

بسمه تعالی



دانشگاه صنعتی امیرکبیر
(پلی تکنیک تهران)

نظریه زبان ها و ماشین ها

پروژه میانترم

مهلت ارسال پاسخ:

25 فروردین 1404 ساعت 23:59

مقدمه:

در این پروژه، هدف پردازش مجموعه‌ای از گرامرهای منظم، تبدیل آن‌ها به ماشین‌های متناهی، و انجام عملیات مجموعه‌ای روی این ماشین‌ها است. گرامرهای منظم یکی از روش‌های توصیف زبان‌های منظم هستند و به راحتی می‌توان آن‌ها را به ماشین‌های متناهی غیرقطعی (NFA) تبدیل کرد. سپس با استفاده از الگوریتم‌های استاندارد، NFAها به ماشین‌های متناهی قطعی (DFA) معادل تبدیل می‌شوند. در نهایت، با توجه به کاربردهای زبان‌های منظم، عملیات مجموعه‌ای روی DFAها مانند اجتماع، اشتراک و متمم انجام خواهد شد.

این مستند شامل تعریف کلی پروژه، جزئیات فنی، مراحل پیاده‌سازی و نحوه نمایش خروجی‌ها خواهد بود.

تعریف کلی از پروژه:

این پروژه یک برنامه‌ی پردازش زبان‌های منظم است که ورودی آن مجموعه‌ای از گرامرهای منظم و خروجی آن نمایش DFAهای معادل و عملیات مجموعه‌ای روی آن‌ها خواهد بود. مراحل پردازش شامل:

- خواندن گرامرهای منظم از فایل متنی
- تبدیل گرامرهای منظم به NFA معادل
- تبدیل NFA به DFA معادل
- انجام عملیات مجموعه‌ای روی DFAها (اجتماع، اشتراک و متمم)
- نمایش DFAهای حاصل و چاپ جداول انتقال آن‌ها

این برنامه کمک می‌کند تا درک بهتری از مفاهیم گرامرهای منظم، ماشین‌های متناهی و عملیات روی زبان‌های منظم داشته باشیم.

شرح پروژه:

این پروژه شامل چندین مرحله برای پردازش و انجام عملیات روی زبان‌های منظم است. پس از دریافت یک فایل متنی شامل چندین گرامر منظم، پردازش آن‌ها به ترتیب انجام می‌شود. مراحل کلی پروژه به شرح زیر است:

قسمت اول: خواندن گرامرهای منظم از فایل متنی

ورودی این برنامه یک فایل متنی است که شامل چندین گرامر منظم است. هر گرامر در فرمت مشخصی نوشته شده و شامل موارد زیر است:

1. نماد شروع
2. حروف الفبای زبان (ترمینالها)
3. مجموعه‌ای از متغیرها (نان ترمینالها)
4. مجموعه‌ای از قواعد تولید به فرم $A \rightarrow a$ یا $A \rightarrow aB$
5. عملیات مورد انتظار برای اعمال (اجتماع، اشتراک و متمم)

مثال قالب ورودی:

$S \rightarrow aA$

$A \rightarrow bS$

$A \rightarrow \epsilon$

Complement

قالب ورودی و چند تست کیس در فایل input.txt موجود است.

قسمت دوم: تبدیل گرامرهای منظم به NFA

پس از خواندن گرامرهای منظم، هر گرامر به یک ماشین متناهی غیرقطعی (NFA) معادل تبدیل می‌شود. این تبدیل به گونه‌ای انجام می‌شود که زبان تولیدشده توسط NFA دقیقاً همان زبان تولیدشده توسط گرامر اولیه باشد.

قسمت سوم: تبدیل NFA به DFA

NFAهای ساخته‌شده در مرحله قبل، برای ساده‌سازی و استفاده در عملیات مجموعه‌ای، به ماشین‌های متناهی قطعی (DFA) معادل تبدیل می‌شوند.

قسمت چهارم: اجرای عملیات مجموعه‌ای روی DFAها

پس از ساخت DFAها، عملیات مجموعه‌ای روی آنها انجام می‌شود. این عملیات شامل موارد زیر است:

- اجتماع دو DFA: زبانی را می‌سازد که شامل تمام کلماتی است که حداقل در یکی از دو DFA اولیه پذیرفته می‌شوند.

- اشتراک دو DFA: زبانی را می‌سازد که تنها شامل کلماتی است که در هر دو DFA اولیه پذیرفته می‌شوند.
- متمم یک DFA: مجموعه‌ی حالات پذیرش و غیرپذیرش را جابه‌جا می‌کند تا زبان مکمل آن ساخته شود.

توجه داشته باشید که پیاده‌سازی شما باید قابلیت اجتماع و اشتراک بیش از دو DFA را هم داشته باشد. (مثلا یک راه این است که تابعی که اجتماع دو DFA را می‌گیرد را دوبار صدا بزنید. به‌طوری که ابتدا برای DFA اول و DFA دوم اجتماع را بدست آورید و در نهایت حاصل را با DFA سوم اجتماع بگیرید.)

مثال:

در اینجا یک مثال را باهم بررسی خواهیم کرد.

G1:

$S \rightarrow aA \mid bC$

$A \rightarrow aA \mid bB$

$B \rightarrow aA \mid bB \mid \epsilon$

$C \rightarrow aC \mid bC$

=====

G2:

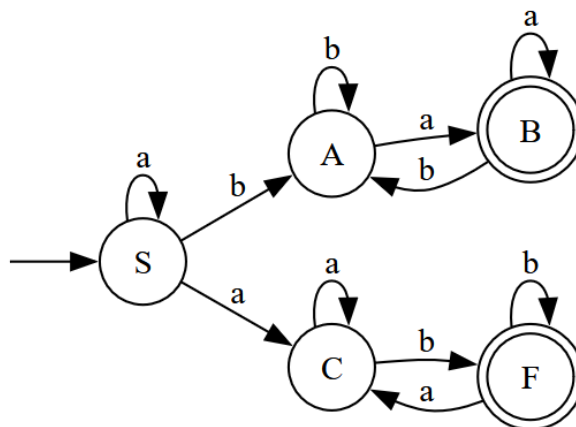
$S \rightarrow aC \mid bA$

$A \rightarrow aB \mid bA$

$B \rightarrow aB \mid bA \mid \epsilon$

$C \rightarrow aC \mid bC$

حاصل اشتراک DFA این دو گرامر منظم:



فرمت خروجی مورد انتظار از شما در فایل output.txt موجود است.

نکات مربوط به تحویل:

فرمت گزارش و فایل‌ها

- گزارش پروژه باید به زبان فارسی و در قالب فایل PDF تهیه شود.
- تمام فایل‌های مربوط به پروژه (گزارش و کدها) باید در یک فایل فشرده قرار داده شوند.
- نام فایل گزارش باید به فرمت P_report_StdNumber.pdf باشد.
- ارسال پروژه بدون گزارش فاقد ارزش است.

کدنویسی و مستندسازی

- پیاده‌سازی پروژه باید با یکی از زبان‌های Python، Java یا C++ انجام شود.
- کامنت‌گذاری (حداقل در حد لازم) ضروری است.
- تمام فایل‌های مربوط به کدها را در یک پوشه با نام codes ذخیره کنید.
- فایل‌های کد و گزارش را مطابق فرمت‌های ذکر شده در یک فایل فشرده (zip) با فرمت P1_StdNumber.zip ارسال کنید.

مهلت تحویل

- مهلت ارسال پروژه تا تاریخ 25 فروردین می‌باشد. لطفاً برنامه‌ریزی مناسبی برای ارسال به‌موقع داشته باشید تا مشمول کسر نمره نشوید.

- تأخیر در ارسال پروژه، مشمول کسر نمره خواهد شد.

رعایت قوانین و اصول امانتداری

- مطابق قوانین دانشگاه، هرگونه کپی برداری، اشتراک گذاری کد با دیگران و استفاده مستقیم از کدهای اینترنتی ممنوع است.

- تولید کد با ابزارهای هوش مصنوعی (مانند ChatGPT) نیز اکیداً ممنوع می باشد.

- تحویل پروژه به صورت آنلاین انجام می شود و انتظار می رود که تسلط کامل روی کدنویسی داشته باشید.

در صورت داشتن سوال یا مشکل در خصوص تحویل پروژه، می توانید از طریق تلگرام با آیدی های زیر در ارتباط باشید:

@HolyBardia

@roza_gp