# Trabalho final – Compiladores

**Professor**:

- Leonardo Pol Suárez

**Integrantes**:

- Vitor Bretas Prata

RA: 21215040

- Guilherme Breda

RA: 21268662

# Índice

**2. Compiladores........................................................................................................................ 3**

**2.1 O que é?............................................................................................................................... 3**

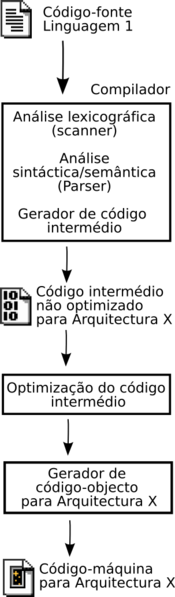
**2.2 Processo de compilação...................................................................................................... 3**

**2.3 Componentes de um compilador....................................................................................... 4**

**3 Bibliografia............................................................................................................................. 5**

# Compiladores

## O que é?

 Um compilador é um programa de sistema que traduz um programa descrito em uma linguagem de alto nível para um programa equivalente em código de máquina para um processador.  Normalmente, o código fonte é escrito em uma linguagem de programação de alto nível, com grande capacidade de abstração, e o código objeto é escrito em uma linguagem de baixo nível, como uma sequência de instruções a ser executada pelo processador.

O compilador, juntamente com o interpretador, é um dos dois tipos mais gerias de interpretador.

Um exemplo das atividades que um compilador deve desempenhar:

\begin{listing}{1}
int a, b, valor;
a = 10; b = 20;
valor = a * (b + 20);
\end{listing}

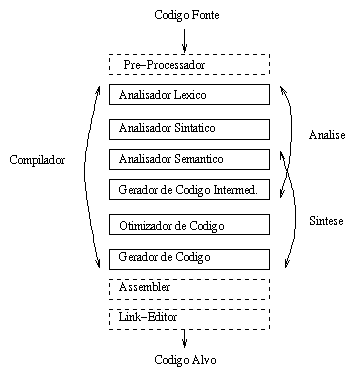
## Processo de compilação

O processo de compilação é composto de **análise** e **síntese**.

Na **análise** tem como o objetivo compreender o código fonte, verificar erros, falhas e inconsistências, para representa-lo em uma estrutura intermediária. Ela é subdividida em analise léxica, análise sintática e analise semântica.

Na **síntese**, ela constrói o código objeto a partir da representação intermediária, ela pode ser composta pelas etapas de Geração de código intermediário, otimização de código e geração do código final, sendo que apenas a etapa de geração de código final é obrigatória.

## Componentes de um compilador

 Um compilador é composto basicamente de cinco unidades, que são o analisador léxico, o analisador sintático, o analisador semântico, o gerador de código e, por último, o otimizador. Estas unidades geram algumas estruturas especiais de dados, que são os tokens, que são as palavras, a árvore sintática abstrata, a tabela de strings e a tabela de símbolos. A seguir temos uma breve descrição de cada uma destas unidades.

* **Analisador Léxico**: Ele é responsável pela identificação de vários tokens existentes no código fonte do programa. Ele é conhecido também pelo nome scanner, sua função é analisar o código fonte a procura de erros de grafia no programa. O analisador léxico também é responsável pela manutenção de tabela de strings, já que é ele quem opera sobre ela.

Suponha a expressão abaixo, descrita em Pascal:

Exp := (A + B) \* 1.5;

Os itens léxicos contidos nesta expressão são:

Exp, :=; (; A; +; B; ); \*; 1,5 e ;.

Os itens léxicos que são reconhecidos pelo analisador léxico são determinados pela gramatica da linguagem fonte. Um erro léxico é gerado caso um item léxico não seja definido pela gramática. Um exemplo seria uma linguagem que só suporte valores inteiros. Então caso haja o valor de entrada 1,5, iria ocasionar um erro.

* **Analisador Sintático**: Conhecido também como parser, o analisador sintático, após a atuação do scanner, entra em ação para verificar se a ordem que os tokens aparecem é gramaticalmente correta. O produto lógico de um parser é uma árvore sintática abstrata, que é algo a formar a seguir, em que os tokens assumem os nós terminais da árvore. Para a realização de analise sintática, baseiam-se em duas abordagens diferentes.
* **Analisador Semântico:** É toda análise feita pelo compilador além da sintática e da léxica. O analisador semântico é responsável por verificar se os aspectos semânticos estão corretos, ou seja, se não existem incoerências quando ao significado das construções utilizadas pelo programador. O analisador semântico depende de uma tabela de símbolos onde estão armazenadas as informações de variáveis declaradas, funções ou métodos, entre outros.
* **Gerador e Otimizador de código:** O compilador, após a verificação de erros sintáticos ou semânticos, realiza uma tarefa de criar o programa objeto que reflete, mediante instruções de baixo nível, os comandos do programa fonte. Há também a otimização do código, onde é aplicada diversas técnicas para a otimização de algumas características do programa objeto, como por exemplo o tamanho e velocidade.

# Bibliografia

* <http://marcogomes.com/blog/2005/compiladores/>
* http://www.oocities.org/walterchagas/compiler.html
* https://pt.wikiversity.org/wiki/Introdu%C3%A7%C3%A3o\_%C3%A0\_Teoria\_dos\_Compiladores/Defini%C3%A7%C3%B5es