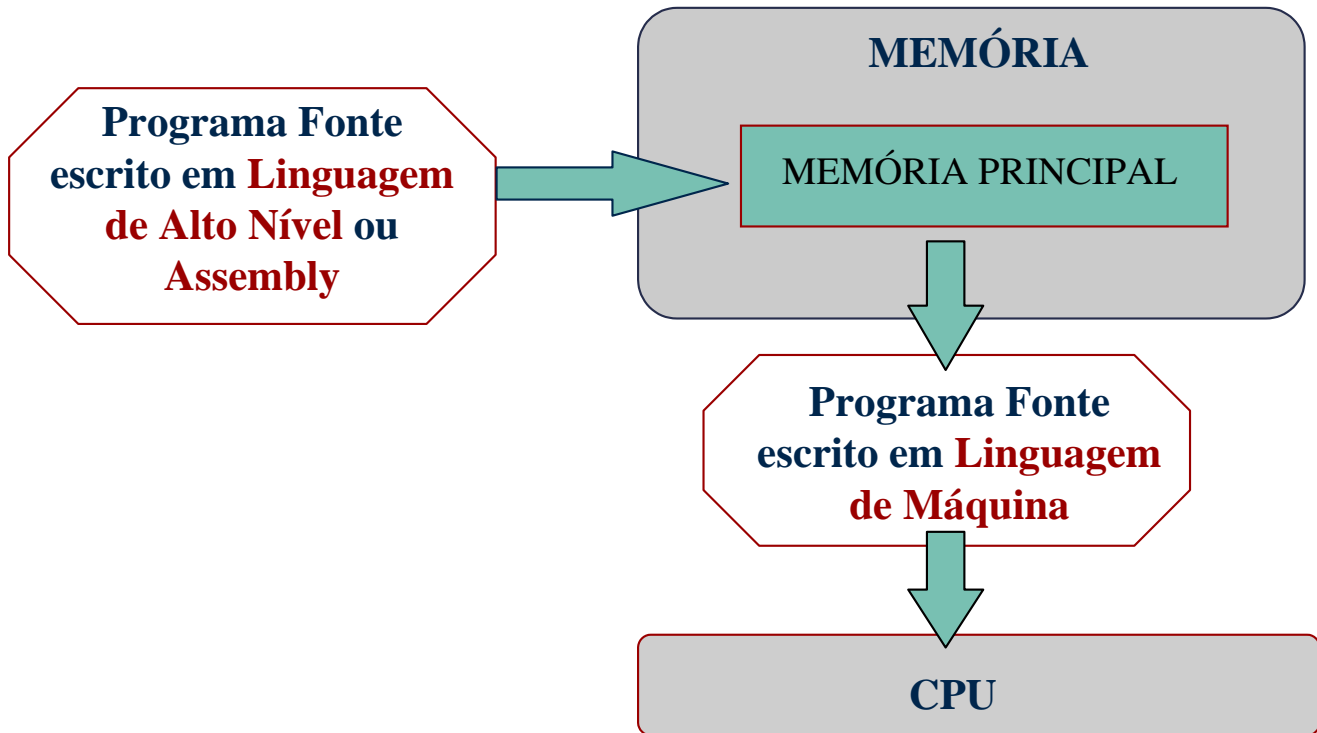


# Linguagens de Programação



## Linguagens de Programação

- Cada linguagem de programação foi projetada para resolver um determinado tipo de problema.
- É importante conhecer os pontos fortes e as limitações de cada uma dessas linguagens e de outras ferramentas de programação para poder escolher a linguagem mais apropriada para escrever o software.

# Principais Linguagens de Programação

- Linguagem Assembly
- FORTRAN
- COBOL
- PL/1
- BASIC
- Pascal
- Ada
- C

## Linguagem Assembly

- Desenvolvida para superar algumas das dificuldades da linguagem de máquina.
- São utilizados códigos e símbolos mnemônicos fáceis de recordar para representar operações e localizações de armazenamento.  
Ex: A significa "Add" (Somar) e L representa "Load" (Carregar).
- Linguagem é muito orientada para a máquina - suas instruções são para um determinado computador e um microprocessador específico.
- A execução de programas em assembly é extremamente rápida e eficiente.
- Mais fácil de usar do que a linguagem de máquina pura, mas ainda extremamente difícil de aprender.
- Utilizada principalmente para a escrita de software de sistemas operacionais.

# Fortran

- FORTTRAN : *FORmula TRANslator*
- desenvolvido em 1954 para facilitar a escrita de software científico, matemático e de engenharia que empregam fórmulas complexas.
- ponto forte: facilidade para os cálculos matemáticos.
- não dispõe de grandes facilidades para atividades de entrada/saída ou para o trabalho com listas.
- não é apropriado para problemas comerciais que envolvem a leitura de grandes volumes de números ou registros e produzem relatórios.
- utilizado com sucesso em problemas comerciais que exigem cálculos sofisticados, como previsões e modelagem.

# COBOL

- COBOL : *COmmon Business-Oriented Language*
- introduzido no início dos anos 60 sendo ainda a linguagem predominante para problemas comerciais.
- projetado para processar grandes arquivos de dados com caracteres alfanuméricos (dados alfabéticos e numéricos misturados), que são característicos dos problemas comerciais.
- pode ler, gravar e manipular registros com muita eficiência.
- considerado mais fácil de aprender que a maioria das outras linguagens de programação
  - utiliza instruções semelhantes ao inglês, é bastante fácil de ler e suporta programas bem-estruturados.
- não manipula bem cálculos matemáticos complexos, e seus programas tendem a ser prolixos e longos.

# Basic

- BASIC : *Beginner's All-Purpose Symbolic Instruction Code*)
- desenvolvido em 1964 para ensinar os alunos do Dartmouth College a usar computadores.
- linguagem de programação extremamente popular para PCs e para ensinar programação nas faculdades e no curso secundário.
- fácil de aprender e tem requisitos de memória mínimos para conversão em código de máquina.
- pode lidar com vários tipos de problemas, embora especialistas afirmem que ele executa poucas tarefas muito bem.
- não tem estruturas fortes para impor um fluxo claro de programas lógicos e bem-organizados, não sendo assim útil para o ensino de práticas de programação estruturada.

# Pascal

- O Pascal deve seu nome a *Blaise Pascal*, o matemático e filósofo do século XVII.
- desenvolvido pelo professor de ciência da computação Niklaus Wirth, de Zurique, no final dos anos 60.
- Wirth queria criar uma linguagem que ensinasse aos alunos as técnicas de programação estruturada.
- os programas consistem em pequenos subprogramas, cada um dos quais já é um programa estruturado.
- a linguagem é usada principalmente nos cursos de ciência da computação para o ensino de práticas eficientes de programação.
- tem recursos limitados de entrada e saída, não sendo por isso muito adequado para a maioria dos problemas comerciais.

# C

- desenvolvido sob o patrocínio da *Bell Laboratories* da *AT&T* no início dos anos 70
- a maior parte do sistema operacional UNIX é escrito em C
- tem muito do estrito controle e da eficiência de execução da linguagem assembly, embora seja mais fácil de aprender e possa ser utilizada por diferentes microprocessadores.
- grande parte do software comercial foi escrita em C, e esta linguagem está começando a ser utilizada para aplicações comerciais. científicas e técnicas em grandes computadores.

# Linguagens de Quarta Geração

- oferecem duas grandes vantagens:
  - possibilitam que os usuários finais desenvolvam software por si mesmos com pouca ou nenhuma ajuda
  - proporcionam substanciais ganhos de produtividade no desenvolvimento de software.
- tendem a ser menos procedurais do que as linguagens convencionais, o que as torna mais adequadas para os usuários finais.
- criaram a plataforma técnica para que não-especialistas desempenhem um papel mais importante na resolução de problemas com sistemas de informação.
- estudos têm demonstrado que as linguagens podem produzir ganhos de 300% a 500% sobre as linguagens convencionais.
- Estão disponíveis alguns tipos de ferramentas de software de quarta geração:
  - Linguagens de consulta
  - Linguagens gráficas
  - Geradores de relatórios
  - Geradores de aplicações
  - Linguagens de programação de altíssimo nível



# Programação Orientada a Objetos

- *Como seria se os carros tivessem motores não-padronizados e construídos a mão?*

Os automóveis não poderiam ter um papel importante na vida diária - seriam muito caros para usar e difíceis de consertar.

- Entretanto, essa é a situação que tem prevalecido para o software.

A maior parte do software ainda é construída peça por peça por artesãos altamente treinados, algo como ferreiros medievais ou entalhadores de madeira.

- A tecnologia orientada a objetos promete criar uma "revolução industrial" do software pela possibilidade de se utilizarem partes intercambiáveis de modo semelhante ao que ocorre na manufatura moderna.
- Essas partes intercambiáveis de software são chamadas de objetos .
- Cada objeto é uma coleção encapsulada de dados e operações executadas sobre esses dados.
- O principal benefício da programação orientada a objetos é a capacidade de reutilização do código.
  - Os diversos objetos podem ser reutilizados em diferentes combinações.

# Programação Visual

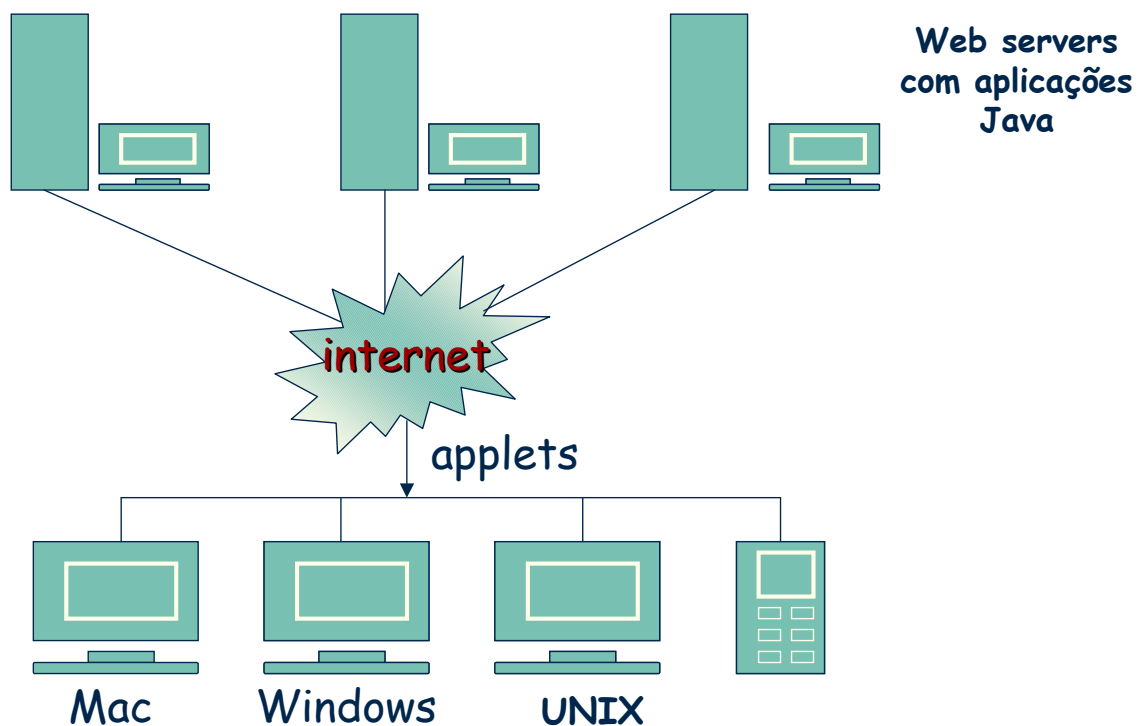
- A programação orientada a objetos foi a base para uma técnica conhecida como programação visual
- A programação visual permite que um programador crie um programa trabalhando com um grupo de objetos, clicando em um determinado objeto e movimentando ou copiando este objeto para outro lugar, ou desenhando uma linha para ligar um objeto a outro
- A programação visual pode melhorar a produtividade dos programadores e dos usuários finais.
- O Visual BASIC é uma ferramenta de programação visual amplamente usada para elaborar muitas aplicações de software.

# Linguagem JAVA

- Java - desenvolvida pela Sun Microsystems para o processamento de dados baseado em redes
- representa um conceito revolucionário no desenvolvimento de software.
  - permite que os desenvolvedores criem aplicativos que podem ser distribuídos pela Internet.
  - é uma plataforma independente. O software pode funcionar em qualquer computador ou dispositivo computacional, qualquer que seja o microprocessador ou o sistema operacional utilizado.
- Os softwares aplicativos tradicionais costumam executar muitas funções no mesmo pacote de software.
  - Ex: um programa de edição de texto inclui não apenas funções usadas costumeiramente como a eliminação de texto, ajuste de margens e modificação do espaçamento de linhas, mas também muitos outros recursos visando a propósitos mais especializados — a fusão de documentos, a inserção de notas de rodapé, a numeração automática, a inserção de gráficos e assim por diante.
  - A maioria das pessoas utiliza apenas um pequeno número dessas funções de cada vez.
- A Java, ao contrário, permite que os usuários criem miniaturas de aplicações chamadas "applets" (miniaplicativos) que residem em uma rede em servidores centralizados.

- A rede oferece apenas os applets para as funções específicas de que você necessita.
- Com os applets Java residindo em sua rede, não é necessário armazenar os dados nem o programa no disco rígido, de modo que é possível usar uma máquina muito menos poderosa para o trabalho.

## Java - A Interface Universal



Um browser que utiliza Java atua como uma interface universal que possibilita que os usuários acessem aplicações escritas na linguagem Java, não importando o tipo de computador ou o sistema operacional dos usuários.

# SELECIONANDO UMA FERRAMENTA DE PROGRAMAÇÃO

- Considerações importantes :

1. A natureza do problema a ser resolvido

- Trata-se de um problema científico que exige modelagem matemática ou um problema comercial que engloba uma grande manipulação de arquivos e operações de entrada / saída ?

2. Requisitos de hardware

- A linguagem ou ferramenta é compatível com os recursos de hardware de seu computador?
- É essencial que ela seja capaz de funcionar em mais de um tipo de máquina?
- Ela funcionará com o sistema operacional do seu computador?
- Existem limitações quanto a tamanho de memória ou a recursos do computador?

3. Facilidade de uso

- Trata-se de uma linguagem ou ferramenta que você pode utilizar, que seu pessoal técnico já está familiarizado, que pode ser aprendida facilmente?

4. Sustentabilidade

- É importante que a linguagem seja altamente estruturada?
- A linguagem ou ferramenta suporta programas que podem ser modificados e mantidos por outros durante um longo período de tempo?