



مقدمه

برای آشنایی با زبان‌های برنامه‌نویسی بهتر است که ابتدا با مفهوم تفکر الگوریتمی آشنا شوید. در فصل اول این کتاب به بررسی چند دستگاه پرداخته‌ایم که طبق دستورالعمل مشخصی دستورات خود را اجرا می‌کنند. در سال‌های اخیر سؤالاتی با این مضامین در مرحله‌ی دوم المپیاد آمده است و منبع منسجمی برای آن‌ها وجود نداشته است. همچنین بیان این موضوعات فضای ذهنی دانش‌پژوهان را به محیط برنامه‌نویسی نزدیک خواهد کرد.

در این فصل چند دستگاه تعریف می‌کنیم و با استفاده از آن‌ها برنامه‌هایی خواهیم نوشت. هر دستگاه از تعدادی دستور تشکیل می‌شود که نحوه‌ی اجرای آن‌ها نیز متفاوت خواهد بود. گام‌به‌گام با مثال‌ها پیش بروید تا با ساختار دستگاه‌ها آشنا شوید.



دستگاه P.A.A

۱-۱

این دستگاه برای اولین بار در یکی از سؤال‌های المپیاد جهانی کامپیوتر آمده بود که در آن سال تنها یک نتیجه‌ی ساده از آن خواسته شده بود. اما به علت قدرت بالای این دستگاه، چندین مثال برای آن ارائه و حل می‌کنیم.

تعریف دستگاه: این دستگاه شامل یک ورودی و چندین دستور است. هر ورودی رشته‌ای از حروف است و دستورات نیز به شکل $s_1 \rightarrow s_2$ می‌باشند. این بدان معنی است اگر در رشته‌ی ورودی، زیررشته‌ی s_1 وجود داشت آن را به s_2 تبدیل می‌کنیم. دستورات به ترتیب از بالا به پایین نوشته می‌شوند و اولویت اجرا

با دستورهای بالاتر است. دستگاه به این ترتیب عمل می‌کند:

از اولین دستور شروع می‌کند هرگاه به دستوری مانند $s_1 \rightarrow s_2$ رسیدیم که رشته‌ی s_1 در ورودی وجود داشته باشد، از سمت چپ اولین زیررشته‌ی s_1 را با زیررشته‌ی s_2 جایگزین می‌کنیم و سپس به دستور اول باز می‌گردیم. اگر هیچ کدام از دستورات قابلیت اجرا نداشته باشند رشته‌ی فعلی را به عنوان خروجی برنامه ارائه می‌دهیم. برای شروع با مثال زیر آغاز می‌کنیم:

مثال ۱-۱

خروجی دستگاه زیر را بیابید.

ورودی:

aabbaba

دستورات:

- aa \rightarrow c
- b \rightarrow aa

راه‌حل: ورودی به صورت زیر تعریف می‌کند:

aa**bbaba \rightarrow cbbaba \rightarrow caababa \rightarrow ccaaaba \rightarrow cccaba \rightarrow cccaaaa \rightarrow ccccaa \rightarrow ccccc**

پس خروجی برنامه رشته‌ی ccccc خواهد بود.

حال با چند مثال دیگر ادامه می‌دهیم:

مثال ۲-۱

اگر ورودی یک دستگاه تعدادی a و دستوراتش به شکل زیر باشد، خروجی آن را بیابید.

- aa \rightarrow a

راه‌حل: دستور این برنامه دو حرف a از اول رشته را برداشته و یک a جای آن‌ها می‌گذارد. پس تا هنگامی که تعداد aها بیش از یک باشد یکی یکی آن‌ها را کم می‌کند تا به یک a برسیم. مثلاً اگر ورودی شامل ۵ تا a باشد خواهیم داشت:

aaaaa \rightarrow aaaa \rightarrow aaa \rightarrow aa \rightarrow a

مثال ۳-۱

ورودی یک دستگاه n عدد a می‌باشد و دستورات ما به شکل زیر است:

- aa \rightarrow b
- a \rightarrow c

خروجی آن را بیابید.

راه‌حل: همان طور که گفته شد تا هنگامی که دستور اول قابلیت اجرا داشته باشد نمی‌توان دستور دوم را



اجرا کرد. ابتدا به ازای هر aa یک b می‌گذاریم. در انتها اگر n زوج باشد $\frac{n}{2}$ تا b خواهیم داشت و اگر n فرد باشد $\lfloor \frac{n}{2} \rfloor$ تا b می‌ماند و آخرین a باقی‌مانده تبدیل به c می‌شود.

مثلاً اگر n برابر ۵ باشد ورودی $aaaaa$ است و به این ترتیب تغییر خواهد کرد:

$$aaaaa \rightarrow \underline{baaa} \rightarrow \underline{bba} \rightarrow \underline{bbc}$$

مثال ۴-۱

فرض کنید ورودی یک دستگاه به شکل زیر باشد:

$$\underbrace{baaa \cdots a}_n$$

برنامه‌ای بنویسید که خروجی آن رشته‌ی زیر شود:

$$\underbrace{aaa \cdots a}_n b$$

راه‌حل: اگر مجموعه دستورات به شکل زیر باشد تا هنگامی که b قبل از a ها باشد یک خانه جلو می‌رود تا به انتهای رشته برسد و در نهایت متوقف خواهد شد.

- $ba \rightarrow ab$

مثال ۵-۱

فرض کنید ورودی یک دستگاه $aaa \cdots a$ باشد. دستگاهی طراحی کنید که در انتها

باقی‌مانده‌ی n بر ۳ را در خروجی نمایش دهد. مثلاً اگر n برابر ۱۰ بود (در ورودی ۱۰ تا a داشتیم)، در خروجی عدد ۱ نشان داده شود.

راه‌حل اول: توجه کنید اگر n بزرگ‌تر از ۳ باشد و ۳ واحد از آن کم کنیم، باقی‌مانده‌اش بر ۳ برابر باقی‌مانده‌ی n خواهد بود. پس به ازای هر aaa یک b می‌گذاریم. در واقع هر b نمایانگر ۳ تا a است. البته تعداد b ها برای ما مهم نیست و آن‌ها را در نهایت یکی‌یکی کم می‌کنیم.

- $aaa \rightarrow b$
- $bb \rightarrow b$
- $baa \rightarrow 2$
- $ba \rightarrow 1$
- $b \rightarrow 0$
- $aa \rightarrow 2$
- $a \rightarrow 1$

توجه کنید دستورهای ششم و هفتم برای حالتی نوشته شده‌اند که در ورودی کم‌تر از ۳ تا a داشته باشیم.

راه حل دوم. از ایده‌ی قسمت قبل استفاده کرده و آن را بهینه می‌کنیم:

- $aaaa \rightarrow a$
- $aaa \rightarrow \circ$
- $aa \rightarrow ۲$
- $a \rightarrow ۱$

در صورتی که طول رشته بیش‌تر از ۳ باشد، ۳ واحد از آن کم می‌شود و در غیر این صورت باقی‌مانده‌ی n بر ۳ جایگزین خواهد شد.

مثال ۶-۱ فرض کنید ورودی دستگاه از تعدادی a و b تشکیل شده باشد و یک c هم در ابتدای آن‌ها باشد.

می‌خواهیم در خروجی تنها دو حرف آخر این رشته باقی بماند. برای مثال اگر ورودی $cabbabba$ باشد، در خروجی باید ba بماند. برنامه‌ای برای این کار بنویسید.

راه حل اول. ابتدا c را به انتها می‌بریم و زمانی که c به انتها رسید از دو حرف آخر می‌پریم و حرف c را به حرف d تغییر می‌دهیم تا دوباره c به انتها نرود و سمت چپ d را حذف می‌کنیم.

- $ca \rightarrow ac$
- $cb \rightarrow bc$

هنگامی که دو دستور اول اجرا نشود یعنی c به انتها رسیده حال دو حرف آخر، چهار حالت روبه‌رو را دارد:

- $aac \rightarrow daa$
- $abc \rightarrow dab$
- $bac \rightarrow dba$
- $bbc \rightarrow dbb$

حرف‌های سمت چپ d را یکی‌یکی حذف می‌کند:

- $ad \rightarrow d$
- $bd \rightarrow d$

در انتها d را حذف می‌کنیم:

- $d \rightarrow \phi$



راه حل دوم. هنگامی که جاوی c حداقل ۳ حرف باشد، حرف اول مسلماً جزو دو حرف آخر رشته نیست. پس آن را حذف می‌کنیم.

- $caaa \rightarrow caa$
 - $caab \rightarrow cab$
 - $caba \rightarrow cba$
 - $cabb \rightarrow cbb$
 - $cbaa \rightarrow caa$
 - $cbab \rightarrow cab$
 - $cbba \rightarrow cba$
 - $cbbb \rightarrow cbb$
 - $c \rightarrow \phi$
-