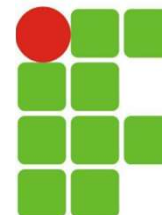


Linguagens de programação

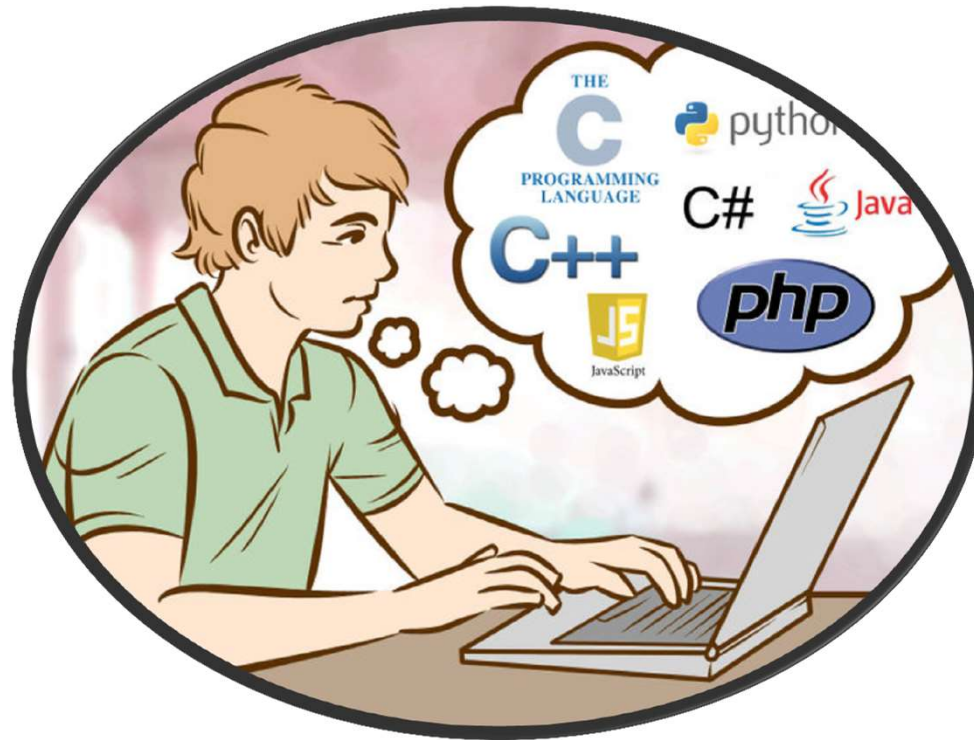


O que pretendemos:

- Apresentar os conceitos, classificação e tipos de linguagens de programação;
- Exemplificar tradutores e ferramentas de programação;



Linguagens de programação



Conceito

Linguagem de programação é:

Um conjunto de **regras sintáticas e semânticas** usadas para definir um algoritmo, que resulta no programa de computador.

- É o meio pelo qual podemos indicar os “passos” que devem ser realizados pelo computador para resolver problemas.
- Os algoritmos escritos numa determinada linguagem são traduzidos para linguagem de máquina para que se tornem portáteis.

Características

- Permite especificar precisamente sobre quais dados um computador vai atuar, como serão armazenados ou transmitidos e quais ações devem ser tomadas sob várias circunstâncias.
- Para que o computador execute o algoritmo proposto, as operações devem ser transcritas para uma **linguagem que a máquina consiga compreender**.
- Os computadores só podem executar algoritmos expressos em linguagem de máquina, que se constituem de um conjunto de instruções capazes de ativar diretamente os dispositivos eletrônicos do computador.

Características

Como funcionam as linguagens de programação?

- São compostas por um grupo de elementos e regras que permitem a construção das instruções utilizadas para resolver os problemas computacionais;
- Com elas, construímos programas que devem ser, posteriormente, transformados em instruções de Linguagem de Máquina;
- Para essa transformação, cada linguagem possui um programa-suporte denominado, genericamente, de TRADUTOR;
- Podem ser classificadas em: linguagens de alto nível (linguagem de máquina, linguagem de montagem), e linguagens de alto nível.

Classificação

Linguagem de máquina (1ª geração)

- São instruções de baixo nível utilizando sequências de *bits*, com manipulação direta por parte dos operadores.

Linguagem de montagem (2ª geração)

- É necessário um interpretador para transformar suas funções em linguagem de máquina. Ex: Assembly.

Linguagem de alto nível

- Utilizam comandos e/ou expressões aritméticas que são quebrados em instruções menores, que por sua vez são transformados em linguagem de máquina.

C# JAVA
Delphi .NET
COBOL
Pascal C++
C

Características

Linguagem de Máquina:

- Diferente para cada tipo de computador, dependendo de sua arquitetura;
- Extremamente rudimentar, pois até as operações mais simples têm que ser expressas em termos de registros, acumuladores e outros dispositivos;
- Totalmente expressa em forma numérica - sistema de numeração binário (0 e 1) ou hexadecimal;
- De difícil aprendizado, por isso surgiram as linguagens de programação de alto nível.

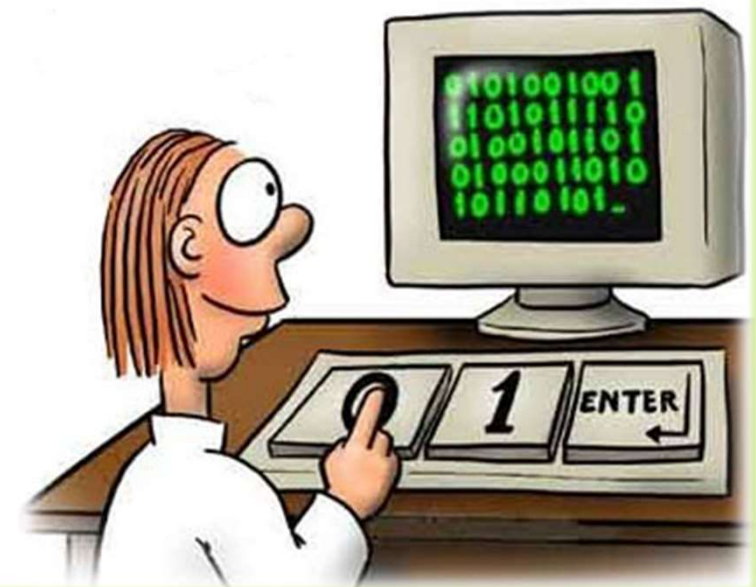


Características

Linguagem de Máquina:

| Endereço | Código da Instrução | Mnemônico da Instrução |
|----------|----------------------|------------------------|
| 0000 | : 111 00000000 1111 | (LG E15) |
| 0001 | : 111 00000000 1110 | (LG E14) |
| 0010 | : 101 1111 1100 0110 | (SE E15 = E12 E6) |
| 0011 | : 000 1110 1101 1101 | (SOM E14 E13 E13) |
| 0100 | : 000 1011 1100 1100 | (SOM E11 E12 E12) |
| 0101 | : 111 00000001 0010 | (VÁ E2) |
| 0110 | : 111 00000010 1101 | (IMP E13) |
| 0111 | : 111 00000011 0000 | (PARE) |
| 1000 | : ? | |
| 1001 | : ? | |
| 1010 | : ? | |
| 1011 | : 000 00000000 0001 | (Valor 1) |
| 1100 | : 000 00000000 0000 | (Valor 0) |
| 1101 | : 000 00000000 0000 | (Valor 0) |
| 1110 | : ? | |
| 1111 | : ? | |

Pode-se editar o código de máquina usando editores hexadecimais. Com esses programas, pode-se ver o código não em binário, mas em hexadecimal.



Veja aqui outros exemplos de código:

www.dcc.fc.up.pt/~nam/aulas/9900/ic/slides/sliic9918/

Características

Linguagens de programação de montagem:

- Conhecida como Linguagem Assembler ou Linguagem de Montagem, ou ainda, Linguagem Simbólica;
- Utiliza números binários, hexadecimais, alguns símbolos e letras para compor os programas. Está muito próxima da Linguagem de Máquina, onde cada instrução simbólica corresponde, praticamente, a uma instrução de máquina;
- Para transformar o programa escrito em Linguagem Assembler em código de máquina executável, é utilizado um programa-suporte denominado de MONTADOR (daí, o termo montagem).

EXEMPLO

Adição: ADD, ADC, INC, XADD, AAA e DAA;
Subtração: SUB, SBB, DEC, AAS e DAS;
Multiplicação: MUL, IMUL e AAM;
Divisão: DIV, IDIV e AAD.

Características

Linguagens de programação de alto nível:

- Utilizam notações matemáticas e grupos de palavras para representar as instruções de máquina, tornando o processo de programação mais próximo do entendimento humano.
- Muitas delas foram desenvolvidas para atender os problemas de áreas de aplicação específicas, como, p. ex., linguagens para aplicações comerciais, científicas, de ensino etc.
- A primeira foi desenvolvida em 1957 → [Fortran](#) (Formula Translator) → para aplicações científicas e de engenharia.
- Estão distribuídas em gerações (3ª, 4ª e 5ª).

Classificação

Linguagem de 3ª geração

- Foi projetada para ser facilmente entendida pelo ser humano, incluindo coisas como variáveis com nomes;
- Ideal para objetivos específicos, como o desenvolvimento de *softwares* comerciais de negócios;
- A maioria das linguagens de terceira geração suportam programação estruturada.
- Ex: [Fortran](#), [ALGOL](#), [COBOL](#), [Pascal](#) e [C](#)

PROGRAM OLA
PRINT *, "Olá Mundo!" ! ← usando comando PRINT
END




Classificação

Linguagem 3ª geração – baseada em objetos

- Os objetos consistem em atributos (dados) e comportamento/métodos (funções), assim como o relacionamento com outros objetos;
 - Ex: objeto ALUNO é matriculado em DISCIPLINA, que é ministrada por PROFESSOR;
- Muito utilizadas na criação de paginas Web;
- São alguns exemplos: [C++](#), [Delphi](#) (sucessor da Pascal), [Visual Basic .NET](#) (sucessor da Basic e do Visual Basic), [Perl](#), [Python](#), [PHP](#), [Java](#).

```
public class AloPessoal {  
    public static void main(String args[]) {  
        System.out.println("Alo pessoal!");  
    }  
}
```



Classificação

Linguagem de 4ª geração

- Utiliza instruções naturais (próximas da linguagem humana) e não procedurais;
- Ideal para objetivos específicos, como o desenvolvimento de *softwares* comerciais de negócios;
- Tem flexibilidade limitada.
- Ex: [ABAP](#), [FOCUS](#), [NATURAL](#), [MATLAB](#), [SQL](#)

```
UPDATE Aluno  
SET conceito = 'Aprovado' WHERE  
disciplina = 'INF025'
```



Classificação

Linguagem de 5ª geração

- São linguagens voltadas à Inteligência Artificial (Prolog) e as linguagens funcionais (Lisp)
- **Prolog** se enquadra no paradigma de Programação em Lógica Matemática, sendo associada com a inteligência artificial e linguística computacional.
- **Lisp** foi projetada primariamente para o processamento de dados simbólicos. Ela é uma linguagem formal matemática. Durante os anos de 1970 e 1980, tornou-se a principal linguagem da comunidade de inteligência artificial, tendo sido pioneiro em aplicações como administração automática de armazenamento, linguagens interpretadas e programação funcional.

Tradutores de Linguagens de programação

Os *softwares* consistem em conjuntos de instruções escritas em uma linguagem própria, que são traduzidos para uma linguagem de máquina específica.

Compiladores → cria um arquivo executável.

Ex.: [C](#), [Java](#), [Delphi](#), [Visual Basic](#)



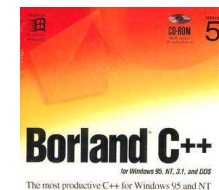
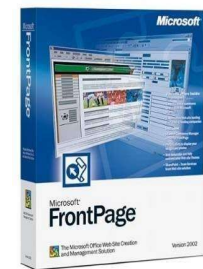
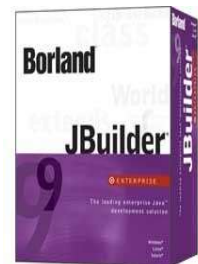
Interpretadores → traduz linha a linha do programa e não cria arquivo executável.

Ex.: [HTML](#), [XML](#), [Java Applet](#), [VB Script](#), [Java Script](#)



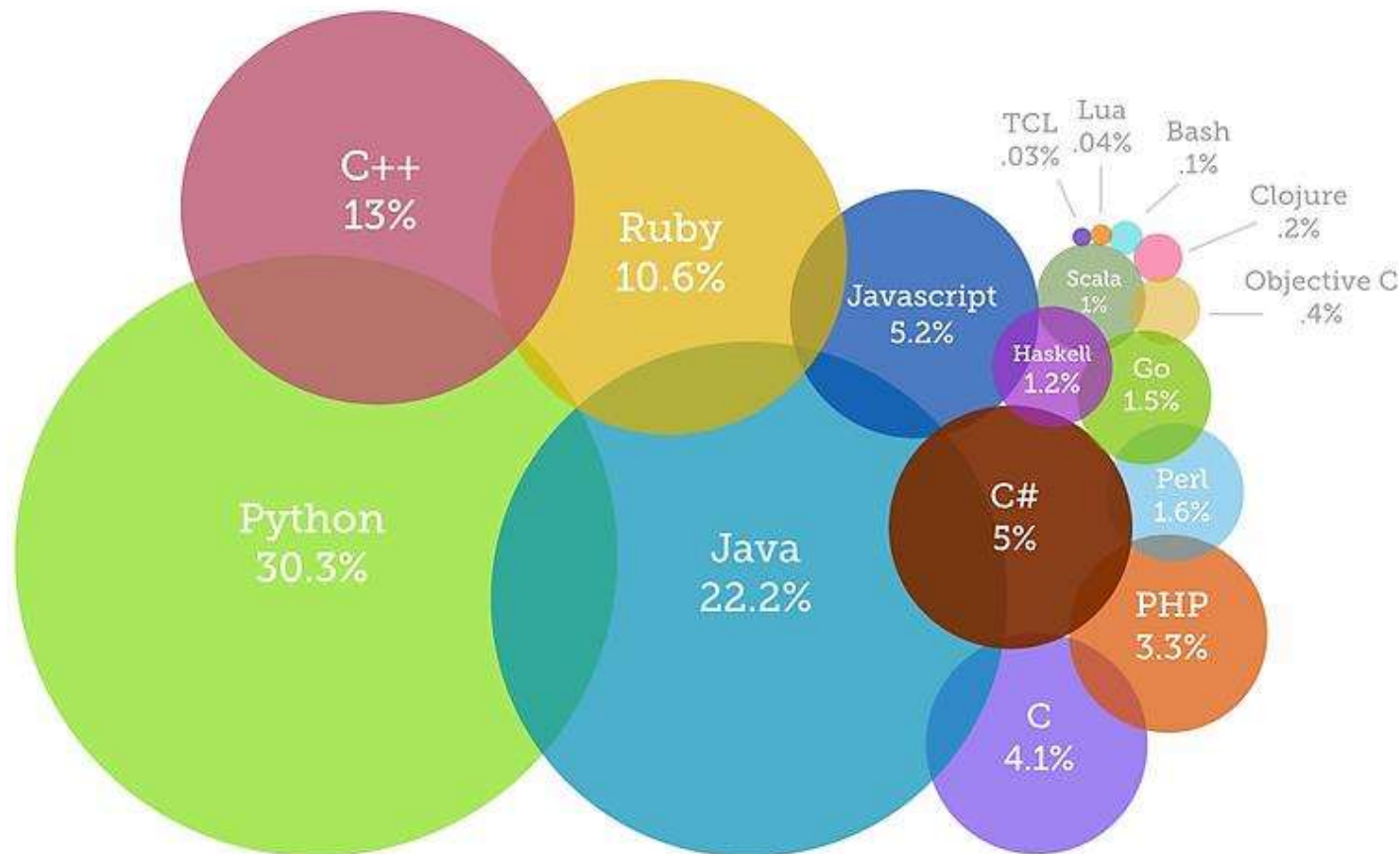
Ferramentas de programação

- É uma interface de programação.
- Auxilia o programador a depurar (identificar/resolver) erros.
- Melhora a produtividade da programação.
- Facilita a documentação dos sistemas.
- São alguns exemplos: [Borland C++](#) (C), [JBuilder](#) (Java), [Delphi](#) (Delphi), [Visual Studio](#) (Visual Basic), [FrontPage](#) (HTML), [Microsoft .NET](#)
- Incluem: geradores de código, biblioteca de objetos e funções, ferramentas de prototipação



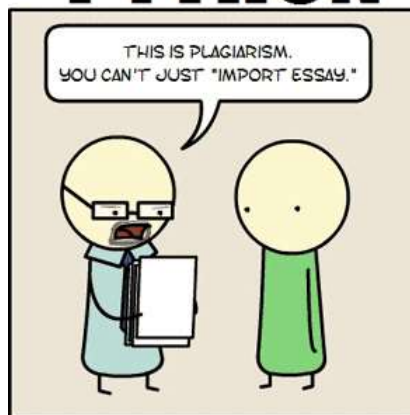
Linguagens de programação populares

Most Popular Coding Languages of 2014

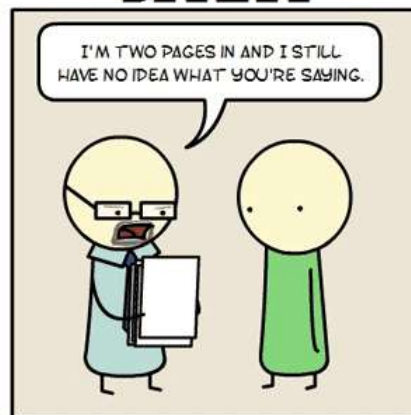


Linguagens de programação populares

PYTHON



JAVA



C++



UNIX SHELL



ASSEMBLY



C



LATEX



HTML



Compilação e execução

- Para executarmos um programa escrito numa linguagem de alto nível é preciso primeiro traduzir o código-fonte para código-objeto.



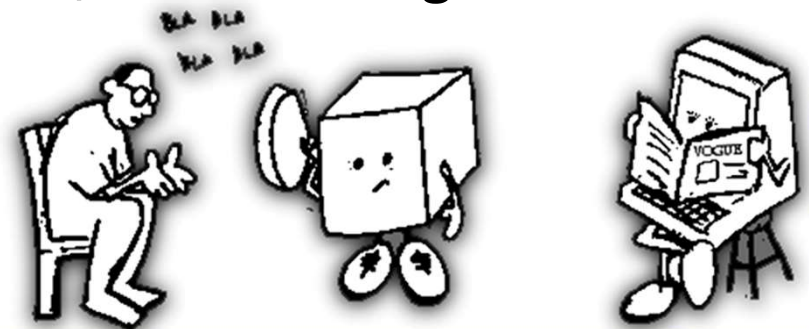
- O processo de tradução pode dar-se em tempo de execução, caso a linguagem use um **interpretador** (traduz e executa instrução a instrução → ex: HTML), ou todas as instruções podem ser traduzidas antes que se inicie a execução do programa, o que ocorre no caso de linguagens que usam tradutores do tipo **compilador**.

Compilação e execução

- **Código-fonte:** não é executável diretamente pelo processador → permite apenas que o programador consiga definir o programa em uma forma legível aos humanos.



- **Compilador:** transforma um conjunto de símbolos em outro, obedecendo a uma série de regras sintáticas e semânticas; traduz todo o código-fonte de programas escritos numa linguagem de alto nível em código-objeto antes da execução do programa. O código-objeto é o código de máquina, ou alguma variação do código de máquina.



Compilação e execução

- **Código-objeto:** é o código produzido pelo compilador; é uma forma intermediária, similar à linguagem de máquina do computador. Apesar de estar representado em binário, não é executável diretamente pelo processador, pois normalmente, o código-objeto referencia partes de programa que não estão necessariamente definidas no mesmo arquivo que o gerou, por exemplo, arquivos de bibliotecas de sub-rotinas.
- **Editores de ligação (ou linkeditores):** reúne módulos compilados e arquivos de dados para criar um programa executável. Os linkeditores têm também outras funções, como a criação de bibliotecas.



Abordagens de programação

Programação estruturada

- Utiliza basicamente três estruturas: decisão, repetição e sequência, inseridas em sub-rotinas e funções.

Programação modular

- O desenvolvimento das rotinas de programação é feito através de módulos, que são interligados entre si através de uma interface comum.

Programação orientada a objetos

- É baseada na composição e interação entre diversas unidades de *software* chamadas de objetos, interrelacionados pelo mecanismo de herança e pela troca de mensagens .

Programação linear

- Utilizada para resolver problemas de otimização.



QUE TAL UM PAUSA,
PARA GANHAR
ENERGIAS E
CONTINUAR?!

How users see the programmers



How programmers see the users



kill your time on 9GAG.COM

Vamos exercitar um pouco?

