



UniRuy & Área 1 | Wyden
PROGRAMA DE CIÊNCIA DA COMPUTAÇÃO
TEORIA DE COMPILADORES

HEBER MAGNO DA SILVA REIS

Teoria de Compiladores: Analisador Léxico

Salvador - Bahia - Brasil

2022

HEBER MAGNO DA SILVA REIS

Teoria de Compiladores: Analisador Léxico

Trabalho Acadêmico elaborado junto ao programa de Engenharia UniRuy & Área 1 | Wyden, como requisito para obtenção de nota parcial da AV1 na disciplina Teoria de Compiladores no curso de Graduação em Ciência da Computação, que tem como objetivo consolidar os tópicos do plano de ensino da disciplina.

Orientador: Prof. MSc. Heleno Cardoso

Salvador - Bahia - Brasil

2022

da Tal, Aluno Fulano

Teoria de Compiladores: Resenha / Mapa Mental / Perguntas

– Aluno Fulano de Tal. Salvador, 2022.
18 f. : il.

Trabalho Acadêmico apresentado ao Curso de Ciência da Computação, UniRuy & Área 1 | Wyden, como requisito para obtenção de aprovação na disciplina Teoria de Compiladores.

Prof. MSc. Heleno Cardoso da S. Filho.

1. Resenha
2. Mapa Mental
3. Perguntas/Respostas (Mínimo de 03 – Máximo de 05)
4. Conclusão

I. da Silva Filho, Heleno Cardoso II. UniRuy & Área 1
| Wyden. III. Trabalho Acadêmico

CDD:XXX

TERMO DE APROVAÇÃO

HEBER MAGNO DA SILVA REIS

TEORIA DE COMPILADORES: ANALISADOR LÉXICO

Trabalho Acadêmico aprovado como requisito para obtenção de nota parcial da AV1 na disciplina Teoria de Compiladores, UniRuy & Área 1 | Wyden, pela seguinte banca examinadora:

BANCA EXAMINADORA

Prof^o. MSc^o. Heleno Cardoso
Wyden

Salvador, 13 de Novembro de 2022

Dedico este trabalho acadêmico a todos que contribuíram direta ou indiretamente com
minha formação acadêmica.

Agradecimentos

Primeiramente agradeço a Deus. Ele, sabe de todas as coisas, e através da sua infinita misericórdia, se fez presente em todos os momentos dessa trajetória, concedendo-me forças e saúde para continuar perseverante na minha caminhada.

E a todos aqueles que contribuíram direta ou indiretamente para a minha formação acadêmica.

"A educaão tem raízes amargas, mas os seus frutos são doces".

Aristóteles.

Resumo

O seguinte trabalho tem como objetivo trazer um breve entendimento dos analisadores léxicos, que é uma das etapas mais importantes dos compiladores, para isso foi realizado pesquisas em repositório de busca, e através dos resultados obtidos selecionou-se dois artigos que foram analisados para trazer os pontos mais importantes a respeito do tema estudado.

Palavras-chaves: Compiladores, fases do compilador, análise léxica, tabela de símbolos, autômato.

Abstract

The following work aims to bring a brief understanding of the lexical analyzers, which is one of the most important steps of the compilers, for that, searches were carried out in a search repository, and through the results obtained, two articles were selected that were analyzed to bring the most important points about the topic studied.

Keywords: Compilers, compiler phases, lexical analysis, symbol table, automaton.

Sumário

1	Análise Léxica	10
1.1	Introdução	10
1.2	Execução/Método	10
1.2.1	Repositório de Pesquisa	10
1.2.2	String de Busca por Repositório	10
1.2.3	Artigos Seleccionados	10
1.2.4	Resenha dos Artigos Seleccionados	11
1.2.4.1	Fases do Compilador	11
1.2.4.2	Análise Léxica	11
1.2.4.3	Implementação de um analisador léxico	12
1.3	Análise de Resultados	13
1.3.1	Perguntas e Respostas	13
1.4	Conclusão	13
	Referências¹	14

¹ De acordo com a Associação Brasileira de Normas Técnicas. NBR 6023.

1 Análise Léxica

1.1 Introdução

Quando estudamos a teoria dos compiladores, um dos assuntos mais vistos são as fases do compilador, onde nela é realizado todo o processo de compilação de um código fonte, dentro dessas etapas existe o analisador léxico que faz algumas das validações do código do programa. Com base nisso o presente trabalho apresenta um resumo do funcionamento e da implementação dos analisadores léxicos.

1.2 Execução/Método

1.2.1 Repositório de Pesquisa

Para a realização do seguinte trabalho, foi utilizado o repositório de busca Google Acadêmico.

1.2.2 String de Busca por Repositório

No repositório de busca Google Acadêmico, foi aplicada a seguinte string de busca: "compiladores AND analisador léxico"

1.2.3 Artigos Selecionados

Na pesquisa realizada foram selecionados os seguintes artigos:

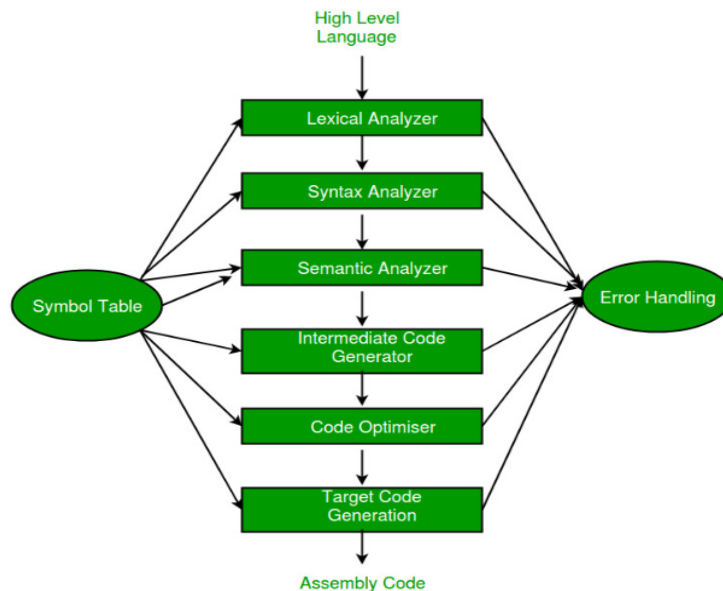
1. DESENVOLVIMENTO DE ANALISADORES LÉXICO E SINTÁTICO PARA PORTUGOL EM JAVA de (PENA, 2010)
2. DESENVOLVIMENTO DE UM ANALISADOR DE CÓDIGO ANSI-C de (LUIZ, 2006)

1.2.4 Resenha dos Artigos Selecionados

1.2.4.1 Fases do Compilador

Basicamente o compilador é composto por duas fases, a fase de análise, que irá fazer as verificações do código fonte, as análises são basicamente três, a análise léxica que é objeto de estudo deste trabalho, a análise hierárquica que categoriza hierarquicamente os tokens e análise semântica que verifica as estruturas do programa analisado. A outra fase do compilador é a chamada fase de síntese que, a partir do código-fonte intermediário obtido na fase anterior, irá criar um programa de destino equivalente.

A seguir uma imagem que representa as fases do compilador:



Disponível em: <https://acervolima.com/fases-de-um-compilador-1/>

1.2.4.2 Análise Léxica

A análise léxica é a primeira fase do compilador. Nessa fase o analisador léxico ou scanner, faz uma leitura do código fonte, passando caractere por caractere e transformam em uma sequência de símbolos léxicos ou tokens.

Os símbolos léxicos são o conjunto de palavras reservadas, funções e constantes de uma linguagem de programação. como por exemplo as palavras reservadas if, for presentes na maioria das linguagens.

Conforme (PENA, 2010) A sequência de caracteres que casa com um padrão de token é chamada lexema, ou seja, numero é o token, 1 o lexema. O analisador léxico,

durante a sua execução, reconhece os lexemas correspondentes ao padrão dos tokens da linguagem. A cada lexema que o analisador encontra, é gerado um token e em seguida inserido na tabela de símbolos. Na tabela de símbolos são armazenadas informações relativas aos tokens, como o lexema, tokens e o valor extraído. Esses dados gerados são importantes para as próximas fases do compilador.

Exemplo de uma tabela de símbolos

Lexema	Token	Atributo (valor do token)	Tabela de Símbolos
if	if		
a	id	1	a
>	Op Log	GT (ou >)	
b	id	2	b
then			
b	id	2	
=	esp	=	
a	id	1	
+	op_mat	+	
2	num	2	

(PENA, 2010)

1.2.4.3 Implementação de um analisador léxico

A implementação do analisador léxico basicamente é feito uma análise gramatical da linguagem de programação C, é feito para cada token uma verificação para definir qual o tipo de identificador pertence, em seguida aciona-se uma rotina específica para sua validação.

Se o token não for pertencente a nenhuma classe ou estiver fora dos padrões da linguagem gera-se imediatamente uma mensagem de erro. As estruturas gramaticais podem ser modeladas através do uso de automatos finitos, onde tem-se um automato correspondente para cada tipo de estrutura.

Um algoritmo correspondente ao autômato deve ser acionado assim que for verificado uma estrutura da linguagem C no código fonte. Por exemplo:

A linha: *float media;*

No momento que o scanner encontra a palavra reservada float, é verificado que se trata de uma declaração de variável, e então, aciona-se o algoritmo para validação sintática da linha (LUIZ, 2006).

1.3 Análise de Resultados

1.3.1 Perguntas e Respostas

1. O que são os símbolos léxicos? **Os símbolos léxicos são o conjunto de palavras reservadas, funções e constantes de uma linguagem de programação.**
2. Qual as duas fases básicas do compilador? **A fase de análise e a fase de síntese**

1.4 Conclusão

Através da realização da pesquisa foi possível compreender um pouco mais sobre toda a estrutura por trás do funcionamento dos compiladores, podendo entender como os compiladores identificam erros no código desenvolvido através da análise léxica que foi estudado no presente trabalho.

Referências¹

LUIZ, T. S. *DESENVOLVIMENTO DE UM ANALISADOR DE CÓDIGO ANSI-C*. 2006. Citado 2 vezes nas páginas 10 e 12.

PENA, M. A. d. O. *DESENVOLVIMENTO DE ANALISADORES LÉXICO E SINTÁTICO PARA PORTUGOL EM JAVA*. 2010. Citado 3 vezes nas páginas 10, 11 e 12.

¹ De acordo com a Associação Brasileira de Normas Técnicas. NBR 6023.