



University of Isfahan

Compiler Phase 2 Project Syntax Analyser

Professor:

Arash Shafiei

Presented By:

Arshia Shafiei Alavijeh - 4003623019

Kimya Mirmoghtadaei – 4012363091

پروژه‌ی ما از 9 قسمت اصلی تشکیل شده است.

در فایل Grammar گرامر نوشته شده است. برای نوشتن گرامر از [این لینک](#) کمک گرفته‌ایم. هرچند که در مسیر نوشتن هرکجا که احساس کردیم قانون بیشتری نیاز است یا قانونی اضافه است آن را اضافه یا حذف کرده‌ایم و قوانین با لینک پیوست شده بسیار متفاوت هستند.

در فایل‌های LeftFactoring و LeftRecursion الگوریتم‌های فاکتورگیری چپ و حذف بازگشت چپ پیاده‌سازی شده‌اند. هردوی این الگوریتم‌ها براساس شبه‌کدهای داخل اسلایدها و کاملاً از صفر پیاده‌سازی شده‌اند. الگوریتم فاکتورگیری چپ برای سادگی بیشتر، به جای پیدا کردن بلندترین زیرقانون مشترک، آن‌ها را یکی یکی فاکتور می‌گیرد. این مسئله ظاهر خروجی را کمی بهم می‌ریزد اما خروجی کاملاً درست است.

در ادامه فایل Fcal قرار دارد که مجدداً طبق شبه‌کدهای داخل اسلایدهای درست به محاسبه‌ی First و Follow ها می‌پردازد. هیچ کدام از این مراحل برای تهیه‌ی جدول تجزیه به صورت دستی محاسبه‌نشده‌اند و همگی مستقیماً از طریق کد به دست آمده‌اند.

با استفاده از گرامر خروجی پس از حذف بازگشت از چپ‌ها و گرفتن فاکتورها و همچنین نتایج حاصل از محاسبات First و Follow اطلاعات به دست آمده را به فایل ChackLL1 می‌دهیم تا سه شرط مربوط به LL1 بودن بررسی شود.

نتیجه نشان می‌دهد که گرامر تهیه شده توسط تیم ما LL1 است و بنابراین می‌توانیم First و Follow ها را به فایل ParsingTable بدهیم تا جدول تجزیه‌ی مربوط به آن ساخته شود. در این قسمت بازبازی خطا توسط توکن همگام‌کننده نیز پیاده‌سازی شده است.

در نهایت جدول تجزیه‌ی ثابت ما با توجه به گرامرمان، به شکل زیر خواهد بود:

Parsing Table:

M[Program, t_int]: Program -> ['Declist']

M[Program, t_char]: Program -> ['Declist']

M[Program, t_bool]: Program -> ['Declist']

M[Program, \$]: synch

=====

M[Declist, t_int]: Declist -> ['Dec', "Declist"]

M[Declist, t_char]: Declist -> ['Dec', "Declist"]

M[Declist, t_bool]: Declist -> ['Dec', "Declist"]

M[Declist, \$]: synch

=====

M[Declist', t_int]: Declist' -> ['Dec', "Declist'"]
M[Declist', t_char]: Declist' -> ['Dec', "Declist'"]
M[Declist', t_bool]: Declist' -> ['Dec', "Declist'"]
M[Declist', \$]: Declist' -> ['ε']

=====

M[Dec, t_int]: Dec -> ['Type', 't_id', 'Declaration']
M[Dec, t_char]: Dec -> ['Type', 't_id', 'Declaration']
M[Dec, t_bool]: Dec -> ['Type', 't_id', 'Declaration']
M[Dec, \$]: synch

=====

M[Declaration, t_assign]: Declaration -> ['Vardec']
M[Declaration, t_lb]: Declaration -> ['Vardec']
M[Declaration, t_comma]: Declaration -> ['Vardec']
M[Declaration, t_semicolon]: Declaration -> ['Vardec']
M[Declaration, t_lp]: Declaration -> ['Funcdec']
M[Declaration, t_char]: synch
M[Declaration, t_bool]: synch
M[Declaration, \$]: synch
M[Declaration, t_int]: synch

=====

M[Type, t_int]: Type -> ['t_int']
M[Type, t_bool]: Type -> ['t_bool']
M[Type, t_char]: Type -> ['t_char']
M[Type, t_id]: synch

=====

M[Vardec, t_assign]: Vardec -> ['Vardeclist', 't_semicolon']
M[Vardec, t_lb]: Vardec -> ['Vardeclist', 't_semicolon']
M[Vardec, t_char]: Vardec -> ['Vardeclist', 't_semicolon']

```
M[Vardec, $]: Vardec -> ['Vardeclist', 't_semicolon']  
M[Vardec, t_bool]: Vardec -> ['Vardeclist', 't_semicolon']  
M[Vardec, t_int]: Vardec -> ['Vardeclist', 't_semicolon']  
M[Vardec, t_semicolon]: Vardec -> ['Vardeclist', 't_semicolon']  
M[Vardec, t_comma]: Vardec -> ['Vardeclist', 't_semicolon']
```

این قسمتی از جدول تجزیه‌ی تولید شده توسط فایل مضمون می‌باشد. مابقی جدول نیز داخل ترمینال قابل بررسی است.

کد مربوط به تجزیه‌ی پیش‌بینی کننده بالا به پایین، در فایل PredictiveParser پیاده‌سازی شده است. درخت تجزیه با استفاده از anytree نیز در همین فایل ساخته و چاپ می‌شود.

خروجی Tokenizer به علاوه‌ی جدول تجزیه را به این فایل می‌دهیم تا طبق الگوریتم تجزیه شود و درخت تجزیه تولید شود.

در صورتی که با خطای نحوی مواجه شدیم، عبارت syntax error با رنگ قرمز داخل ترمینال پرینت می‌شود و تجزیه ادامه پیدا می‌کند.

با توجه به این که گفته شد برای این قسمت به داک جامعی احتیاج نداریم، این داک صرفاً جهت توضیح عملکرد قسمت‌های مختلف جهت شفاف‌سازی تهیه شده‌است. در صورتی که سوالات بیشتری در زمینه‌ی کد مطرح است داک کامل‌تری قابل ارائه خواهد بود.

با تشکر.