

# Linguagem de Programação: Grupo 1 - Pascal

## Trabalho Teórico Prático (TTP) - Disciplina: Linguagens de Programação

Edgard de Paiva Melo Filho, Maicon Gomes Mesias



Pontifícia Universidade Católica de Minas Gerais  
Instituto de Ciências Exatas e Informática  
Curso de Engenharia de Computação

16 de Setembro de 2025



# Sumário / Roteiro / Agenda

- 1 Introdução
- 2 Linha do Tempo Genealogia da Linguagem
- 3 Paradigma(s) a que Pertence (Características Principais)
- 4 Características mais Marcantes da Linguagem
- 5 Linguagens Relacionadas (Influenciadores, Influenciadas, Similares, “Opostas”, etc.)
- 6 Exemplo(s) de Programa(s)
- 7 PRÁTICA: Tutoriais de Instalação, Uso e Programação + Exemplos
- 8 Considerações Finais
- 9 Bibliografia
- 10 Apêndice



# Introdução

- A linguagem de programação Pascal surgiu em 1970, criada por Niklaus Wirth na Universidade Federal de Zurique (ETH Zurich), com o objetivo de oferecer uma linguagem estruturada, clara e eficiente para o ensino da programação e da ciência da computação.
- Importância: Adotada amplamente em universidades e escolas técnicas durante as décadas de 1970, 1980 e início de 1990; introduziu conceitos fundamentais de programação estruturada que influenciaram linguagens modernas como Python, Java e C++ direta ou indiretamente.



# Linha do Tempo

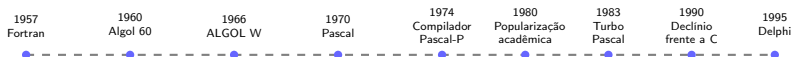


Figure: Linha do tempo da evolução da linguagem Pascal



# Linha do Tempo

## LINHA DO TEMPO

**1957 – Fortran**  
Primeira linguagem de alto nível amplamente utilizada.

**1966 – ALGOL W**  
Extensão do Algol, precursor direto de Pascal

**1974 – Compilador Pascal**  
Popularizou a linguagem em ambiente acadêmico.

**1983 – Turbo Pascal**  
Lançado pela Borland, tornou Pascal prático para softwares comerciais.

**1990 – Declínio frente ao C**  
Linguagens mais poderosas dominaram o mercado.



1957  
Fortran

1960  
Algol 60

1966  
ALGOL W

1970  
Pascal

1974  
Compilador  
Pascal-P

1980  
Popularização  
acadêmica

1983  
Turbo  
Pascal

1990  
Declínio  
frente a C

1995  
Delphi

**1960 – Algol 60 →**  
Base conceitual para linguagens estruturadas.

**1970 – Pascal →**  
Criação por Niklaus Wirth, com foco educacional.

**1980 – Popularização acadêmica →** Forte adoção em universidades e escolas técnicas.

**1995 – Delphi →** Evolução orientada a objetos com suporte a interfaces gráficas.

de Paiva Melo Filho, Maicon Gomes Mesias



# Genealogia da Linguagem

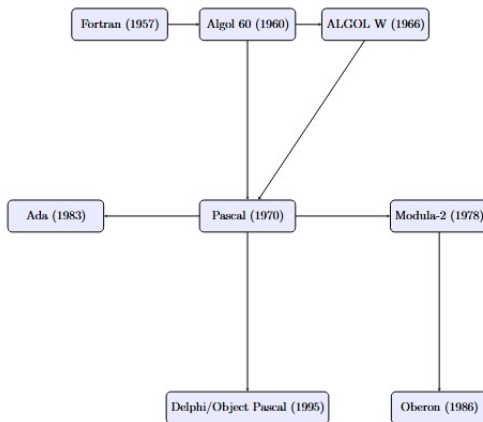


Figura 2 – Genealogia da linguagem Pascal e suas influências



# Paradigmas e Características Principais

- **Paradigma Imperativo:** O programa é descrito como uma sequência de comandos que alteram o estado do computador, tendo instruções claras de 'o que fazer' e 'como fazer'.
- **Paradigma Procedural:** Código organizado em procedimentos e funções para modularização e reutilização (ex.: `procedure ExibirMensagem; begin writeln('Olá, mundo!'); end;`).
- **Paradigma Estruturado:** Ênfase em blocos (`begin...end`), estruturas de controle (`IF...THEN...ELSE`, `FOR`, `WHILE`) e fluxo lógico claro, evitando `goto` excessivo.
- Ênfase em legibilidade, ensino e organização lógica do código.



## Características Marcantes

- Tipagem forte e estática: Variáveis devem ter tipo definido; previne erros em tempo de compilação.
- Tipos de dados: Integer, Real, Char, String, Boolean, Array, Record, Set, File.
- Palavras reservadas: and, begin, case, const, do, else, end, if, of, procedure, program, repeat, then, until, var, while, etc.
- Operadores: Aritméticos (+, -, \*, /, div, mod); Relacionais (=, <>, <, >, <=, >=).
- Estruturas de controle claras (if, while, for); suporte a procedimentos/funções aninhadas.





## Características Marcantes (Cont. - Resumo Visual)

Característica	Descrição
Fortemente Tipada	Garante tipos definidos, prevenindo erros entre tipos incompatíveis.
Estruturada	Blocos begin...end e fluxo lógico claro, evitando goto.
Procedural	Organização em procedimentos/funções para modularidade.
Legibilidade	Sintaxe próxima à linguagem natural (if, for, while).
Segurança	Tipagem forte detecta erros em compilação.
Suporte a Dados	Arrays, records, sets, files para organização eficiente.
Facilidade de Ensino	Didática para algoritmos e lógica.
Portabilidade	Compiladores para múltiplas plataformas.



# Características Marcantes - com Exemplos

## Tipagem Forte

- Toda variável precisa ter o tipo definido.
- Evita erros de atribuição indevida.

```
var
  idade: integer;
begin
  idade := 20; { Correto }
  { idade := 'vinte'; → Erro, pois string não pode ser atribuída a integer }
end.
```

## Sintaxe Simples e Legível

- Código escrito de forma clara, próximo da lógica humana.

```
begin
  writeln('Ola, mundo!'); { Imprime na tela }
end.
```

## Modularização (Procedimentos e Funções)

- Permite dividir o programa em partes menores.

```
procedure MostrarMensagem;
begin
  writeln('Bem-vindo ao Pascal!');
end;

begin
  MostrarMensagem;
end.
```

## Paradigma Estruturado

- Uso de estruturas de controle bem definidas.

```
var
  i: integer;
begin
  for i := 1 to 5 do
    writeln('Contagem: ', i);
  end.
```



# Linguagens Relacionadas

- **Influenciadores:** ALGOL 60 (sintaxe e estruturas), Fortran (arrays), PL/I (modularidade).
- **Influenciadas:** Modula-2 (1978, evolução por Wirth), Oberon (1986), Ada (1983, tipagem forte), Delphi/Object Pascal (1995).
- **Similares:** C/C++ (estruturada e procedural, mas C menos rígido em tipagem).
- **“Opostas”:** Linguagens dinâmicas como Python ou JavaScript (tipagem dinâmica, menos estrutura rígida).



# Linguagens Relacionadas - Visual

## LINGUAGENS RELACIONADAS

### Influências recebidas:

- **Fortran (1957)**: pioneiro em programação científica.
- **Algol-60 (1960)**: trouxe estruturas de controle modernas.
- **ALGOL W (1966)**: serviu de protótipo direto para Pascal.

### Linguagens que foram Influenciadas:

- **Modula-2 (1978)**: foco em modularização.
- **Influência em Ada (1980s)**: herdou princípios estruturados e de tipagem forte.
- **Oberon (1986)**: simplificação e foco em eficiência.
- **Delphi/Object Pascal (1995)**: orientação a objetos + interfaces gráficas.



# Exemplo 1: Cálculo da Média

```
program CalcMedia;  
var  
    num1, num2, num3, media: real;  
begin  
    writeln('Digite tr s n meros:');  
    readln(num1, num2, num3);  
    media := (num1 + num2 + num3) / 3;  
    writeln('A m dia : ', media:0:2);  
end.
```

**Descrição:** Cálculo da média de três números inseridos pelo usuário.



## Exemplo 2: FizzBuzz

```
program FizzBuzz;  
var  
    i: integer;  
begin  
    for i := 1 to 20 do  
        begin  
            if (i mod 3 = 0) and (i mod 5 = 0) then  
                writeln('FizzBuzz')  
            else if i mod 3 = 0 then  
                writeln('Fizz')  
            else if i mod 5 = 0 then  
                writeln('Buzz')  
            else  
                writeln(i);  
        end;  
    end.
```



# Prática: Tutoriais e Exemplos

- 1 Instalação:** Baixe o Free Pascal Compiler (FPC) de <https://www.freepascal.org/>. Para Windows/Linux/macOS, siga o instalador; Lazarus IDE recomendada para desenvolvimento gráfico.
- 2 Uso:** Crie arquivo .pas, compile com `fpc programa.pas`, execute o binário (`./programa` ou `programa.exe`).
- 3 Exemplo prático:** Verificar se número é par ou ímpar.
- 4 Opção Alternativa:** Utilize uma IDE online como por exemplo [https://www.onlinegdb.com/online\\_pascal\\_compiler](https://www.onlinegdb.com/online_pascal_compiler)



# Exemplo Prático: Par ou Ímpar

```
program ParImpar;  
  var numero: integer;  
begin  
  writeln('Digite um numero:');  
  readln(numero);  
  if numero mod 2 = 0 then  
    writeln('Numero par')  
  else  
    writeln('Numero mpar ');  
  readln;  
end.
```

Compile e execute para testar.





# Considerações Finais

- Legado de Pascal: Importância histórica e pedagógica; moldou gerações de programadores com conceitos de programação estruturada e tipagem forte.
- Vantagens: Clareza, segurança, facilidade de ensino; influenciou linguagens modernas.
- Limitações atuais: Menos usada em desenvolvimento moderno devido a linguagens mais versáteis, mas persiste em educação e legados.
- Perspectivas futuras: Manutenção via Free Pascal; relevância em ensino de fundamentos.



# Bibliografia

- EVARISTO, Jaime. *Programando com Pascal: a linguagem do Turbo Pascal e do Delphi*. 2. ed. revista. São Paulo: E-Book Express, 2000.
- JENSEN, Kathleen; WIRTH, Niklaus. *Pascal user manual and report*. 4. ed. Revisão de Andrew B. Mickel; James F. Miner. Berlin: Springer-Verlag, 1974.
- Documentação Free Pascal:  
<https://www.freepascal.org/docs.html>.



## Apêndice: Manifestação Individual

**Edgard de Paiva Melo Filho:** Foi com considerável surpresa que durante a pesquisa foi visto que as primeiras versões de programas relativamente recentes em comparação com a idade da linguagem de programação, foram feitos utilizando variações de Pascal, por exemplo o Skype, onde o cliente desktop inicial foi rapidamente prototipado usando Object Pascal no ambiente de desenvolvimento Delphi.

**Maicon Gomes Mesias:** Sinceramente, eu vejo o Pascal como uma linguagem que marcou época. Ela pode parecer "antiga" hoje em dia, mas teve um papel muito importante: ajudou muita gente a aprender lógica de programação de forma organizada. O legal do Pascal é que ele é bem estruturado, então não dá pra sair escrevendo código bagunçado; ele praticamente "obriga" a programar com disciplina.

