

ATIVIDADE 27/09/2023

COMPILADORES



FADERGS

CIÊNCIA DA COMPUTAÇÃO CIÊNCIA DA COMPUTAÇÃO

Rafael Leindecker Nunes de Souza

1. O que é um compilado?

Um compilador é um programa responsável por traduzir o código-fonte escrito em uma linguagem de programação de alto nível para um código de máquina específico para a arquitetura do computador. Ele realiza essa tradução em etapas, convertendo o código legível para humanos em instruções compreensíveis pelo processador.

2. Para que sirva um compilador?

O compilador é utilizado para transformar o código fonte de uma linguagem de programação em um programa desenvolvido. Ele converte o código de alto nível escrito pelo programador em linguagem de máquina, que é realizado e realizado diretamente pelo hardware do computador.

3. Fases de execução de um compilador e suas explicações elaboradas:

Análise Léxica: Esta é a primeira fase do compilador, onde o código-fonte é dividido em tokens. O analisador léxico identifica palavras-chave, identificadores, símbolos e números, além de eliminar espaços em branco e comentários.

Análise Sintática: Nesta fase, o compilador verifica a estrutura gramatical do código, construindo uma árvore sintática para representar a estrutura do programa. Ele valida se a sequência de tokens segue as regras gramaticais da linguagem.

Análise Semântica: Aqui, o compilador verifica o significado e contexto do código. Ele garante a consistência de tipos, verificações de variáveis, verifica se as operações são válidas e se o uso de variáveis está correto de acordo com as regras da linguagem.

Geração de Código Intermediário: Nesta etapa, o compilador cria uma representação do código. Esse código intermediário é mais próximo da linguagem de máquina do que o código fonte original, mas ainda é independente da arquitetura do hardware.

Otimização de Código: O compilador realiza otimizações no código intermediário para melhorar a eficiência do programa. Isso inclui uma reorganização do código para reduzir o tempo de execução ou uso de memória.

Geração de Código: Por fim, o compilador traduz o código intermediário otimizado em código de máquina específico para a arquitetura do designer alvo, gerando o programa detalhado.

4. Qual a diferença entre um compilador e um interpretador?

Compilador: Converte o código fonte inteiro para código de máquina antes da execução. O resultado é um arquivo seguinte independente do código fonte original.

Interpretador: Leia, analise e execute o código linha por linha. Não gera um arquivo seguinte separado; ele executa diretamente o código fonte.

5. Vantagens e desvantagens de compiladores e interpretadores:

Vantagens do compilador: Execução geralmente mais rápida, código otimizado para a máquina alvo, não necessita do código fonte após a compilação.

Desvantagens do compilador: Processo de compilação mais demorado, gera arquivos agrupados maiores, requer recompilação para atualizações no código.

Vantagens do interpretador: Maior portabilidade, facilita a depuração, permite a execução do código sem a necessidade de compilação.

Desvantagens do interpretador: Geralmente mais lento que um código compilado, necessita do interpretador instalado no sistema para execução.

Ambos têm seus usos e vantagens em situações específicas, dependendo das necessidades do projeto e das preferências do desenvolvedor.