Listagem

- binário
- assembly
- linguagens embarcadas
- linguagens compiladas
- linguagens interpretadas
- linguagens híbridas
- low code
- linguagens nativas
- linguagens dinâmicas
- linguagens estáticas

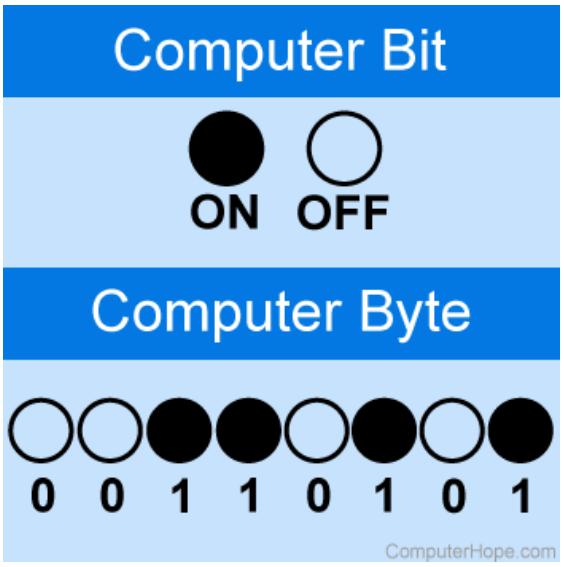
Linguagens

Binário

O código binário é uma linguagem de máquina. Ele é usado para que as máquinas possam interpretar as informações que são enviadas por meio dos comandos – como um texto digitado no teclado, por exemplo, que se transforma em uma letra.

Ao observar o termo “binário”, podemos ter uma ideia do que ele é. “Bi” é igual a dois e “nário” significa número. O código binário é composto por dois números, o 0 e o 1. Isso significa que somente esses dois números são usados para representar todos os caracteres do computador, incluindo as letras, caracteres especiais e números.

Na computação, a informação é processada usando o código binário para converter os valores 0 e 1 em informações. Na imagem abaixo, você pode conferir a representação de 1 bit (on) como ligado e, embaixo, 8 dígitos que representam 1 byte e são formados pelos bits 10101100.



O *binary digit* (dígito binário) é o 1 bit. É a menor unidade de informação do computador. Ela pode ser compartilhada ou armazenada em banco de dados com os valores 0 e 1. Já o byte, consiste no agrupamento (soma) de 8 bits para facilitar o processo das informações que são enviadas ao computador. Além dele, existem ainda: kilobyte (8.192 bits), megabyte (8.388.608 bits), gigabyte (8.589.934.592) e terabyte (8.796.093.022.208

Assembly

Uma linguagem assembly é um tipo de linguagem de programação de baixo nível destinada a se comunicar diretamente com o hardware de um computador. Ao contrário da linguagem de máquina, que consiste em caracteres binários e hexadecimais, as linguagens assembly são projetadas para serem legíveis por humanos.

Linguagens de programação de baixo nível, como a linguagem assembly, são uma ponte necessária entre o hardware subjacente de um computador e as linguagens de programação de nível superior – como Python ou JavaScript – nas quais os programas de software modernos são escritos.

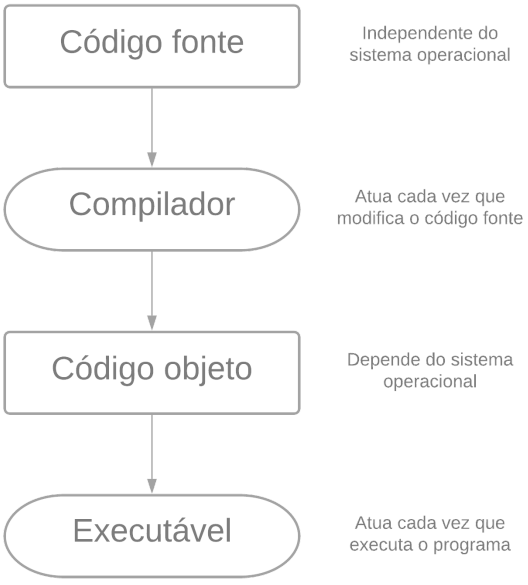
Linguagens Compiladas

(Compilar)
Fazer a conversão de um programa, criado numa linguagem simbólica de nível elevado, para a linguagem de máquina."

Uma linguagem compilada terá o seu código fonte "transformado" para a linguagem de máquina de um determinado processador, digamos que você escreva um hello world em C, Esse código por si só, não passa de um arquivo de texto, para executarmos e ele efetivamente exibir na tela o que foi pedido, precisamos traduzir (compilar) de maneira que as instruções possam ser interpretadas pelo computador, ou qualquer outra máquina para qual esteja programando. A linguagem compilada irá gerar um arquivo binário executável, os famosos ".exe" do windows.

Nesta estratégia, os softwares que realizam a tradução e a execução atuam em fases separadas do processo.

Segue abaixo um fluxograma que ilustre esse funcionamento:



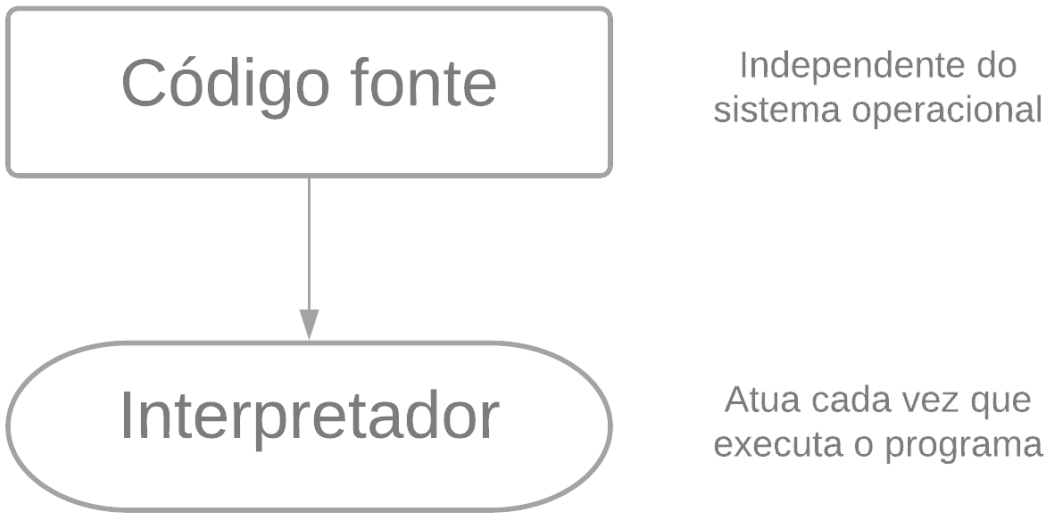
exemplo: **C ,C + +,Rust,Go (Golang) etc.**

Linguangens Interpretadas

Nesta estratégia, o software que comanda a tradução também comanda a execução do programa. Tal software é chamado de interpretador. Ele faz a leitura de cada linha do código fonte, traduz em linguagem de máquina os comandos correspondentes, e executa estes comandos.

As linguagens interpretadas, normalmente não necessitam de um passo específico de compilação para executar, o próprio código fonte escrito será executado através de um interpretador, o JavaScript por exemplo será interpretado pelo navegador, o texto em JavaScript será o seu código fonte e o navegador será o interpretador deste código, também chamamos esse processo como "compilação em tempo de execução", o interpretador vai fazendo uma espécie de tradução para a máquina por baixo dos panos durante o tempo em que está fazendo a leitura do seu código fonte.

Um fluxograma que ilustra o funcionamento de um código interpretado, seria basicamente assim:



Como exemplo de linguagem interpretada,temos o **JS, e além dele temos outras como PHP, R, Ruby,Python,Shell Scripting e etc.**

Linguagens Híbridas

O método híbrido, representa um meio termo entre os compiladores e os interpretadores. Ele traduz (compila) os programas em linguagem de alto nível para uma linguagem intermediária projetada para facilitar a interpretação. Depois, usa um interpretador.

Os sistemas de implementação híbridos são mais rápidos do que a interpretação pura pelo fato de trabalharem com a etapa de compilação, porque as sentenças da linguagem fonte são decodificadas apenas uma vez. Em um interpretador puro, todas as linhas são interpretadas todas as vezes que o programa é rodado. No modelo híbrido, o código compilado pode ser reutilizado, por isso algumas partes do código-fonte são decodificadas apenas uma vez, evitando repetir a mesma operação nas versões seguintes

Exemplp:**Java,C# (C Sharp),Scala,Kotlin e etc.**

Sistemas embarcados

Os sistemas embarcados (embedded systems) são sistemas computacionais baseados em microprocessador projetados para realizar tarefas em tempo real, dentro de um sistema maior.

Esses sistemas também atuam de forma independente, ou seja, sem a necessidade de participarem de um sistema maior.

Eles **são programáveis** ou podem também ter funcionalidades fixas, e essa versatilidade é uma de suas principais vantagens.

Exemplos de sistemas embarcados:**Dispositivos Médicos,Automóveis,Equipamentos Industriais e etc.**

Low code

O low-code é uma abordagem de desenvolvimento de software que requer pouco ou nenhum código para construir aplicações e processos. Uma plataforma de desenvolvimento de low-code utiliza interfaces visuais com funcionalidades simples de lógica de arrastar e largar em vez de extensas linguagens de programação.

Exemplos de Plataformas de Low Code:**OutSystems,Microsoft Power Apps e Appian.**

Linguagem Nativa

Geralmente se refere a linguagens de programação que são compiladas para código de máquina específico da arquitetura do processador alvo.
As linguagens de programação nativas são compiladas diretamente para código de máquina, o que significa que o código resultante é executado diretamente pelo processador do computador, sem a necessidade de uma máquina virtual ou interpretador.

Exemplo:**C++,Assembly e C**

Linguagem Estática

Nas linguagens estáticas, o tipo de dado de uma variável é determinado em tempo de compilação e geralmente não pode ser alterado durante a execução do programa.
Em linguagens estáticas, o compilador verifica os tipos de dados em tempo de compilação. Isso significa que erros relacionados ao tipo de dados, como atribuir um tipo de dado incorreto a uma variável, são identificados antes da execução do programa.

Nestas linguagens geralmente é necessário declarar o tipo de uma variável explicitamente antes de usá-la.

Como o tipo de dados é conhecido em tempo de compilação, o compilador pode otimizar o código de maneira mais eficaz, o que pode resultar em melhor desempenho em algumas situações.

Exemplos de linguagens de programação estáticas incluem **C, C + + e Java.**

Linguagens Dinâmicas

Nas linguagens dinâmicas, o tipo de dado de uma variável é determinado em tempo de execução. Isso significa que uma variável pode armazenar diferentes tipos de dados em momentos diferentes durante a execução do programa.

Em linguagens dinâmicas, a verificação de tipos ocorre em tempo de execução. Isso significa que erros relacionados ao tipo de dados podem ocorrer durante a execução do programa.

As linguagens dinâmicas geralmente oferecem mais flexibilidade em termos de manipulação de tipos de dados. Por exemplo, você pode criar novos tipos de dados em tempo de execução.

Como não é necessário declarar explicitamente o tipo de uma variável, as linguagens dinâmicas tendem a ter menos código boilerplate e podem ser mais concisas.

Exemplos de linguagens de programação dinâmicas incluem **Python, JavaScript, Ruby e PHP.**

Referências:

- https://www.creatio.com/page/pt-pt/low-code
- https://victorvision.com.br/blog/sistemas-embarcados/
- https://www.investopedia.com/terms/a/assembly-language.asp
- https://blog.xpeducacao.com.br/codigo-binario/
- https://www.gabrielmoya.dev/posts/linguagens-compiladas-interpretadas-hibridas

Lara Pires & Abdaisy Conceição