

Linguagens de Programação

Prof. Edson David Pereira

Linguagens de Programação

Seria tão bom se pudéssemos conversar com o computador utilizando o nosso próprio idioma!

Ele, porém, não nos permite essa liberdade de expressão, e exige, para essa comunicação, linguagens específicas, que devem ser utilizadas somente dentro de padrões bem definidos.

Assim mesmo, sonhamos com o dia em que isto será possível; com o dia em que poderemos dialogar livremente com o computador.

Linguagens de Programação

Uma linguagem natural, como as utilizadas pelos seres humanos, só é possível porque nós somos dotados de uma estrutura que torna possível este meio de comunicação. Nosso cérebro possui todos os "circuitos" e "ligações" necessárias para que funcionemos dessa forma, e se assim não fosse, a culpa não seria nossa, mas de quem projetou o corpo humano.

Por aí podemos perceber que, se não podemos conversar com o computador como gostaríamos, a culpa não é dele, mas nossa, porque fomos nós que o projetamos. Suas limitações são, portanto, apenas um reflexo das nossas limitações, mas nós também estamos evoluindo do ponto de vista cultural e tecnológico e, a cada dia que passa, conseguimos projetar novos equipamentos mais potentes e capazes.

Assim, a finalidade de qualquer linguagem de programação é a de propiciar comunicação entre nós, seres humanos, e o computador, ressalvadas as limitações já existentes, tanto as nossas quanto as dele.

Essa tarefa já foi difícil no passado, e houve época em que conversar com o computador era uma tarefa sonhada por muitos, mas realizada apenas por uns poucos especialistas.

Hoje, decorridas apenas umas poucas décadas desde que os primeiros computadores surgiram, muitas linguagens foram criadas e aperfeiçoadas, algumas delas muito fáceis de se aprender e manipular.

Linguagens de Programação

Tais linguagens podem ser classificadas de várias formas, tais como: **de baixo ou alto nível**; de uso geral ou especial: orientada para processos ou para problemas; compiladas ou interpretadas; estruturadas ou não-estruturadas, interativas ou não-interativas.

O **BASIC**, é uma linguagem de alto nível, de uso geral, orientada para processos, geralmente interpretada, não-estruturada e interativa.

Vamos abordar as várias formas de classificar as linguagens de computação, dar um passeio pelas linguagens mais difundidas e tentar visualizar, neste contexto, o que o futuro nos reserva. Existe compilador que transforma a codificação em **Basic** interpretado em compilado. https://www.tutorialspoint.com/execute_cbasic_online.php

Linguagens de baixo e de alto nível

Por razões óbvias, a indústria de computadores está cada vez mais interessada na produção de equipamentos fáceis de serem utilizados pelas pessoas em geral. Quanto mais prático o seu uso, tanto maior se torna a expectativa de ter um deles vendido a cada cidadão. É nesse ponto que entra o valor de uma linguagem de programação.

As de **baixo nível** são mais voltadas para o computador, enquanto que as de **alto nível**, para o usuário.

Os primeiros computadores somente entendiam a linguagem de mais baixo nível, que é o código de máquina. O grande problema é que só especialistas conseguiam operar os equipamentos.

Pesquisadores, então se voltaram para o desenvolvimento de outras linguagens, visando facilitar o diálogo entre leigos e computadores.

Assim as linguagens de baixo nível permitem aos programadores acessar de modo direto o equipamento no que diz respeito às locações de memória, registros; portas de entrada e saída de informações.

```
100 print "Informe  
seu nome:"  
110 input nome$  
120 print "Muito  
prazer " + nome$
```

Linguagens de Programação

Já as linguagens de alto nível, apesar de forçar seus programadores a trabalhar com algumas abstrações tais como arquivos, variáveis, etc., não os obriga a se envolver com aspectos de hardware dos equipamentos.

Nesse caso, a questão é idiomática, e todo o trabalho se resume em aprender um novo idioma.

BASIC, FORTRAN, PASCAL, são exemplos de linguagens de alto nível. Algumas linguagens, como o **FORTH** e o **C**, combinam características de linguagens de baixo e de alto nível.

Linguagens para uso geral e uso especial

Algumas linguagens foram criadas com o propósito específico de ensinar programação, controlar robôs, fazer gráficos, etc., enquanto outras foram desenvolvidas visando a utilização de equipamentos de modo geral.

BASIC, PASCAL, LISP, FORTRAN, COBOL, APL, são exemplos de linguagens criadas para uso geral. **FORTH, C**, e outras foram criadas para a programação de sistemas operacionais. **PROLOG** foi idealizada para a programação lógica. Há ainda outros exemplos de linguagens criadas para usos especiais, mas é importante que se diga que a maioria dessas linguagens acabou por exceder o plano de seus programadores.

Linguagens de Programação

Linguagens orientadas para processos ou para problemas

São orientadas por processos as linguagens que o usuário especifique uma série de operações a serem executadas em uma certa ordem.

Tais linguagens foram criadas independentemente dos equipamentos em que serão utilizadas. **Basic e Fortran** são exemplos de linguagens desse tipo.

Já as linguagens orientadas para problemas são as que se prestam à solução de problemas específicos, como o [dbase](#) e o **Fryday**.

Linguagens interpretadas e compiladas


Um computador só entende mesmo instruções em código de máquina. Isto significa que para que ele entender um programa escrito em uma linguagem de alto nível, este programa deve ser traduzido para o seu código de máquina. Para isto ele deve estar equipado com um interpretador ou um compilador. Um interpretador traduz o programa para o código de máquina à medida em que este vai sendo executado, enquanto que um compilador executa a tradução antes do processamento, e você pode guardar a versão compilada do programa para futuras execuções. **Fortran, Cobol Pascal** são linguagens compiladas; **Basic** existe nas duas versões; **APL** é, em geral, interpretada.

Linguagens estruturadas e não-estruturadas

As linguagens estruturadas são as que exigem um padrão lógico bastante rígido na formulação dos programas. Aqui, nem tudo que é lógico é permitido para as interligações das partes de um programa. Perde-se um pouco de abrangência com as dificuldades que envolvem a tarefa de se produzir programas bem estruturados, mas ganha-se em clareza, facilidade de correção e de atualização do programa.

A maioria das linguagens permite a geração de programas bem estruturados, e o grau de estruturação vai depender da qualidade do programador. Algumas linguagens mais modernas, como o **Pascal**, não permitem que os dados sejam introduzidos, a não ser de forma logicamente estruturada.

[dbase](#)




Byte: restantes: 3815

CURSOR <-- --> Carac: ← → Plura: Home End Pan: ^← ^→	INSERT Carac: Ins Campo: ^N Help: F1	DELETE Carac: Del Plura: ^Y Campo: ^U	Acima 1 Campo: ↑ Abaixo 1 Campo: ↓ Sai/Grava: ^End Aborta: Esc
--	--	---	---

Campo	Nome	Tipo	Larg.	Dec
1	CODIGO	Numerico	5	0
2	NOME	Character	40	
3	ENDERECO	Character	40	
4		Character		

CREATE ||<C:>||TABELA ||Campo: 4/4 ||NunCaps

Informe o nome do campo.
Nomes de campos comecam com letra e devem conter letras, digitos e simbolos.

 Datassette

[Visitar](#)

dBase III Plus 1.0 Português for MS-DOS |
Datassette

Linguagens de Programação

Linguagens interativas e não-interativas

Linguagens interativas são as que permitem a comunicação direta com o computador, não só quando estamos executando o trabalho de digitação, mas também quando executamos um programa. Tais linguagens permitem o desenvolvimento mais rápido de programas que, ao serem rodados, têm os resultados analisados, os erros localizados, e rapidamente corrigidos.

Já as linguagens não-interativas exigem que o programa todo seja digitado, depois compilado, para que os erros sejam localizados.

Linguagens de Programação

Fortran (1954)

O **Fortran** teve a honra de ser a primeira linguagem de alto nível. Seu nome foi extraído das primeiras sílabas de "Fórmula Translation" (tradução de fórmula), e sua primeira versão surgiu em 1954, mas só em 1957 tivemos uma versão desenvolvida pela **IBM**, o ralk que muito contribuiu para a difusão dessa linguagem.

Seu objetivo inicial era o de poder expressar qualquer problema em computação numérica, especialmente em problemas envolvendo agrupamentos de fórmulas e muitas variáveis.

Admitia-se, inicialmente, que esta linguagem não fosse a ideal para resolver problemas de outra natureza. O fato, porém, é que o **Fortran** conquistou espaço também em áreas como o comércio e a educação, talvez devido à sua popularidade nas universidades. Quase sempre compilado, a sua gramática, os seus símbolos e regras e a sua sintaxe são derivadas de convenções usadas na matemática.

Apesar de ser mais adequada a cálculos, foram escritos diversos programas para processamento de gráficos e para administração de bancos de dados nessa linguagem. Uma de suas grandes contribuições, foi o fato dela ter inspirado a criação do **Basic**, a mais popular de todas as linguagens, e que, em essência, nada mais é do que uma versão simplificada do **Fortran**.

Os programas em **Fortran** são escritos num formato rígido, o que é uma marca do tempo em que se fazia uso intenso dos cartões perfurados com 80 colunas.

Aspectos positivos

- Fórmulas algébricas são escritas praticamente do modo como as utilizamos na matemática;
- Há possibilidade de utilização de funções exponenciais e transcendentais (trigonométricas e hiperbólicas);
- O acervo de programas de domínio público, para a resolução de praticamente de todos os problemas científicos, é muito grande;
- Sua estrutura modular permite re-utilizar suas rotinas, que podem ser compiladas e testadas separadamente.

Linguagens de Programação

Cobol (1960)

O **Cobol**, "**Common Business Oriented Language**", conforme o seu nome indica, foi desenvolvido tendo em vista a computação empresarial e o processamento de dados comerciais. Um dos pontos interessantes dessa linguagem é que seus programas podem ser explicitados em palavras naturais, precisas e facilmente aprendidas, podendo ser lidas por usuários leigos em computação.

Apesar disso, a estrutura de um programa em Cobol é muito rígida. Todo programa deve ter divisões, que se dividem em seções, e que, por sua vez, se dividem em parágrafos onde são escritas as sentenças.

Todo programa em Cobol possui quatro divisões, cada uma com uma função específica :

- Identification Division;
- Environment Division;
- Data Division; e
- Procedure Division.

Aspectos positivos

- É muito parecida com o inglês;
- São permitidas até 30 caracteres nos nomes de variáveis;
- É de fácil manutenção;
- Permite o manuseio de grandes quantidades de informações;
- Tem capacidade sofisticada de formatação de dados para impressão;

O **Cobol** é uma linguagem altamente especializada, e aí reside o seu trunfo. Não tendo sido criada para fins gerais, acaba concentrando todo o seu potencial na consecução do seu objetivo específico; o controle da função empresarial e do mundo dos negócios.

Linguagens de Programação

Basic (1964)

A linguagem **Basic**, criada em 1964, tem seu nome derivado de “Beginner’s All-purpose Symbolic Instruction Code”, instruções simbólicas de propósito geral para principiantes.

Incorporando aspectos essenciais do **Fortran** e do **Algol**, o **Basic** foi projetado como uma linguagem interativa, de alto nível que, em virtude de sua popularidade, hoje vem implementada em quase todos os microcomputadores, na versão interpretada.

Isto significa que, além de escrever os seus próprios programas com facilidade, o usuário leigo e o "hobbysta" podem facilmente modificar programas escritos por outras pessoas.

O **Basic** tem sido, ao lado dos aplicativos fáceis de serem utilizados, uma das molas propulsoras da indústria de computadores no seu objetivo de colocar um computador em cada lar. Como linguagem de aplicação ela também tem se distinguido, com aplicativos que vão desde o ambiente doméstico até o semi-profissional, apesar de suas desvantagens, visíveis quando se deseja escrever um programa muito longo.

Características principais

- Todas as suas linhas de programação são numeradas, indicando a ordem de execução;
- É muito semelhante ao **Fortran** na sua forma de codificar expressões aritméticas;
- Quase todas as versões são capazes de manipular variáveis alfanuméricas e funções trigonométricas.
- Sua versão padrão, desenvolvida pela **Microsoft**, possui todos os comandos estruturados.

Linguagens de Programação

Basic (1964)

Vantagens

- É fácil de se utilizar.
- Mensagens de erro são simples, mas completas.
- Estimula a atividade de desenvolvimento de programas.

Desvantagens

- A velocidade de processamento é muito baixa;
- É difícil de se traduzir um programa de uma, entre as diversas versões (dialetos) de **Basic**, para outra.

LOGO (1967)

Desenvolvida por **Seymour Papert**, **Logo** é uma linguagem de programação de alto nível, muito didática, de fácil emprego, e que permite ensinar os fundamentos da programação até mesmo ao público infantil. Experiências realizadas nos Estados Unidos e na Europa demonstraram que os princípios que norteiam o **Logo** podem ser aprendidos por crianças de até três anos de idade.

Desenvolvido no **MIT**, a linguagem é simples e poderosa, permitindo a geração de música, a movimentação de um robô chamado "**turtle**" (tartaruga), desenhando figuras, e a criação da imagem do desenho.

Não se trata de uma linguagem semelhante às que normalmente são usadas em processamento de dados.

Aqui as crianças não fazem programas, mas ensinam a tartaruga, um simpático bichinho que vive na tela do computador.

A programação é aprendida intuitivamente, e o mesmo processo se dá com outros conceitos, principalmente os ligados à matemática: coordenadas cartesianas, razões de proporcionalidade, ângulos, comprimentos de segmentos e de arcos, todos apresentados informalmente, e compreendidos sem dificuldade.

Linguagens de Programação

Pascal (1971)

Pascal é uma linguagem que nasceu de um projeto do **Prof. Niklaus Wirth**, da **Escola Politécnica** de Zurique e recebeu este nome em homenagem ao filósofo e matemático do século XVII, **Blaise Pascal**.

Seu objetivo inicial era o de criar uma linguagem simples para ser usada em ambientes educacionais. Seu idealizador acreditava que o ensino da programação deveria ser abordado de forma sistemática e, com essa finalidade, traçou as bases dessa linguagem. O resultado foi uma linguagem simples e poderosa, realmente providencial, pois a criação de grandes programas tem sido um dos maiores problemas de desenvolvimento de software.

Há sistemas que requerem o esforço de programadores e podem levar anos para serem projetados, implementados, testados e depurados.

A linguagem **Pascal** alivia estes problemas, ao desenvolvedor de maneira fácil programas compreensíveis e de simples manutenção.

Os programas em **Pascal** são processados muito mais rapidamente que os programas em **Basic**, e se constituem sempre de três seções:

o header, a seção de declaração e o corpo do programa. O header é constituído pela palavra **PROGRAM** e o nome do programa; a seção de declaração é uma listagem das variáveis, dados e sub-programas usados no programa; o corpo do programa é constituído de declarações que dão conteúdo às variáveis.

Pascal tem sido adotada por um grande número de universidades como a linguagem de programação dos primeiros cursos. Além disso, ela tem sido utilizada por algumas indústrias no desenvolvimento de seus sistemas, e escolhida como linguagem-base por alguns fabricantes de microcomputadores que já produzem equipamentos com **Pascal** implementada de fábrica.

Indiscutivelmente, porém, seu valor reside de modo bastante acentuado no ensino de programação: quem aprende a programar em **Pascal**, aprende a programar de modo bem estruturado. **Pascal** tem servido também como modelo para o estudo e desenvolvimento de novas linguagens de programação.

Linguagens de Programação

Prolog (1970)

A linguagem **Prolog**, cujo nome deriva de "**Programming in Logic**" (*Programação em Lógica*), é uma linguagem simples e poderosa, com raízes na Lógica-matemática, originalmente desenvolvida para facilitar a demonstração automática de teoremas.

Muito utilizada na pesquisa em Inteligência Artificial, ela teve sua origem na Universidade de **Aix-Marseille, na França**, em um departamento dessa especialidade. Desde 1972, entretanto, muitos outros centros a adotaram, incluindo o Departamento de Inteligência Artificial da Edinburgh University.

A linguagem **Prolog** também é utilizada para raciocínios dedutivos.

C (1972)

Esta é uma linguagem de programação estruturada que mais se presta ao desenvolvimento de software básico, tais como editores, sistemas operacionais, etc.

Desde a sua criação em 1972, no **Laboratório Reli**, esta linguagem sempre esteve associada ao sistema operacional Unix, e hoje começa a conquistar seguidores entre os usuários de micros. Descrita, em princípio, como um tipo de **Algol** implementável e um **Pascal** que não tem medo de sujar as mãos, a linguagem **C** produz código que se aproxima da linguagem de máquina em densidade e eficiência, permitindo acesso ao hardware que estiver sendo utilizado, já que possui um certo número de instruções equivalentes às instruções em código **Assembly**.

A linguagem **C** também oferece também características de linguagem de alto nível, permitindo a programação estruturada.

Linguagens de Programação

C++ (1983)

Considerada a quarta linguagem de programação mais popular do mundo, segundo o Índice Tiobe, a linguagem **C++** é muito utilizada para o desenvolvimento de sistemas de alta performance, jogos, interfaces gráficas e no ensino sobre orientação a objetos na programação.

Considerada como uma extensão da linguagem **C** por utilizar comandos e estruturas dessa linguagem, essa linguagem proporciona maior desempenho, fácil aprendizado, é **open source** e continua em constante melhorias em suas novas versões.

As interfaces gráficas **DirectX**, **OpenGL**, **SDL** e **Allegro** a utilizam pelo desempenho e interação com a máquina.

Conheça algumas das empresas e sistemas que utilizam essa linguagem:

Software	Empresa
Chromium	Google
Photoshop e Acrobat Reader	Adobe
MySQL	Oracle Corporation
SQL Server	Microsoft
Navegador Mozilla	Mozilla Foundation
Winamp	Nullsoft
SpaceX	NASA
Word, Excel, Access, Power Point, Outlook	Microsoft

Linguagens de Programação

Preciso aprender C para aprender C++ e C#?

Aprender a linguagem **C** não é um precedente para começar a utilizar as linguagens **C++** e **C#**, mas ela possui recursos, funções e estruturas, utilizados tanto em uma quanto na outra que facilitarão o seu aprendizado, saiba quais são:

- Linguagem **C++**: compartilha comandos, biblioteca e estrutura da linguagem **C**.
- Para o desenvolvimento de softwares empresariais, APIs, dispositivos móveis, plataformas de Website, escolha a linguagem **C#**.
- <https://dotnetfiddle.net/>

Se você quer aprender a desenvolver jogos, interfaces gráficas, e sistemas que trabalham mais perto da máquina, a linguagem **C++** é mais indicada do que a linguagem **C**.

Java 1991

Java é uma linguagem de programação amplamente usada para codificar aplicações Web.

Ela tem sido uma escolha popular entre os desenvolvedores há mais de duas décadas, com milhões de aplicações **Java** em uso hoje.

Java é uma linguagem multiplataforma, orientada a objetos e centrada em rede que pode ser usada como uma plataforma em si.

É uma linguagem de programação rápida, segura e confiável para codificar tudo, desde aplicações móveis e software empresarial até aplicações de big data e tecnologias do servidor.

```
using System;

public class Program
{
    public static void Main()
    {
        Console.WriteLine("Informe seu nome");
        string nome;
        nome = Console.ReadLine();
        Console.WriteLine("Seja bem-vindo " + nome);
    }
}
```


Linguagens de Programação

HTML

HTML é a linguagem de marcação que nós usamos para estruturar e dar significado para o nosso conteúdo web. Por exemplo, definindo parágrafos, cabeçalhos, tabelas de conteúdo, ou inserindo imagens e vídeos na página.

CSS

CSS é uma linguagem de regras de estilo que nós usamos para aplicar estilo ao nosso conteúdo HTML. Por exemplo, definindo cores de fundo e fontes, e posicionando nosso conteúdo em múltiplas colunas.

JavaScript

JavaScript é uma linguagem de programação que permite a você criar conteúdo que se atualiza dinamicamente, controlar multimídias, imagens animadas, e tudo o mais que há de interessante.

Não tudo, mas é maravilhoso o que você pode efetuar com algumas linhas de código **JavaScript**.

JavaScript é uma linguagem de programação que permite a você implementar itens complexos em **páginas web** — toda vez que uma página da web faz mais do que simplesmente mostrar a você informação estática — mostrando conteúdo que se atualiza em um intervalo de tempo, mapas interativos ou gráficos 2D/3D animados, etc. — você pode apostar que o **JavaScript** provavelmente está envolvido.

É a terceira camada do bolo das tecnologias padrões da web, duas das quais (***HTML e CSS***).

<https://codepen.io/pen>

Linguagens de Programação

PHP

PHP é uma linguagem de programação voltada para a web e que tem conquistado cada vez mais adeptos. Fácil de utilizar, robusta e com melhorias constantes, ela é uma escolha certa para quem quer trabalhar em projetos qualificados e sem complicação.