به اولین عنصر enqueue شده یعنی # برسیم. خروجی این مرحله به صورت زیر خواهد بود:

ته صف $\leftarrow (\#, a), (a, b), (b, \$), (\$, c), (c, \#)$ → سر صف

در مرحله بعدی عناصر را یکی یکی dequeue و سپس enqueue می‌کنیم تا به عنصر (w, x) ای برسیم که $x = \#$. زمانی که این تساوی برقرار بود، w و # را enqueue می‌کنیم و هر عنصری که در سر صف بود را dequeue می‌کنیم.

خروجی این مرحله بصورت زیر خواهد بود:

ته صف $\leftarrow c, \#, (\$, c), (b, \$), (a, b)$ → سر صف

در مرحله سوم و آخر تا زمانی که به # برسیم، به ترتیب عناصر را dequeue می‌کنیم و به ازای هر (w, x) ، مقدار w را enqueue می‌کنیم. خروجی این مرحله بصورت زیر خواهد بود:

ته صف $\leftarrow cab\$$ → سر صف

به زبان ساده، مراحل بالا برای جایگزین کردن هر عنصر با عنصر چپش بود تا بتوانیم یک چرخش برعکس را اجرا کنیم.

حال برای شبیه‌سازی ماشین تورینگ مقدار موجود روی نشانگر را از صف dequeue و سپس:

- اگر ماشین تورینگ با خواندن a مقدار b را می‌نویسد و به راست می‌رود، مقدار b را enqueue می‌کنیم.

- اگر ماشین تورینگ با خواندن a مقدار b را می‌نویسد و به چپ می‌رود، مقدار b را enqueue می‌کنیم و دوبار شیفت دورانی راست را انجام می‌دهیم.

نشان دادیم که ماشین تورینگ استاندارد را توسط ماشین صفی مذکور می‌توان شبیه‌سازی کرد. برای تکمیل اثبات باید عکس این موضوع را نیز نشان دهیم. برای اینکار از یک ماشین تورینگ دو نواره استفاده می‌کنیم؛ یک نوار برای خواندن ورودی و یک نوار برای و دیگری برای نگهداری مقادیر موجود در صف. در ابتدا یک $\$$ روی نوار صف نوشته می‌شود. زمانی که عنصری در صف enqueue می‌شود، ماشین تورینگ اولین خانه خالی روی نوار را یافته و روی آن می‌نویسد. برای dequeue کردن نیز ماشین تورینگ اولین عنصر روی نوار که $\$$ نیست را یافته و آن را با $\$$ جایگزین می‌کند.

۴. از یک ماشین تورینگ دو نواره استفاده می‌کنیم. از نوار دوم برای ذخیره زیررشته‌ای استفاده می‌کنیم که حدس می‌زنیم در رشته تکرار شده.

از ابتدای نوار (از چپ به راست) حروف را پیمایش می‌کنیم و در هر مرحله به طور غیرقطعی تصمیم می‌گیریم خواندن ورودی‌ها را ادامه دهیم و یا ۲۰ کاراکتر بعدی را در نوار دوم کپی کنیم. کپی کردن رشته موجود در طی ۲۰ استیت انجام می‌شود. پس از کپی کردن زیررشته، نشانگر نوار دوم را به ابتدای نوار (منتهی الیه چپ نوار) می‌بریم. سپس در هر مرحله به طور غیرقطعی تصمیم می‌گیریم نوار شماره یک را به سمت راست پیمایش کنیم و یا چک کنیم که آیا ۲۰ کاراکتر بعدی با ۲۰ کاراکتر ذخیره شده روی نوار ۲ برابر است. اگر برابر بودند رشته را قبول می‌کنیم.

۵. برای اینکار یک تابع یک به یک بصورت $f: \mathbb{N} \times \mathbb{N} \rightarrow \mathbb{N}$ ارائه می‌دهیم. تابع [canto pairing](#)، برای اینکار مناسب

می‌باشد. این تابع بصورت $f(x, y) = \frac{1}{2}(x + y + 1)(x + y) + y$ است.

برای شبیه‌سازی این ماشین تنها کافی است هر حرکت نشانگر آن را ابتدا به تابع $f(x, y)$ داده و سپس عدد بدست آمده را در نوار ماشین استاندارد بیابیم.

۶. از روش‌های مختلفی برای تبدیل $\mathbb{N} \times \mathbb{N} \rightarrow \mathbb{N}$ می‌توان استفاده نمود. در اینجا ما از تابع [Cantor pairing](#) استفاده

می‌کنیم. در این جا ما نیاز به یک تابع برای تبدیل $\mathbb{N} \times \mathbb{N} \times \mathbb{N} \rightarrow \mathbb{N}$ نیاز داریم. در ادامه روشی برای این کار ارائه می‌دهیم:

تابع cantor به صورت $f(x, y) = \frac{1}{2}(x + y + 1)(x + y) + y$ است. فرض کنید نشانگر ماشین تورینگ با فضای سه بعدی از خانه فعلی به یکی از ۶ خانه مجاورش حرکت می‌کند. فرض کنید خانه جدیدی که نشانگر این ماشین روی آن قرار می‌گیرد دارای مختصات (a, b, c) است. ابتدا این مختصات را با استفاده از تابع cantor به محلی در دستگاه دو بعدی تبدیل می‌کنیم. برای اینکار دو مولفه اول را توسط تابع به یک مولفه تبدیل می‌کنیم. یعنی (a, b, c) بصورت $(f(a, b), c)$ می‌شود. سپس این دو مولفه را مجدداً به تابع داده و عدد متناظر با آن را روی نوار یک بعدی پیدا می‌کنیم. عدد نهایی به صورت $f(f(a, b), c)$ می‌شود.

بنابراین هرگاه ماشین با فضای سه بعدی، نشانگرش را به خانه‌ای برد، ابتدا مختصات آن خانه را توسط شیوه مذکور به عددی یکتا تبدیل کرده و سپس نشانگر ماشین تورینگ استاندارد را نیز به خانه متناظر با آن عدد می‌بریم.

برای شبیه‌سازی ماشین تورینگ استاندارد توسط ماشین تورینگ سه بعدی نیز می‌توان از معکوس تابع cantor و یا به صورت راحت‌تر فقط از یک نوار یک بعدی در فضای سه بعدی استفاده کنیم.

بنابراین هر دو ماشین را می‌توان توسط دیگری شبیه‌سازی کرده و اثبات می‌شود این دو قدرت یکسانی دارند.

۷.

- a. شبیه‌سازی PDA با دو استک توسط ماشین تورینگ استاندارد به راحتی قابل انجام است. تنها کافی است سر هر استک را با یک سمبول مانند $\#_i$ ، روی نوار ماشین تورینگ نمایش دهیم. هنگامی که می‌خواهیم از استک $\#_i$ عملیات pop را انجام دهیم، ابتدا نوار را پیمایش می‌کنیم تا به $\#_i$ برسیم؛ سپس حرف سمت چپ آن را پاک کرده و $\#_i$ و تمام حروف نوشته شده جلو آن را یک خانه به چپ انتقال می‌دهیم. برای push کردن به استک $\#_i$ نیز ابتدا نوار را پیمایش می‌کنیم تا به $\#_i$ برسیم؛ سپس $\#_i$ و تمام حروف سمت راست آن را یک خانه به راست انتقال داده و سپس در سمت چپ $\#_i$ که حالا خالی است، کاراکتری که می‌خواستیم پوش کنیم را می‌نویسیم.
- دقت شود در یک PDA الفبای ورودی بلافاصله پوش می‌شوند در حالیکه در اینجا ما ابتدا یک عملیاتی را انجام داده و سپس حرف ورودی را روی نوار می‌نویسیم. از آنجا که تعداد حروف الفبا محدود است، برای حفظ کردن الفبای ورودی می‌توان از مجموعه استیت‌ها کمک گرفت.
- حال ماشین تورینگ را با PDA داده شده شبیه‌سازی می‌کنیم. در هر لحظه که نشانگر روی بخشی از نوار ماشین تورینگ قرار دارد، نوار به دو بخش راست و چپ تقسیم می‌شود. بخش چپ را در استک اول و بخش راست را در استک دوم ذخیره می‌کنیم. خانه‌ای که اشاره‌گر نوار به آن اشاره می‌کند همواره بالای استک اول خواهد بود.
- هنگامی که نوار به سمت راست حرکت می‌کند، از استک دوم حرفی را پاپ کرده و آن را در استک اول پوش می‌کنیم. هنگامی که اشاره‌گر به سمت چپ نوار حرکت می‌کند نیز برعکس همین کار را انجام می‌دهیم. اگر روی خانه اشاره‌گر چیزی بنویسیم، تنها کافی است از استک اول پاپ کرده و حرفی که می‌خواهیم را پوش می‌کنیم.
- بدین صورت اثبات شد که PDA با دو استک قدرتی مشابه ماشین تورینگ استاندارد دارد.
- b. برای حالت $i = j$ این موضوع بدیهی است. برای حالتی که $i > j$ باشد نیز اثبات می‌کنیم (حالت $i > j$ نیز فرقی ندارد):

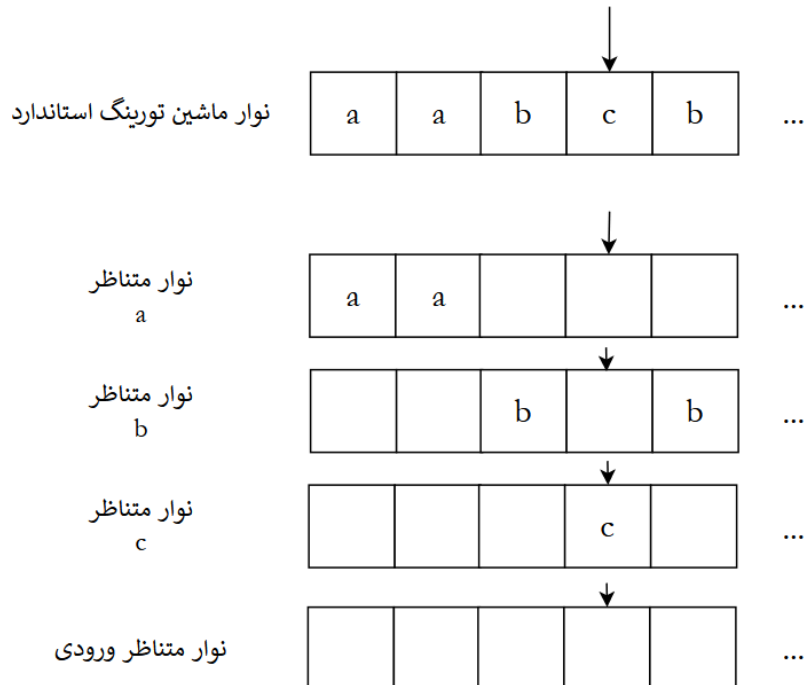
برای اثبات این موضوع تنها کافی است اثبات کنیم قدرت هر PDA با i استک، مشابه PDA با $i + 1$ استک است؛ چرا که $i + 1$ نیز قدرتی مشابه $i + 2$ و الی آخر داشته و قضیه اولیه اثبات می شود. برای اینکار از استقرا استفاده می کنیم:

پایه: P_1 با دو استک و P_2 با سه استک. بالای استک سوم P_2 را با # در استک دوم P_1 نمایش می دهیم. تمام محتویات استک سوم P_2 در ابتدای استک دوم P_1 قرار داده و سپس روی آن ها # را می گذاریم. پس از آن محتویات استک دوم P_2 روی آنها قرار می گیرد. محتویات استک اول هر دو نیز مشابه خواهد بود. واضح است پوش و پاپ در دو استک اول فرقی با یک دیگر ندارند. هنگامی که می خواهیم به استک سوم پوش کنیم، ابتدای تمام محتویات استک دوم P_1 را کرده و در استک اول پوش می کنیم (بالای استک اول را با علامتی خاص مشخص می کنیم). سپس # را برداشته و حرفی مورد نظر را پوش می کنیم. سپس # و تمام محتویاتی که در استک اول قرار داده بودیم را به استک دوم بازمی گردانیم. پاپ نیز به همین صورت انجام می شود.

گام استقرا برای k استک به همان شیوه ای است که در پایه توضیح داده شد. برای استک $k + 1$ ام از استک k استفاده می کنیم و بقیه استک ها تفاوتی با یکدیگر نخواهند داشت (هر استک $(i < k)$ ام در PDA با k استک مشابه $(i < k)$ ام در PDA با $k + 1$ استک خواهد بود.

۸. شبیه سازی ماشین تورینگ استاندارد توسط ماشین مذکور: اشاره گر تمام $k + 1$ نوار را همواره هم زمان جابجا می کنیم. هرگاه ماشین تورینگ استاندارد اشاره گرش رو به راست یا چپ برد و تغییری در خانه فعلی روی نوار نداد، ما نیز نوار همه اشاره گرها را در این ماشین جابجا می کنیم؛ اما اگر خانه ای در ماشین استاندارد تغییر کرد، مثلاً از x به y تغییر کرد، ابتدا از بین k نوار، نواری را که می توان روی آن x نوشت را می یابیم و سپس x را از خانه فعلی آن نوار پاک کرده: پس از آن نواری را که می توان روی آن y را نوشت پیدا می کنیم و در خانه فعلی آن y را می نویسیم؛ پس از اتمام این مراحل نوار را به چپ یا راست، مطابق حرکت اشاره گر ماشین تورینگ استاندارد، حرکت می دهیم. در هر مرحله تناظر نوار ماشین استاندارد و ماشین

مذکور به صورت زیر خواهد بود: (الفبای نوار به صورت $\Gamma = \{a, b, c\}$ می باشد).



شبیه سازی ماشین مذکور توسط ماشین تورینگ استاندارد: $k + 1$ بخش با اندازه یکسان را بر روی نوار ماشین تورینگ استاندارد مشخص می کنیم؛ ابتدا و انتهای هر بخش توسط یک کاراکتر خاص مشخص شده و شماره هر نوار در خانه اول هر بخش نوشته می شود. محتویات هر نوار را در هر بخش متناظر با آن روی نوار ماشین استاندارد قرار می دهیم و با تغییر هر خانه در نوار i ام، محتویات خانه متناظر با آن در بخش i ام روی نوار را تغییر می دهیم. هرگاه اندازه هر بخش کم بود، تمام $k + 1$ بخش را بزرگ تر می کنیم. (این کار به سادگی و با شیفت دادن خانه های نوار امکان پذیر است)