**Centro Universitário de Brasília – CEUB**

**Curso: Ciências da Computação (CCO)**

**Disciplina: Compiladores**

**Professor: Leonardo Pol**



**Integrantes:**

1.Nome: João Pedro Issmael Vieira **RA: 22252263**

2.Nome: Kaio Lucas Araújo Silva **RA: 22250719**

Data:01/09/2025

**3ª Produção de Atividade Intelectual – PAI (Conceitos Teóricos para a 1ª Avaliação bimestral)**

**Fonte: AULA Nº04 (Slides) Tradutores de Linguagens**

**Construa sua Produção de Atividade Intelectual – PAI, respondendo às questões abaixo:**

**1.**- Na Compilação, explique o que são denominadas de linguagens de baixo nível?

**R.-**

Linguagem de Baixo nível consiste tipicamente de sequências de zeros e uns.

**2.**- Na compilação, explique o que são denominadas de linguagens de Alto nível?

**R.-**

Linguagem de Alto nível é o meio de comunicação entre o indivíduo que deseja resolver um determinado problema e o computador escolhido para ajudá-lo na solução.

**3.**- Na Compilação, explique o que são programas em linguagens de máquina?

**R.-**

Programas em linguagens de máquina eram a própria linguagem máquina (0’s e 1’s) difícil, longo e principalmente caro de o construir Era também difícil de ser entendido por outros programadores.

**4.**- Na Compilação, explique o que são programas em linguagens de montagem?

**R.-**

Programas em linguagem de montagem são programas que permitem controlar diretamente os registradores e memória do processador, oferecendo um desempenho alto. Esses programas são difíceis de fazer, longos e caros de construir, além disso são difíceis de ser entendidos por outros programadores devido a complexidade da linguagem.

**5.**- Na Compilação, explique o que são tradutores de linguagens de programação?

**R.-**

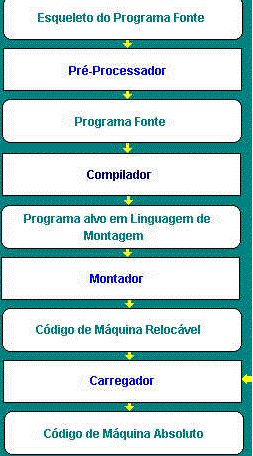
Tradutor é um sistema que aceita como entrada um programa escrito em uma linguagem de programação e produz com o resultado um programa equivalente em outra linguagem.

**6.**- Na Compilação, explique como funcionam os interpretadores?

**R.-**

O interpretador recebe o código intermediário de um programa anteriormente traduzido e produzem o efeito e execução do algoritmo original sem mapeá-lo para a linguagem de máquina.

Contexto de um compilador



**7.**- No Contexto de um compilador explique cada uma das fases do esquema acima?

1. Esqueleto do programa fonte:

R:

Um rascunho ou uma versão inicial e incompleta do código do programa. Ele pode conter diretivas de pré-processamento, mas ainda não está pronto para a compilação.

1. Pré-processador:

R:

Esta é a primeira fase do processo de compilação. O pré-processador processa o esqueleto do código, expandindo macros, substituindo constantes simbólicas e incluindo o conteúdo de arquivos de cabeçalho. O resultado é um programa fonte completo, sem as diretivas de pré-processamento.

1. Programa fonte:

R:

O código do programa em uma linguagem de alto nível que foi processado pelo pré-processador. Ele é o texto completo, pronto para ser lido e analisado pelo compilador.

1. Compilador:

R:

O coração do processo. O compilador traduz o programa fonte de alto nível para uma linguagem de baixo nível. Esse processo é dividido em várias sub-fases, e o resultado final é um programa em linguagem de montagem.

1. Programa alvo em linguagem de montagem:

R:

O resultado da compilação. É um código de baixo nível, mais próximo da arquitetura do hardware, que usa mnemônicos para representar as instruções da máquina. No entanto, ele ainda não é executável diretamente e precisa ser convertido para código de máquina.

1. Montador:

R:

O montador (ou assembler) é a ferramenta que traduz o código em linguagem de montagem para código de máquina relocável. Ele converte os mnemônicos e endereços simbólicos em códigos binários (opcodes) e resolve referências a rotinas e dados dentro do mesmo arquivo. O resultado não tem endereços de memória fixos.

1. Código de Máquina Relocável:

R:

Um arquivo objeto que contém instruções binárias (opcodes) e dados prontos para serem executados pelo processador. No entanto, ele é "relocável" porque os endereços de memória para funções e dados ainda são relativos. Ou seja, ele não sabe exatamente onde será carregado na memória RAM.

1. Carregador:

R: O carregador é parte do sistema operacional. Ele pega o código de máquina relocável e o carrega na memória RAM do computador. Durante esse processo, ele resolve os endereços relativos e os transforma em endereços de memória absolutos, garantindo que o programa seja executado corretamente no local onde foi carregado.

1. Código de máquina absoluto:

R: O programa final, totalmente traduzido em código binário e com todos os endereços de memória resolvidos. Ele está pronto para ser executado diretamente pelo processador. Este é o código que o processador realmente lê e executa, instrução por instrução