

پروژه نهایی کامپایلر

فاز دوم – تحلیلگر لغوی

دکتر فرید فیضی – ترم ۱۴۰۱۰۲

در فاز اول پروژه‌ی اصول طراحی کامپایلر، یک تحلیلگر لغوی (Analyzer Lexical) بر اساس قوانین مطرح شده طراحی گردید، حال نوبت آن است که یک تحلیلگر نحوی (Parser) مناسب برای آن طراحی کنیم که ویژگی‌های مورد نظر برای این تحلیلگر نحوی در ادامه بررسی خواهد شد. در این مرحله می‌توانید از تحلیلگر لغوی‌ای که در فاز قبلی طراحی کردید استفاده کنید و در صورت نیاز در آن تغییرات لازم را برای استفاده در تحلیلگر نحوی ایجاد کنید. ابزار مورد استفاده در این فاز جاوا کاپ (cup) خواهد بود که می‌توانید برای اطلاعات بیشتر به مثال‌های این ابزار مراجعه کنید.

• نحوه‌هایی که تحلیلگر باید حتماً آن‌ها را پشتیبانی کند

- تعریف متغیرهای عدد صحیح، اعشاری و رشته‌ی متنی همانند زبان جاوا (int, double, float, string)
 - ساختارهای حلقه (for, do...while, while)
 - ساختارهای شرطی (if, else if, else, switch)
 - ساختار توابع
 - تابع از پیش تعریف شده‌ی println():
- هر زمان در کد ورودی از این تابع استفاده شود مقدار متنی داخل یا عبارت ریاضی داخل پرانتز، در خروجی نمایش داده شود. مثال:

```
//input:
println("Hello")
//output:
Hello

//-----
//input:
println(2*3+1)
//output:
7
```

- **عملیات‌های ریاضی:**

تحلیل‌گر نحوی باید قادر به انجام محاسبات ریاضی با رعایت اولویت محاسباتی (برای مثال اولویت دادن ضرب به جمع) باشد که برای نمایش آن می‌توان این عبارت را در یک تابع `println` قرار داد یا آن را درون متغییر دیگری ذخیره کرد. (شامل تمامی اعمال ریاضی که در فاز اول تشخیص داده شد مانند: `+`, `-`, `*`, `/`, `**`)

- **پاک کردن صفحه پس از یک زمان مشخص:**

در این نحو، پس از آمدن `CLEAR(t)`، پس از گذشت `t` ثانیه، صفحه پاک شود (در صورت عدم پشتیبانی کنسول از دستور پاک کردن می‌توانید از تعدادی `Enter` به جای آن استفاده کنید).

- **نمره‌ی اضافی**

- **پشتیبانی از مقداردهی متغیرها:**

تحلیل‌گر باید قادر به نگهداری مقادیر متغیرها جهت نمایش آن‌ها در آینده یا استفاده آن‌ها در اعمال ریاضی باشد. (تعداد و نام این متغیرها ثابت نیست) مثال:

```
//input:
int x = 1;
int y = 2;
println(2*x+y);
//output:
4
```

نکات مهم:

پروژه می‌تواند به شکل انفرادی و یا در قالب گروه های حداکثر دو نفره تحویل داده شود. تمامی اعضای گروه باید در ارائه‌ی پروژه حضور فعال داشته باشند.

- پروژه را در قالب یک فایل فشرده (zip) ، شامل فایل (های) مربوط به **jflex** ، **cup** و گزارش را در کوثر ارسال کنید.
- یک فایل **pdf** به نام **report** به عنوان گزارشکار از عملکرد پروژه شامل توضیحات مورد نیاز درباره کد را به همراه تصاویری از مراحل اجرا در فایل فشرده پروژه قرار دهید.
- مسئولیت اشکال در فایل ارسالی و یا عدم اجرا صحیح پروژه به هر شکلی، بر عهده دانشجو خواهد بود.
- پروژه‌ی ارسالی با استفاده از الگوریتم شباهت‌یاب **Stanford** با یکدیگر و با پروژه‌های موجود مقایسه خواهند شد. مسئولیت شباهت مشکوک/غیرقابل توجیه بر عهده‌ی دانشجو خواهد بود.
- تاریخ و زمان ارائه‌ی پروژه، متعاقباً اعلام خواهد شد.