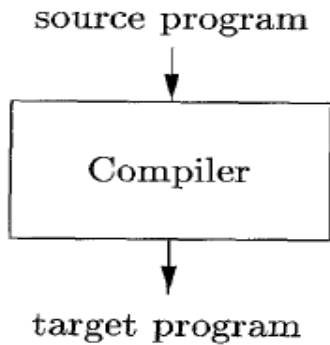


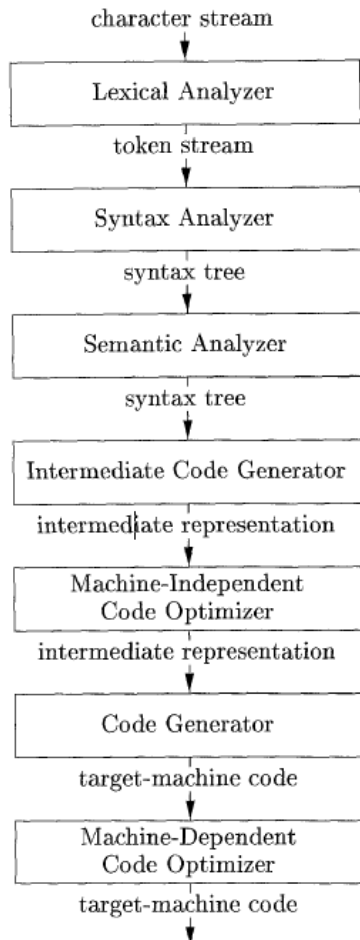
گزارش فاز اول پروژه درس طراحی کامپایلر:



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 ببینیم متوجه می شویم که کامپایلر از دو بخش analysis و synthesis تشکیل شده است.

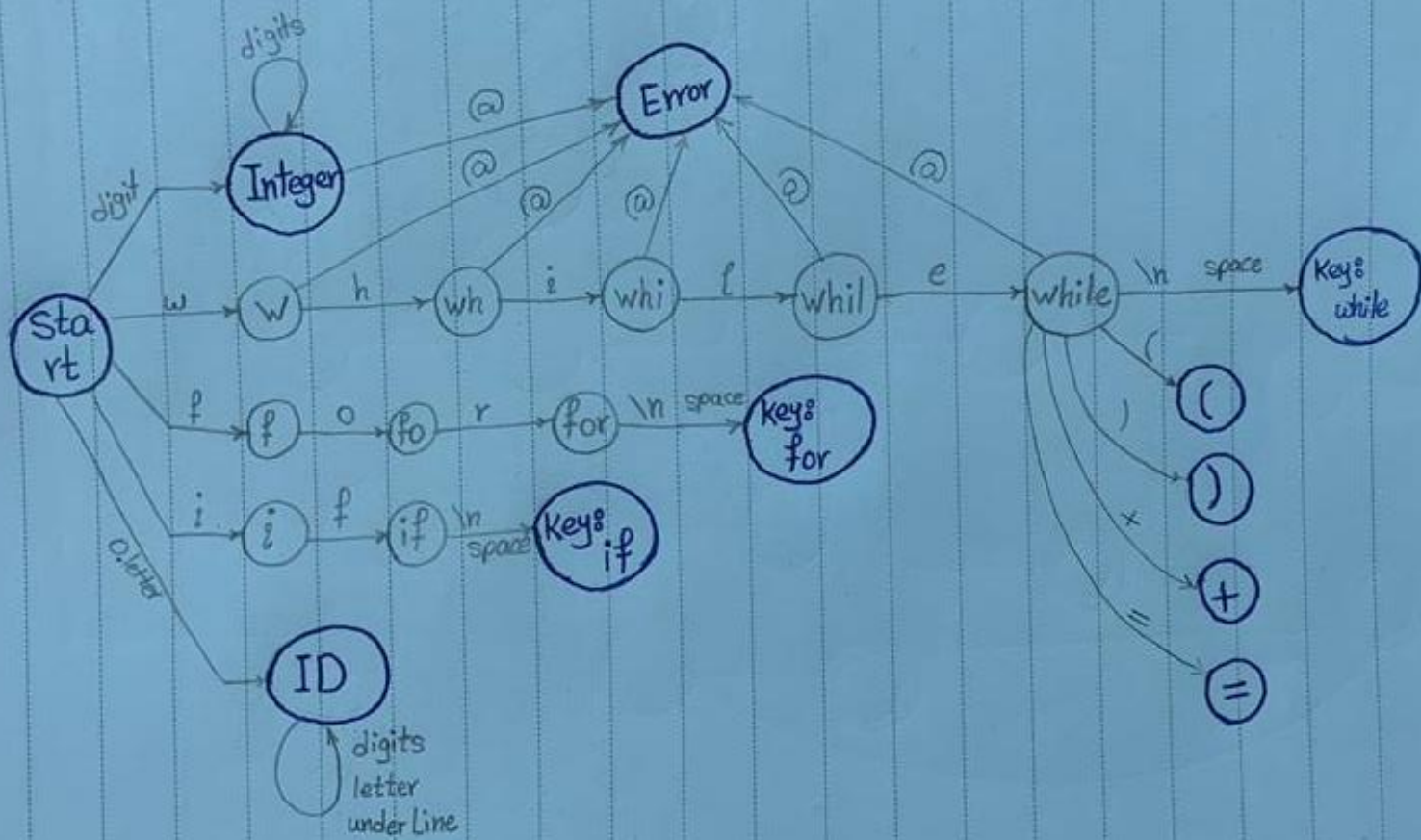
بخش analysis برنامه ورودی را به چند قسمت تقسیم کرده و ساختارهای گرامری را بر آن ها اعمال و از این ساختارها برای ساختن یک نمود کلی از برنامه استفاده می کند. اگر بخش آنالیز متوجه وجود خطاهای syntactically و semantically در برنامه بشود باید به کاربر پیام آگاهی دهنده ای ارسال کند. این بخش همچنین موظف است 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 کاراکترها 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. $\langle \text{token-name}, \text{attribute-value} \rangle$ این خروجی را به مرحله بعدی از کامپایلر یعنی syntax analysis ارسال می کند.

**Phases of a
compiler**



	w	f	i	h	l	e	o	r	letter	digit	\n	"	"	()	+	=	@	
start	w	f	i	h	l	e	o	r	letter	digit	\n	"	"	()	+	=	Error	
w	identifier	identifier	identifier	wh	identifier	identifier	identifier	identifier	integer	start	start	start	start	()	+	=	Error	
wh	identifier	identifier	whi	identifier	identifier	identifier	identifier	identifier	integer	start	start	start	start	()	+	=	Error	
whi	identifier	identifier	identifier	whil	identifier	identifier	identifier	identifier	integer	start	start	start	start	()	+	=	Error	
whil	identifier	identifier	identifier	identifier	while	identifier	identifier	identifier	integer	start	start	start	start	()	+	=	Error	
while	identifier	identifier	identifier	identifier	identifier	identifier	identifier	identifier	KEY: whileKEY: while	()	()	()	+	=	Error	
KEY: while	w	f	i	identifier	identifier	identifier	identifier	for	integer	start	start	start	start	()	+	=	Error	
f	identifier	identifier	identifier	identifier	identifier	identifier	for	integer	integer	start	start	start	start	()	+	=	Error	
fo	identifier	identifier	identifier	identifier	identifier	identifier	for	integer	integer	start	start	start	start	()	+	=	Error	
for	identifier	identifier	identifier	identifier	identifier	identifier	integer	KEY: for	KEY: for	()	()	()	+	=	Error	
KEY: for	w	f	i	identifier	identifier	identifier	integer	integer	integer	start	start	start	start	()	+	=	Error	
i	identifier	if	identifier	identifier	identifier	identifier	integer	integer	integer	start	start	start	start	()	+	=	Error	
if	identifier	identifier	identifier	identifier	identifier	integer	integer	integer	KEY: if	KEY: if	()	()	()	+	=	Error
KEY: if	w	f	i	identifier	identifier	integer	integer	integer	integer	start	start	start	start	()	+	=	Error	
identifier	identifier	identifier	identifier	identifier	integer	integer	integer	integer	integer	start	start	start	start	()	+	=	Error	
Integer	Error	Error	Error	Error	Error	Error	Error	Error	Integer	start	start	start	start	()	+	=	Error	
Error	Error	Error	Error	Error	Error	Error	Error	Error	Error	start	start	start	start	()	Error	start	Error	
+	w	f	i	identifier	identifier	integer	integer	integer	integer	start	start	start	start	()	+	=	Error	
=	w	f	i	identifier	integer	integer	integer	integer	integer	start	start	start	start	()	+	=	Error	
(w	f	i	integer	integer	integer	integer	integer	integer	start	start	start	start	()	+	=	Error	
)	w	f	i	integer	integer	integer	integer	integer	integer	start	start	start	start	()	+	=	Error	