

# Educação Profissional Paulista

Técnico em  
**Ciência de  
Dados**

# Lógica de programação e algoritmos

## Introdução à lógica de programação e algoritmos

Aula 2

[DADOS]ANO1C3B2S9A2

# Exposição



## Objetivo da aula

Apresentar mais informações sobre portas lógicas e ensinar expressões booleanas.



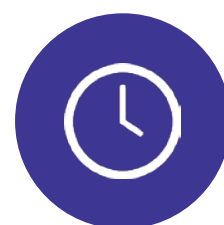
## Competências da Unidade (técnicas e socioemocionais)

- Usar técnicas para explorar e analisar dados, aplicar modelos estatísticos, identificar padrões, realizar inferências e tomar decisões baseadas em evidências;
- Compreender e dominar técnicas de manipulação de dados;
- Extrair, transformar e carregar conjuntos de dados de diferentes fontes, garantindo a qualidade e a integridade dos dados;
- Criar e compreender visualizações gráficas.



## Recursos didáticos

- Recurso audiovisual para exibição de vídeos e imagens;
- Acesso ao laboratório de informática e/ou à internet.



## Duração da aula

50 minutos.

# Estruturas condicionais – SE/ENTÃO/SENÃO

## O que é?

A estrutura condicional **SE** e **SENÃO** (IF e ELSE) é uma construção-chave na programação, permitindo que um programa tome decisões com base em condições específicas.

Ela avalia se uma expressão é verdadeira e executa um bloco de código associado quando essa condição é atendida; caso contrário, executa outro bloco de código.



## Tome nota

Além do SE e do SENÃO, podemos usar **SE SENÃO** para adicionar condições extras, tornando apenas o "SENÃO" uma estrutura independente.

## Estruturas condicionais – SE/ENTÃO/SENÃO

### Como utilizar?

**SE** (condição) **ENTÃO**

# Bloco de código a ser executado se a condição for verdadeira

**SENÃO SE** (condição 2) **ENTÃO**

# Bloco de código a ser executado se a condição 2 for verdadeira e a condição 1, falsa

**SENÃO**

# Bloco de código a ser executado se todas as condições forem falsas

**FIM\_SE**

### Evite:

- aninhar excessivamente SEs e ELSEs, pois isso pode tornar o código difícil de entender;
- não fornecer um bloco de código para o SENÃO, tornando o comportamento inesperado;
- utilizar lógicas de decisão como SE e SENÃO, em vez de lógicas de repetição, equivocadamente.



# Estruturas condicionais – SE/ENTÃO/SENÃO

### Observações finais:

- **Clareza:** torne as condições e os blocos de código claros e fáceis de entender.
- **Indentação:** use uma indentação consistente para melhor legibilidade.
- **Blocos de código:** garanta que as ações no bloco SE e SENÃO façam sentido para a lógica do programa.

A estrutura condicional SE e SENÃO é poderosa para controlar o fluxo de um programa, permitindo que ele tome decisões dinâmicas com base duas condições específicas.

# Estruturas condicionais – SE/ENTÃO/SENÃO

Vamos repetir os enunciados e evoluir a solução com o SENÃO?

Você foi designado para criar um sistema de **votação** muito simples em pseudocódigo e fluxograma. O sistema deve permitir que uma pessoa vote apenas se tiver **18 anos** ou **mais**. Crie um pseudocódigo para esta situação.

### Requisitos:

- Solicite à pessoa que informe a idade dela.
- Verifique se a idade é maior ou igual a 18.
- Se a idade atender aos requisitos, permita que a pessoa vote.
- Caso contrário, informe que a pessoa não está autorizada a votar devido à idade.

## Estruturas condicionais – SE/ENTÃO/SENÃO

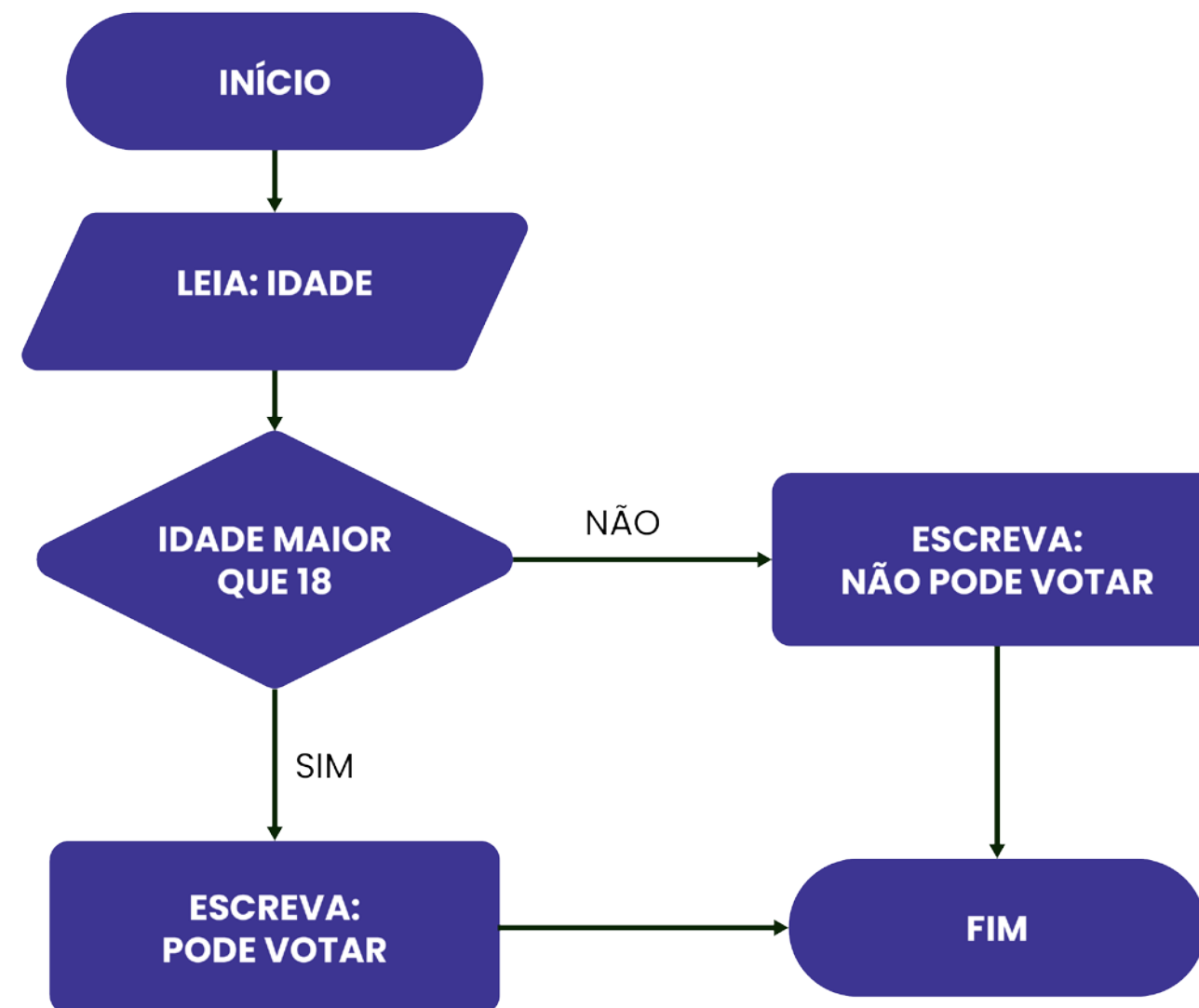
Solução:

1. Início
2. LEIA idade
3. SE idade  $\geq$  18 ENTÃO
4.     ESCREVA "Você está autorizado a votar."
5.     SENAO
6.     ESCREVA "Você não está autorizado a votar devido à idade."
7.     FIM\_SE
8. Fim



## Estruturas condicionais – SE/ENTÃO/SENÃO

Exemplo elementar em fluxograma:



# Estruturas condicionais – SE/ENTÃO/SENÃO

Desenvolva um pseudocódigo para verificar se um número fornecido pelo usuário é **par** ou **ímpar**. Siga as instruções abaixo:

### Requisitos:

- Solicite ao usuário que insira um número inteiro.
- Use uma estrutura de decisão IF para determinar se o número é par ou ímpar.
- Se o número for par, exiba uma mensagem informando que é par.
- Caso contrário, ou seja, se o número for ímpar, exiba uma mensagem informando que é ímpar.

## Estruturas condicionais - SE/ENTÃO/SENÃO

Solução:

1. Início
2. LEIA número
3. SE número dividido por dois resta zero ENTÃO
4.     ESCREVA "O número é par."
5.     SENAO
6.     ESCREVA "O número é ímpar."
7.     FIM\_SE
8. Fim

# Estruturas condicionais – ESCOLHA/CASO

A estrutura condicional **ESCOLHA CASO** (ou SWITCH CASE, em inglês) é uma construção utilizada em programação para realizar **seleções múltiplas** com base no **valor de uma expressão**.

Ela oferece uma alternativa mais eficiente e legível em comparação com encadeamentos de estruturas condicionais SE-SENÃO.

**Avaliação da expressão:** uma expressão é avaliada e seu valor é comparado com diferentes casos possíveis.

**Seleção do bloco correspondente:** o bloco de código associado ao caso que corresponde ao valor da expressão é executado.

**Break:** após a execução do bloco correspondente, o controle é transferido para fora da estrutura ESCOLHA CASO com o *break*. Caso seja omitido, o programa executa a próxima condição dentro da ESCOLHA CASO, ou seja, o próximo CASO.

## Estruturas condicionais – ESCOLHA/CASO

Como utilizar?

**ESCOLHA** valor de referência:

**CASO** valor1:

# Bloco de código para valor1

**BREAK**

**CASO** valor2:

# Bloco de código para valor2

**BREAK**

...

**PADRAO:**

# Bloco de código padrão (opcional)

**FIM\_ESCOLHA**



# Estruturas condicionais – ESCOLHA/CASO

### Dicas:

- É interessante fornecer a instrução "BREAK" ao final de cada "CASO". Isso porque existem ocorrências em que a estrutura executa os blocos subsequentes.
- É interessante incluir o bloco "PADRÃO", pois assim se mantém um controle maior de erros das lógicas aplicadas nos blocos de "CASO".

### Bloco PADRÃO:

O bloco PADRÃO funciona como o SENÃO, ou seja, é outra condição que pode ser usada como medida de segurança do programa caso algo não funcione corretamente e as opções não sejam atendidas.

# Estruturas condicionais – ESCOLHA/CASO

### Observações finais:

- **Clareza:** torna o código mais legível em comparação com uma série de estruturas SE-SENÃO.
- **Eficiência:** o ESCOLHA CASO é especialmente útil quando há várias opções a serem comparadas.
- **O uso do PADRÃO:** o bloco PADRÃO é opcional e é executado quando nenhum caso corresponde à expressão.
- **Flexibilidade de casos:** permite o tratamento de múltiplas condições de forma mais clara e estruturada que múltiplos IF-ELSE.

# Estruturas condicionais – ESCOLHA/CASO

Desenvolva um pseudocódigo para verificar se um número fornecido pelo usuário é positivo ou negativo. O programa deve seguir os seguintes requisitos:

### REQUISITOS:

- Solicite ao usuário que insira um número.
- Utilize uma estrutura de decisão IF para verificar se o número é positivo ou negativo.
- Se o número for positivo, exiba uma mensagem informando que é positivo.
- Caso contrário, ou seja, se o número for negativo, exiba uma mensagem informando que é negativo.
- Considere a situação em que o número pode ser zero e inclua uma mensagem para essa condição.

## Estruturas condicionais

### Solução com SE e SENÃO:

1. Início
2. LEIA número
3. SE número  $> 0$  ENTAO
4.     ESCREVA "O número é positivo."
5. SENAO SE número  $< 0$  ENTAO
6.     ESCREVA "O número é negativo."
7. SENAO
8.     ESCREVA "O número é zero."
9. FIM\_SE
10. Fim

### Solução com ESCOLHA CASO:

1. Início
2. LEIA número
3. ESCOLHA (Verdadeiro)
4.     CASO (número  $> 0$ ):
5.         ESCREVA "O número é positivo."
6.         BREAK
7.     CASO (número  $< 0$ ):
8.         ESCREVA "O número é negativo."
9.         BREAK
10.    PADRAO:
11.         ESCREVA "O número é zero."
12.         BREAK
13. FIM\_ESCOLHA
14. Fim



Vamos  
fazer uma  
**atividade**

## Planejando atividades de lazer

Um indivíduo está planejando suas atividades de lazer e deseja escolher a partir de seus interesses. Crie um pseudocódigo para ajudá-lo a tomar essa decisão de forma simples. Confira, ao lado, os requisitos.

Em seguida envie sua atividade pelo AVA.

 **20 minutos**

 **Individual**

- 1** O indivíduo deve informar seu tipo de preferência: "**Esportes**" ou "**Artes**".
- 2** Se a preferência for "**Esportes**", o sistema deve sugerir "**Praticar esportes**".
- 3** Se a preferência for "**Artes**", o sistema deve sugerir "**Explorar museus**".
- 4** Antes de apresentar a sugestão, o sistema deve **verificar** se o indivíduo tem uma **assinatura ativa**.
- 5** Se a assinatura estiver **ativa**, o sistema deve **aplicar** um **desconto de 20%** na atividade sugerida.
- 6** Se a assinatura **não** estiver ativa, o sistema deve **sugerir** que o indivíduo a **adquira** para obter descontos.





© Getty Images

O que nós  
**aprendemos**  
**hoje?**

## Hoje desenvolvemos:

- 1** Conhecimento sobre as **estruturas** condicionais SE e SENÃO (IF *and* ELSE);
- 2** Aplicação do **SENÃO** para evitar comportamentos inesperados;
- 3** Aplicação da estrutura condicional ESCOLHA CASO (SWITCH CASE).



# Saiba mais

Quer apreender mais sobre o tema que estudamos hoje? Veja um curso que ensina a trabalhar estruturas condicionais com linguagem diferenciada:

ALURA. *JavaScript e HTML*: desenvolva um jogo e pratique lógica de programação. Disponível em: <https://cursos.alura.com.br/course/logica-programacao-javascript-html>. Acesso em: 20 fev. 2024.

# Referências da aula

Identidade visual: Imagens © Getty Images

FORBELLONE, A. L. V.; EBERSPÄCHER, H. F. *Lógica de programação: a construção de algoritmos e estruturas de dados*. São Paulo: Pearson; Porto Alegre: Bookman, 2022.

# Educação Profissional Paulista

Técnico em  
**Ciência de  
Dados**