



دانشکده مهندسی کامپیوتر

دکتر رضا انتظاری ملکی

بهار ۱۴۰۰

پروژه دوم

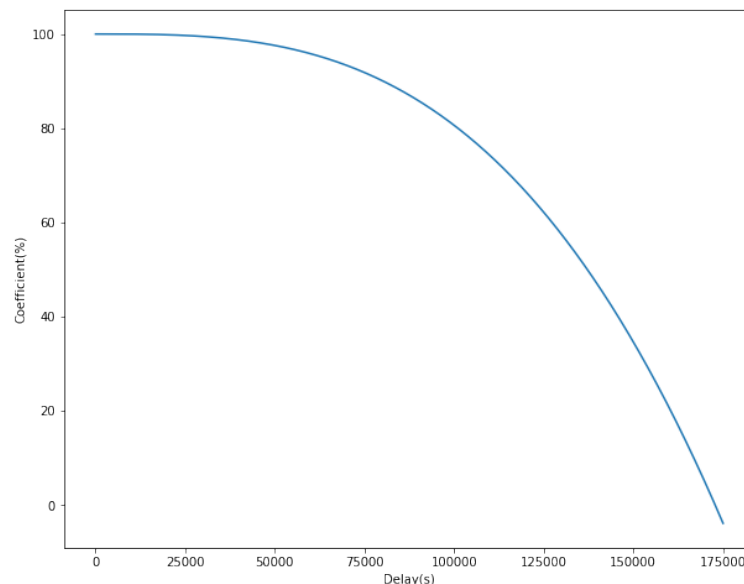
نظریه زبانها و ماشینها

طراح: گروه دستیاران آموزشی درس

تاریخ تحویل: ۲۰ تیر ۱۳۹۹:۵۹

قوانین

- در صورت مشاهدهی هرگونه تقلب (از گروه‌های دیگر یا از کدهای موجود در اینترنت)، نمره‌ی کل پروژه را از دست خواهید داد.
- پروژه باید به صورت انفرادی انجام گیرد.
- به دلیل فشرده بودن برنامه‌ی ترم، امکان تمدید مهلت پروژه وجود ندارد.
- در صورت وجود هرگونه سوال از طریق گروه تلگرام یا تیمز مطرح کنید. (لطفا پی‌وی پیام ندهید.)
- دقت کنید که پروژه دوم به صورت سوال به سوال خواهد بود و هر سوال را باید به طور مجزا در کوئرا آپلود کنید.
- توجه کنید علاوه بر داوری خودکار کوئرا، کدهای شما به صورت دستی نیز بررسی خواهند شد.
- برای پیاده‌سازی پروژه می‌توانید از هر زبان برنامه‌نویسی دلخواه خود استفاده کنید.
- محور افقی این نمودار مقدار تاخیر به ثانیه و محور عمودی ضریب اعمالی در نمره پروژه است.



شکل ۱: نمودار تاخیر



فهرست مطالب

۳	۱ Grammar
۴	۲ Turing Machine
۵	۳ PDA Calculator
۷	۴ نحوه ارسال

شرح پروژه

۱ Grammar

برای یک گرامر، یک متود پیاده‌سازی 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.

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

به عنوان نمونه:

ورودی:

```
3
< S > → a < S > b | a < A > | b < B >
< A > → a < A > | #
< B > → b < B > | #
aaab
```

خروجی:

Accepted

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

راهنمایی: الگوریتم شما باید ابتدا گرامر را به فرم نرمال چامسکی تبدیل نماید (در صورتی که گرامر در فرم نرمال CNF نباشد). و سپس قبول شدن یا نشدن رشته داده‌شده را با استفاده از الگوریتم CYK بررسی کند.

Turing Machine ۲

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.)

اولین خط از ورودی یک رشته باینری (متشکل از ۰ و ۱) 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.

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.

به عنوان نمونه:

ورودی:

101101011011001010110101

3

11011011

110111011

خروجی:

Accepted

Accepted

Rejected

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 می‌شود در ماشین تورینگ داده‌شده، حلقه بی‌نهایت رخ نمی‌دهد.

۳ PDA Calculator

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.

پیاده سازی شما حتما باید دارای موارد زیر باشد:

۱. طراحی گرامر جهت تشخیص معتبر بودن رشته ورودی
۲. استفاده از PDA و پشته برای محاسبه حاصل عملیات

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. توابع معکوس مثلثاتی: asin, acos, atan
۷. تابع علامت: sgn
۸. تابع قدرمطلق: abs
۹. توابع نمایی: exp, ln
۱۰. توابع هیپربولیک: sinh, cosh, tanh

نکات مهم:

- ورودی تنها شامل یک خط است که در آن یک رشته به شما داده می شود.
- ورودی عملگرهای ۴ تا ۱۰ 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 ۰ نیز باعث INVALID شدن می شود.
- حاصل عبارت خروجی باید به صورت اعشاری و با دو رقم اعشار باشد.



به عنوان نمونه:

ورودی:

$10 / 2 + 4 * -3.5$

خروجی:

-9.00

ورودی:

$(10 + 9) * 4)$

خروجی:

INVALID

ورودی:

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

خروجی:

-0.97

۴ نحوه ارسال

شما برای هر سوال تنها باید یک فایل را آپلود کنید. در صورتی که نیازمند تعریف کلاس‌های متعدد بودید، تمام کلاس‌ها باید در یک فایل نوشته شوند و همان یک فایل را به عنوان پاسخ بارگزاری کنید.

علاوه بر ارسال پاسخ سوالات، شما باید گزارشی مختصر از کدهای خود و روند اجرای برنامه، در قالب یک فایل PDF، در محل مشخص شده در کوئرا بارگزاری کنید.

توجه کنید که کیفیت گزارش شما اهمیت دارد و بر نمره پروژه تاثیر خواهد گذاشت. پس برای گزارش خود زمان کافی بگذارید.

موفق باشید.