

به نام خدا

طراحی و پیاده سازی Lexical Analyzer و Syntax Analyzer برای

زبان برنامه نویسی مایک

مستند فاز اول پروژه کامپایلر

استاد مربوطه:

دکتر مریم اسدی

تهیه شده توسط:

سارا کهتری ۹۸۳۶۱۳۰۴۷

پریسا عسگرزاده ۹۸۳۶۱۳۰۳۹

گروه ۹

پاییز و زمستان ۱۴۰۰

فهرست

- ۱ - مقدمه-----۳
- ۲ - فرآیند انجام پروژه-----۳
- ۳ - کلمات کلیدی زبان-----۳
- ۴ - نام متغیرها و توابع-----۴
- ۵ - مقادیر ثابت-----۴
- ۶ - کامنت های تک خطی و چند خطی-----۴
- ۷ - عملگرها و سایر توکن ها-----۵
- ۸ - جدول-----۵

۱ - مقدمه

در این فاز از پروژه درس کامپایلر، به طراحی یک تحلیل‌گر لغوی پرداخته می‌شود. تحلیل‌گر لغوی بخش اول کامپایلر است و توسط آن، از رشته ورودی، توکن‌ها استخراج می‌شوند.

لکسیم‌هایی که در این برنامه تشخیص داده می‌شوند عبارتند از:

۱ کلمات کلیدی زبان

۲ نام متغیرها و توابع

۳ مقادیر ثابت

۴ کامنت‌های تک خطی و چند خطی

۵ عملگرها و سایر توکن‌ها

در انتهای سند، جدولی شامل تمامی توکن‌هایی که تشخیص داده می‌شوند، ارائه می‌شود.

۲ - فرآیند انجام پروژه

در این فاز از پروژه، سه جلسه بین اعضای تیم تشکیل شد.

در جلسه اول، دو سند مربوط به پروژه توسط اعضای تیم، مطالعه شده و فیلم مربوط به آموزش نصب `vscode` و `flex` دیده شد. پس از نصب برنامه `vscode` و افزونه `flex`، لینک‌های آموزشی موجود در سند پروژه، خوانده شد. سپس با استفاده از اطلاعات اسناد داده شده، توکن‌های مربوطه شناخته و استخراج شده و در کد پروژه وارد شدند. ابتدا توکن‌های آسان‌تر مثل کلمات کلیدی و عملگرها نوشته شدند و در نهایت سعی شد که برای توکن‌های متغیرها و توابع و مقادیر ثابت نامگذاری مناسب پیدا شود.

در جلسه دوم مطالب پیچیده‌تر پروژه، مانند کامنت‌ها و نام متغیرها نوشته شدند. همچنین برخی ایرادات از جلسه قبل رفع شد.

در جلسه سوم و آخر، سعی شد که رنج عدد ثابت `int` تصحیح شود. و در ادامه به نوشتن مستند گزارش فاز اول، پرداخته شد.

۳ - کلمات کلیدی زبان

کلمات کلیدی یک زبان کلماتی هستند که در یک زبان برنامه نویسی شناخته شده هستند و نمی‌توان از آن‌ها در مفهوم دیگری استفاده کرد. این کلمات نسبت به کلمات دیگر در اولویت هستند بنابراین در این پروژه نیز اولویت داده شده و قبل از باقی کلمات (نام متغیرها و توابع و یا کلمات دیگر) تشخیص داده می‌شوند.

۴ - نام متغیرها و توابع

تنها شرطی که برای نام متغیرها وجود دارد این است که با عدد شروع نشوند. بنابراین اولین حرف باید با "-" و یا با حروف بزرگ و کوچک انگلیسی باشد. در ادامه می‌توانیم مجموعه‌ای از ارقام ۰ تا ۹، "-" و حروف بزرگ و کوچک انگلیسی را داشته باشیم. همچنین لازم به ذکر است که به جز اعداد و حروف انگلیسی، تنها استفاده از "-" مجاز است. در صورتی که نام متغیری با عدد شروع شود، آن توکن ناشناخته محسوب می‌شود.

۵ - مقادیر ثابت

مقادیر ثابت، شامل اعداد و رشته‌ها است که در این برنامه اعداد در بازه ۳۲۷۶۸- تا ۳۲۷۶۷ یعنی یک سیستم ۱۶ بیتی، شناخته می‌شوند.

برای شناخت اعداد، از دو قانون استفاده شده است. در قانون اول، فقط اعداد پنج رقمی تشخیص داده می‌شوند. در صورتی که در بازه مذکور قرار داشته باشند، به عنوان `TOKEN_INT_CONST` شناخته می‌شوند و در غیر این صورت، `TOKEN_INT_CONST_OUT_OF_RANGE` شناخته می‌شوند. در صورتی که تعداد ارقام از پنج رقم بیشتر باشد، مشخص است که عدد از رنج یک سیستم ۱۶ بیتی، خارج است. این عدد در قانون بعدی شناخته می‌شود.

مشخص است که قانون اول نسبت به قانون دوم اولویت بیشتری دارد پس بالای قانون دوم نوشته شده است. زیرا قانون دوم تمام اعداد بدون محدودیت رقم را شامل می‌شود.

۶ - کامنت‌های تک خطی و چند خطی

در کامنت‌های چند خطی رشته باید با زیر رشته `"#!"` شروع شده و با زیر رشته `"#"` پایان یابد. بین این دو زیر رشته، نباید عبارت `"#!"` را داشته باشیم. پس یا نباید عبارت با `"!"` شروع شود یا اگر با `"!"` شروع می‌شود نباید بلافاصله پس از آن `"#"` بیاید. چون کامنت چند خطی است می‌توانیم `"\n"` نیز داشته باشیم. در کامنت‌های یک خطی، رشته باید با کاراکتر `"#"` شروع شده و با کاراکتر `"#"` پایان یابد. پس بین این دو زیررشته نباید کاراکتر `"#"` را داشته باشیم. همچنین چون کامنت تک خطی ست، `"\n"` شامل نمی‌شود. کامنت چند خطی به کامنت تک خطی اولویت دارد و قانون مربوط به کامنت چند خطی بالاتر نوشته می‌شود. همچنین این کامپایلر در صورت تشخیص کامنت، عبارت کامنت شده را تشخیص می‌دهد و توکنی را برای این عبارت چاپ نمی‌کند.

۷ - عملگرها و سایر توکن ها

در این زبان، عملگرهای شرطی، منطقی و محاسباتی داریم. با توجه به عملگری که شناخته می‌شود، با نامی مطابق جدول زیر، شناخته می‌شود.

۸ - جدول

۱	int	TOKEN_INT
۲	void	TOKEN_VOID
۳	main	TOKEN_MAIN
۴	char	TOKEN_CHAR
۵	if	TOKEN_IF
۶	else	TOKEN_ELSE
۷	elseif	TOKEN_ELSEIF
۸	while	TOKEN_WHILE
۹	for	TOKEN_FOR
۱۰	return	TOKEN_RETURN
۱۱	continue	TOKEN_CONTINUE
۱۲	break	TOKEN_BREAK
۱۳	VARIABLE AND FUNCTION NAME	TOKEN_IDENTIFIER
۱۴	CONST_INT	TOKEN_INT_CONST
۱۵	CONST_STRING	TOKEN_STRING
۱۶	<	TOKEN_LESS
۱۷	>	TOKEN_MORE
۱۸	<=	TOKEN_LESS_EQUAL

۱۹	>=	TOKEN_MORE_EQUAL
۲۰	==	TOKEN_EQUAL_TO
۲۱	!=	TOKEN_NOT_EQUAL
۲۲		TOKEN_LOGICAL_OR
۲۳	&&	TOKEN_LOGICAL_AND
۲۴		TOKEN_BITWISE_OR
۲۵	&	TOKEN_BITWISE_AND
۲۶	^	TOKEN_BITWISE_XOR
۲۷	!	TOKEN_NOT
۲۸	+	TOKEN_PLUS
۲۹	-	TOKEN_MINUS
۳۰	*	TOKEN_MULTIPLY
۳۱	/	TOKEN_DIVISION
۳۲	(TOKEN_LEFTPAREN
۳۳)	TOKEN_RIGHTPAREN
۳۴	{	TOKEN_LEFTB
۳۵	}	TOKEN_RIGHTB
۳۶	[TOKEN_LEFT_BRACKET
۳۷]	TOKEN_RIGHT_BRACKET
۳۸	,	TOKEN_COMMA
۳۹	.	TOKEN_DOT
۴۰	=	TOKEN_ASSIGN