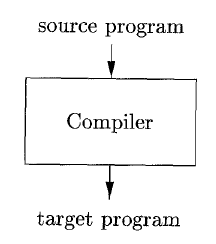
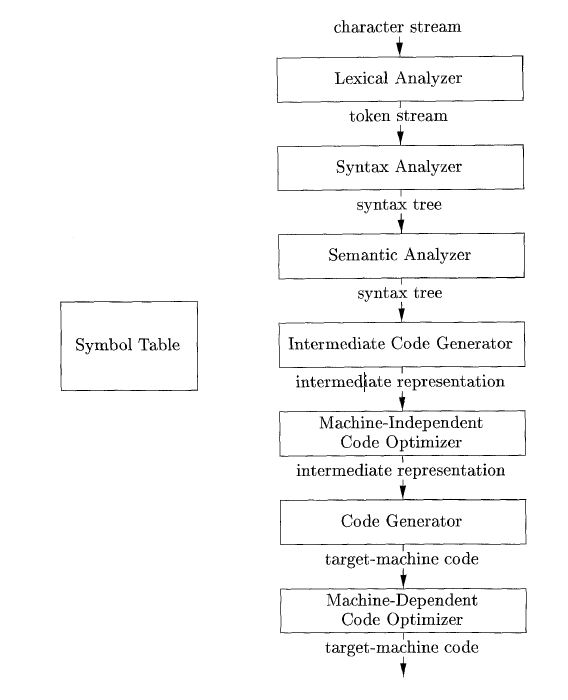
**گزارش فاز اول پروژه درس طراحی کامپایلر:**



یک کامپایلر برنامه ای است که می تواند دستورات برنامه ای به یک زبان (the source language) را خوانده و آن را به برنامه ی معادلش در زبانی دیگر(the target language) تبدیل کند. یکی از وظایف مهم کامپایلر گزارش خطاهایی است که در برنامه ورودی هنگام ترجمه آن رخ می دهد.

در صورتی که نگاهی دقیق تر به عملکرد کامپایلر بیاندازیم متوجه می شویم که کامپایلر از دو بخشanalysis وsynthesis تشکیل شده است.

بخش analysis برنامه ورودی را به چند قسمت تقسیم کرده و ساختار های گرامری را بر آن ها اعمال و از این ساختار ها برای ساختن یک نمود کلی از برنامه استفاده می کند. اگر بخش آنالیز متوجه وجود خطاهای syntactically و semantically در برنامه بشود باید به کاربر پیام آگاهی دهنده ای ارسال کند. این بخش همچنین موظف است اطلاعاتی راجع به برنامه ورودی جمع آوری کرده و آن ها را در ساختمان داده ای به نام symbol table ذخیره کند. بخش synthesis هدف مطلوب برنامه ورودی را با استفاده از نمود کلی و اطلاعاتی که در symbol table موجود می باشد تعیین می کند.



**Lexical Analysis:**

مرحله اول یک کامپایلر Lexical Analysis نامیده می شود.

Lexical Analyzer رشته ای از کاراکتر ها که برنامه ورودی را

تشکیل می دهند را خوانده و آن ها را به دسته های معنی دار که

lexeme گروه بندی و تقسیم می کند. برای هر یک از lexeme ها

Lexical Analyzer یک توکن به عنوان خروجی به فرم زیر تولید

می کند. <token-name , attribute-value>

این خروجی را به مرحله بعدی از کامپایلر یعنی syntax analysis

ارسال می کند.

**Phases of a compiler**

