

COMO UM COMPUTADOR ENTENDE O PROGRAMA?

Se não for possível que o computador entenda, o programa será apenas um amontoado de palavras. Compilador que faz a análise do programa e traduz o programa para linguagem de máquina.

- O Programa **Fonte** é o que fazemos no Java ou Python e depois é enviado para o compilador. Ex: `file1.cpp` ou `file2.cpp`
- O **Programa Objeto** é a linguagem de máquina, que permite que o computador entenda o que está sendo pedido.

No *compilador* algumas análises que podem ser feitas serão: lexical analyzer ➡ syntax analyzer ➡ semantic analyzer ➡ code generator.

Que serão salvas como `file1.obj` e `file2.obj`

Interpretador puro não gera um programa fonte, ele já vai direto para o compilador.

TRADUÇÃO: 1º ocorre a geração do programa fonte ou objeto. Depois então, 2º ocorre a execução do programa objeto no compilador.

Prós: Execução mais rápida e os programas são menores .

Exemplo: C++ e Java

INTERPRETAÇÃO: o programa fonte é executado diretamente.

Prós: Possui uma maior flexibilidade, mas é mais lento.

Exemplo: Ruby, Python e Javascript

CARACTERÍSTICAS DE UM PROGRAMA

Diretrizes que devemos levar em consideração na hora de desenvolver o programa:

- Legibilidade (Fácil de ler e compreender, além da definição adequada das estruturas, código coerente);
- Redigibilidade (código seja de fácil escrita);
- Confiabilidade (o seu código faz o que se propõe a fazer);
- Custo (usar os recursos sabiamente que o código seja eficiente).

ANÁLISES DE CÓDIGO

Os tipos de análise que o compilador faz no seu código:

- **LEXICAL ANALYZER:** A primeira etapa é conhecida como scanner ou leitura. A função dela é fazer a leitura de letra por letra do código fonte.

Portanto, ela participa, classifica e elimina aquilo que não é necessário.

Participar é onde ela identifica os elementos que são tokens e vai agrupá-los. Esses elementos serão identificadores, palavras reservadas, números e strings. Ou seja, tudo que é relevante para a codificação do programa.

Ela vai eliminar caracteres em branco, comentários...

- **SYNTAX ANALYZER:** a sintaxe de um programa é a forma que ele define, através de palavras reservadas de símbolos especiais, qual que é a estrutura relacionada para a codificação dentro daquela linguagem específica.

Ou seja, ela define a corretude do programa.

Ela depende da linguagem de programação usada, pois cada linguagem de programação possui a sua sintaxe associada.

- **SEMANTIC ANALYZER:** está relacionada ao estudo do significado. Incide sobre a relação entre significantes, como: palavras, frases, sinais e símbolos. É a lógica do programa.

Exemplo errado:

```
If (x = 0)
    printf("O valor e' nulo\n");
```

Corrigindo:

```
If (x == 0)
    printf("O valor e' nulo\n");
```

Pois, apenas um sinal de igual significa atribuição, e dois sinais de igual significa comparação.

PARADIGMAS DE PROGRAMAÇÃO

Paradigma é a forma de resolução de problemas com diretrizes e limitações específicas de cada paradigma utilizando linguagem de programação.

CLASSIFICAÇÕES DE PARADIGMAS:

Orientação à objeto: (Python, Java, C++, Lua)

Procedural: chamadas sucessivas e procedimentos separados. (Fortran e Lua)

Funcional: instruções são baseadas em funções. (Python, Java, Lua e Javascript)

Estruturado: estrutura de blocos aninhados. (C++, Javascript e Java)

Computação distribuída: funções executadas de forma independente. (Ada)

Lógico: (Java).

PARADIGMA ESTRUTURADO :

- Esse conceito traz a ideia de sequência, de decisão e de interação.
- Ênfase em instruções sendo executadas em sequências.
- Programado em C
- Utilizado para problemas simples e diretos. É bom para aprender a programar com ele, pq ele te

ensina a fazer tudo.

PROGRAMAÇÃO ORIENTADA A OBJETO

Esse é um paradigma de programação baseado na utilização de objetos e suas interações. Aqui tudo pode ser objeto: carro, ônibus, gato, cachorro, uma mulher, etc.

Pois, um objeto é descrito por características específicas (O que tenho.), comportamentos (Sou capaz de fazer.) e estado (Como faço.).

Exemplo: Vamos supor que eu tenho ou sou uma caneta, então eu respondo as seguintes perguntas:

O QUE EU TENHO SENDO UMA CANETA:

Modelo
Cor
Carga
Corpo
Tampa
Ponta

O QUE SOU CAPAZ DE FAZER?

Escrever
Desenhar
Rabiscar
Pintar
Destampar

COMO FAÇO?

Tampada
Destampada
Em uso

Portanto:

O que eu tenho ➡ Atributos.

O que sou capaz de fazer ➡ Métodos

Como faço ➡ Estados

PILARES DA ORIENTAÇÃO A OBJETO:

HERANÇA: uma classe filha herda os comportamentos e estado geral da classe mãe. Mas, as filhas podem ter comportamentos e estado específico.