

Linguagens de Programação

Motivação para estudo de Linguagens de Programação

Samuel da Silva Feitosa

Aula 2



Razões para estudar conceitos de Linguagens de Programação

Comentários Iniciais



“Frankly, we didn’t have the vaguest idea how the thing [FORTRAN language and compiler] would work out in detail...”

John Backus

- Na época, acreditava-se que linguagens de alto nível seriam ineficientes.
- Sucesso inesperado é muito comum.
 - O navegador de internet é um exemplo.

Razões para estudar Linguagens de Programação

- Aumentar a habilidade para expressar ideias
- Melhorar seu conhecimento para escolher linguagens apropriadas
- Aumentar a habilidade para aprender novas
- Melhor entendimento do significado das implementações
- Avanço geral da computação



Domínios de Programação

- Computação Científica
 - Fortran, Matlab, R, etc.
- Aplicações para Negócios
 - Cobol, C#, Java, Delphi, SQL, etc.
- Inteligência Artificial
 - Prolog, Lisp, Python, R, Matlab, etc.
- Programação de Sistemas
 - Assembly, C, e outras.



Critérios de Avaliação de LPs

- **Legibilidade**
 - Facilidade na qual programas podem ser lidos e entendidos
- **Capacidade de Escrita**
 - Facilidade na qual a linguagem pode ser usada para criação de programas
- **Confiabilidade**
 - Conformidade com a sua especificação.
- **Custo**
 - Custo total geral

Legibilidade



- Simplicidade geral
 - Conjunto gerenciável de features e construtores.
- Ortogonalidade
 - Poucos construtores que podem ser combinados.
- Estruturas de controle e tipos de dados
 - Presença de estruturas de controle bem definidas (If, While) e facilidades para definir estruturas de dados.
- Considerações sobre a sintaxe
 - Construtores e palavras-chave de fácil compreensão.

Confiabilidade



- Checagem de tipos
 - Verificação de erros de tipos.
- Manipulação de exceções
 - Interceptação de erros de tempo de execução e tomada de medidas corretivas.
- Legibilidade e capacidade de escrita
 - Uma linguagem que não suporta formas “naturais” de expressar um algoritmo, irá usar abordagens “não naturais”, consequentemente reduzindo a confiabilidade.

Custos

- Treinamento de programadores
- Licenciamento de ferramentas
 - Disponibilidade de compiladores gratuitos
- Compilando e executando programas
- Manutenção de programas
- Outros critérios
 - Portabilidade (código e plataforma).
 - Generalidade (utilização para diferentes aplicações).
 - Conformidade com a especificação oficial.





Gerações de Linguagens de Programação

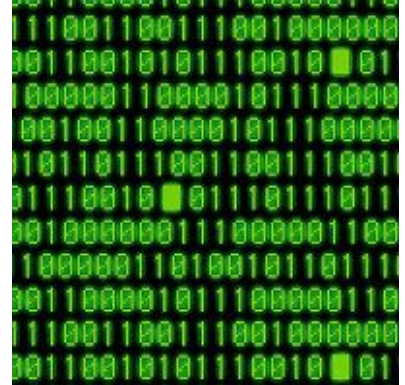
Gerações de Linguagens de Programação

- **1ª Geração:** Linguagem de máquina
- **2ª Geração:** Linguagens de montagem (assembly)
- **3ª Geração:** Linguagens procedurais
- **4ª Geração:** Linguagens orientadas a aplicação
- **5ª Geração:** Linguagens do conhecimento



Primeira Geração

- Todo computador possui um conjunto de instruções que seu processador é capaz de executar.
- Essas instruções, chamadas de código de máquina, são representadas por sequências de bits.
- Esse código é chamado de **código binário**.
 - São formados apenas por 0s e 1s.



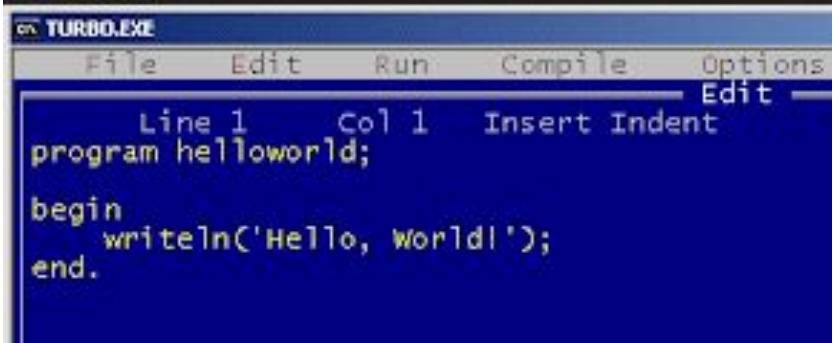
Segunda Geração

- Nível imediatamente acima da linguagem de máquina.
- Mesma estrutura e conjunto de instruções que a linguagem de máquina, porém permite que o programador utilize nomes e símbolos em lugar dos números.

```
Ltmp1993:
    .cfi_offset %rbp, -16
    movq %rsp, %rbp
Ltmp1994:
    .cfi_def_cfa_register %rbp
    addl $42, %edi
    movl %edi, %eax
    popq %rbp
    retq
```


Terceira Geração

- Elemento básico de programação é a procedure (sub-rotinas ou funções).
- Algumas das linguagens de alto nível mais utilizadas (C, Pascal, BASIC, FORTRAN, COBOL, Ada) são todas linguagens procedurais.

A screenshot of the Turbo Pascal IDE window titled 'TURBO.EXE'. The menu bar includes 'File', 'Edit', 'Run', 'Compile', and 'Options'. Below the menu bar, there is a status bar showing 'Line 1 Col 1 Insert Indent'. The main text area contains the following Pascal code:

```
program helloworld;  
  
begin  
    writeln('Hello, World!');  
end.
```

Quarta Geração

- Geralmente são linguagens com propósitos específicos (domain-specific languages).
- Um exemplo é a linguagem SQL, que se tornou um padrão para manipulação e consulta de bancos de dados.



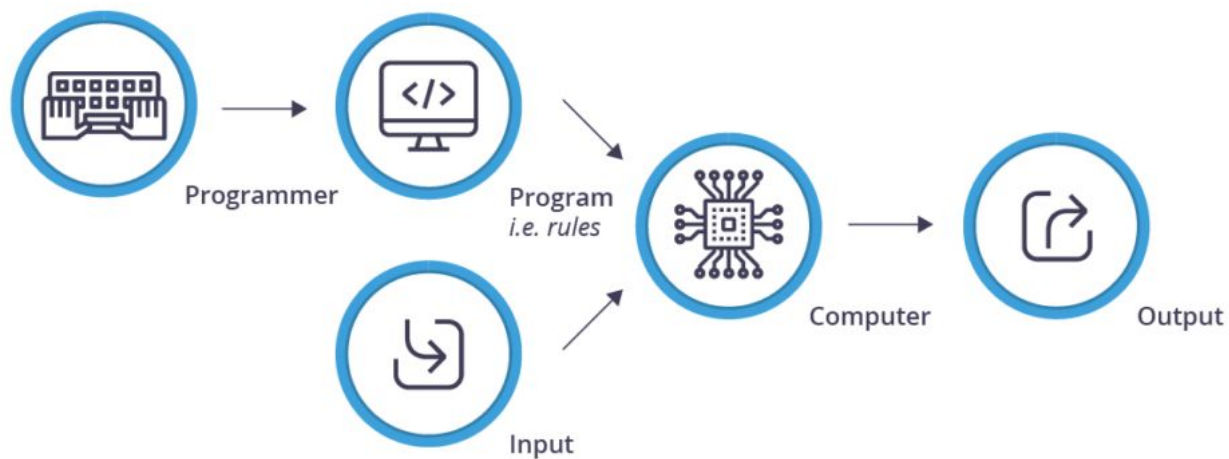
**DOMAIN
SPECIFIC
LANGUAGES**

Quinta Geração

- Geralmente associadas a inteligência artificial (IA).
- PROLOG é um exemplo de linguagem, muito utilizada na área de IA, que permite fazer programação utilizando conceitos de lógica.
- LISP é um exemplo de linguagem funcional, também bastante utilizada anos atrás que pode ser considerada dessa geração.

1ª a 5ª Geração

Traditional Computing

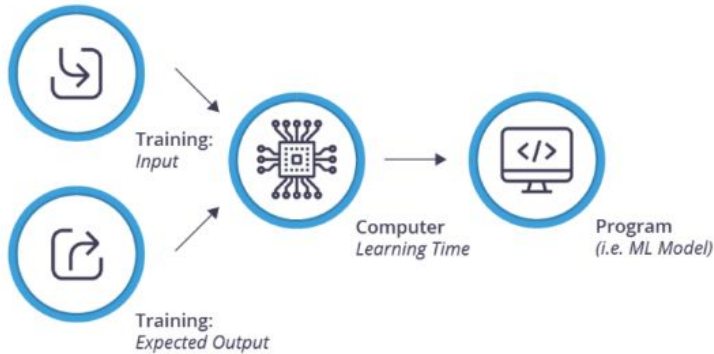


Existe a Sexta Geração?

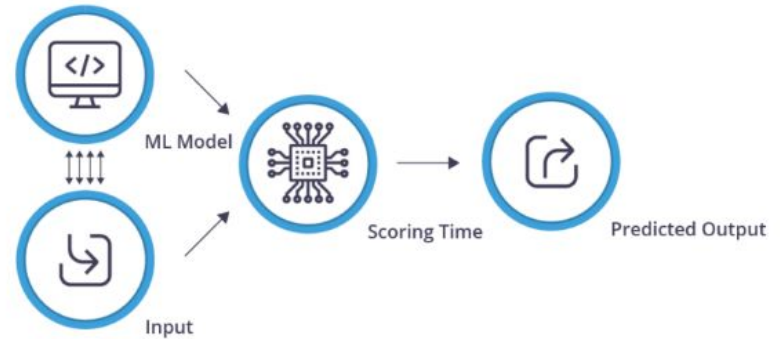
- Algumas bibliografias trazem a sexta geração de linguagens de programação.
- Considera-se a utilização de redes neurais para aferir resultados a partir de dados fornecidos pelo usuário.
- Também existem projetos que utilizam aprendizagem de máquina para a construção de programas automaticamente.

Sexta Geração

The Machine Learning **Training** Process



The Machine Learning **Scoring** Process



Comentários Finais

- Na computação é muito comum o sucesso inesperado.
- Diversas razões para se estudar conceitos de linguagens de programação.
- As linguagens tem evoluído com o passar do tempo.
- Novas linguagens e popularidade varia com o passar do tempo.