

Grammar

chapter6-1 , 6-2

- محدودیت زمان: ۲ ثانیه
- محدودیت حافظه: ۲۵۶ مگابایت

برای یک گرامر، یک متود پیاده سازی کنید که به عنوان ورودی، یک شیء گرامر و یک رشته را دریافت کند و در صورتی که رشته توسط گرامر تولید شود، Accepted و در غیر این صورت، Rejected را به عنوان خروجی برگرداند.

در گرامر ورودی، متغیرها داخل $\langle \rangle$ هستند، # نشان دهنده nullable transition است و قواعد مختلف با | از همدیگر جدا می شوند.

ورودی

نحوه ورودی گرفتن برنامه به این صورت است که در ابتدا یک عدد n که بیانگر تعداد متغیرهای گرامر می باشد، در خط اول می آید. سپس، در هریک از n خط بعدی، یکی از قواعد گرامر به عنوان ورودی به شما داده می شود. در انتها به عنوان آخرین خط ورودی، یک رشته داده می شود که باید پذیرفته شدن یا نشدن آن را توسط گرامر بررسی کنید.

توجه: به فاصله ها در ورودی دقت کنید که دقیقاً ورودی را به همین فرمت باید دریافت کنید.

$$1 \leq n \leq 10$$

خروجی

خروجی برنامه ی تنها شامل یک خط است که باید پذیرفته شدن یا نشدن رشته مربوطه را در گرامر به ترتیب با Accepted یا Rejected نمایش دهید.

مثال

ورودی نمونه ۱

3

 $\langle S \rangle \rightarrow a \langle S \rangle b \mid a \langle A \rangle \mid b \langle B \rangle$ $\langle A \rangle \rightarrow a \langle A \rangle \mid \#$ $\langle B \rangle \rightarrow b \langle B \rangle \mid \#$

aaab

$$\begin{aligned}
 S &\rightarrow aSb \mid aA \mid bB \\
 A &\rightarrow aA \mid \lambda \\
 B &\rightarrow bB \mid \lambda
 \end{aligned}$$

خروجی نمونه ۱

 $S \rightarrow aSb \rightarrow aaAb \rightarrow aaaAb \rightarrow aaab$

Accepted

ورودی نمونه ۲

4

 $\langle S \rangle \rightarrow \langle S \rangle a \mid \langle S \rangle b \mid \langle A \rangle a \mid \langle B \rangle b$ $\langle A \rangle \rightarrow ab \langle A \rangle \mid \langle B \rangle ca \mid \#$ $\langle B \rangle \rightarrow b \langle B \rangle \mid \langle C \rangle f$ $\langle C \rangle \rightarrow a \langle C \rangle \mid \#$

abbfcaba

خروجی نمونه ۲

Rejected

PDA Calculator

- محدودیت زمان: ۱ ثانیه
- محدودیت حافظه: ۶۴ مگابایت

با استفاده از **PDA** یک ماشین حساب بسازید که به عنوان ورودی یک عبارت ریاضی را دریافت کند و در صورت معتبر بودن، مقدار محاسبه شده آن را نمایش دهد و در صورتی که ورودی معتبر نباشد، پیام INVALID را چاپ کند.

ماشین حساب شما باید دارای عملگرهای زیر باشد:

۱. پرانتز: (،)
۲. چهار عمل اصلی: + ، - ، * ، /
۳. توان: ^
۴. جذر: $\sqrt{}$
۵. توابع مثلثاتی: \sin, \cos, \tan
۶. تابع قدرمطلق: abs
۷. توابع نمایی: exp, ln

نکات مهم:

- ورودی تنها شامل یک خط است که در آن یک رشته به شما داده می شود.
- ورودی عملگرهای 4 تا 7 داخل پرانتز قرار خواهند گرفت.
- در رشته ورودی، متغیر وجود نخواهد داشت و ورودی تنها شامل اعداد حقیقی خواهد بود.
- هرگونه ورودی خارج از دامنه ی تابع و تقسیم بر 0 نیز باعث INVALID شدن می شود.
- حاصل عبارت خروجی باید به صورت اعشاری و با دو رقم اعشار باشد.

ورودی

ورودی برنامه شامل یک خط است که یک رشته از اعداد، عملگرها و پرانتزها می باشد.

خروجی

اگر ورودی معتبر باشد، در یک خط حاصل آن باید نمایش داده شود. درغیراین صورت، باید عبارت INVALID چاپ شود.

مثال

ورودی نمونه ۱

$10 / 2 + 4 * -3.5$

خروجی نمونه ۱

-9.00

ورودی نمونه ۲

$(10 + 9) * 4$

خروجی نمونه ۲

INVALID

ورودی نمونه ۳

$\sin(\ln(5 ^ 6 * 3 - \cos(\sin(6))))$

خروجی نمونه ۳

-0.97

Turing Machine

- محدودیت زمان: ۱ ثانیه
- محدودیت حافظه: ۱۲۸ مگابایت

برنامه‌ای بنویسید که رشته‌ی کدگذاری شده‌ی یک ماشین تورینگ را دریافت کرده، آن را decode کند و سپس با توجه به آن، ماشین تورینگ مربوطه را پیاده‌سازی کند و هریک از رشته‌های ورودی را بر روی ماشین تست کند. (برای اطلاع از نحوه کدگذاری به اسلایدهای درس مراجعه کنید.)

ورودی

اولین خط از ورودی یک رشته باینری (متشکل از 0 و 1) است که رشته‌ی کدگذاری شده‌ی یک ماشین تورینگ است. در خط دوم ورودی، عدد صحیح n می‌آید که بیانگر تعداد رشته‌هایی است که باید روی ماشین به عنوان ورودی تست شود. سپس در هریک از n خط بعدی، در هر خط یک رشته کدگذاری شده به عنوان ورودی می‌آید که باید روی ماشین پیاده‌سازی شده تست شوند.

$$1 \leq n \leq 10$$

خروجی

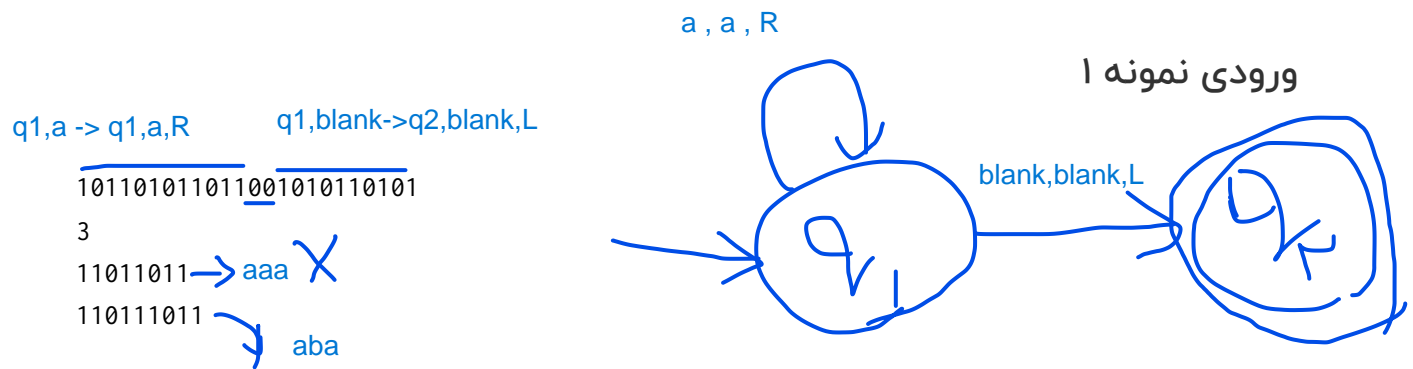
خروجی برنامه دقیقاً شامل n خط است که در هر خط، نتیجه تست هریک از ورودی‌ها روی ماشین تورینگ، به ترتیب چاپ می‌شوند. در صورت قبولی هر ورودی، باید عبارت Accepted و در غیر این صورت، عبارت Rejected چاپ شود.

نکات مهم

- کد حالت آغازین را ۱ در نظر بگیرید.
- کد حالت پایانی را $1^{(number\ of\ states)}$ در نظر بگیرید.

- کد کاراکتر خالی (blank) را در نوار ۱ در نظر بگیرید.
- تضمین می شود در ماشین تورینگ داده شده، حلقه بی نهایت رخ نمی دهد.

مثال



Accepted

Rejected

blank = 1

a = 11 b = 111

ورودی نمونه ۲

101110110111101100101101111011011001101101101100110111011011011001101111101101101

5

111011111011111111

1111101110111

1110111011111011111

111011111

خروجی نمونه ۲

Rejected

Rejected

Accepted

Accepted

