

# MAC2166 - Introdução à Computação

**Prof. Dr. Helder Oliveira**



# Agenda

- **Indicadores de Passagem**
- **Objetivo:**
  - Entender como utilizar indicadores de passagem para criar lógica condicional eficiente no código.

# Indicadores de Passagem

- Termo se refere a um valor ou condição que determina quando um processo ou um fluxo de execução "passa" de um estado para outro.
- Indicadores de passagem são variáveis usadas para controlar o fluxo de execução de um programa.
- Frequentemente utilizados para armazenar o estado de uma condição ou evento.
- **Exemplo:** Um indicador de passagem pode ser um **True** ou **False**, indicando se uma condição foi ou não cumprida.

# Tipos Comuns de Indicadores

- **Booleanos**: Variáveis do tipo **True** ou **False**.
- **Contadores**: Variáveis que registram quantas vezes um evento ocorreu.
- **Flags**: Indicadores simples que alteram o fluxo conforme um critério.

# Exemplo de Indicador de Passagem (Boleano)

```
1 # Exemplo simples de indicador de passagem
2 usuario_autenticado = False
3
4 # Condicional que verifica se o usuário está autenticado
5 if usuario_autenticado:
6     print("Acesso permitido.")
7 else:
8     print("Acesso negado.")
```

# Indicadores com Variáveis Simples

Exemplo de controle de idade:

```
1 # Indicador de passagem para verificar maioridade
2 maioridade = False
3 idade = 20
4
5 # Verificando se a pessoa é maior de idade
6 if idade >= 18:
7     maioridade = True
8
9 if maioridade:
10     print("Você é maior de idade!")
11 else:
12     print("Você é menor de idade.")
```

# Usando Indicadores para Alterar Fluxo

```
1 # Indicador de passagem simples
2 permitir_entrar = False
3
4 # Verificação simples
5 if permitir_entrar:
6     print("Você pode entrar.")
7 else:
8     print("Acesso negado.")
```

# Indicadores de Passagem com Flags

```
1 # Flag para controlar execução
2 flag = False
3
4 # Lógica condicional
5 if not flag:
6     print("Primeira execução.")
7     flag = True # Muda o valor da flag
8 else:
9     print("Já executado antes.")
```

# Indicadores com o Laço while

```
1 # Indicador de passagem com loop while
2 contador = 0
3 indicador = False
4
5 # Usando o while para contar até 5
6 while contador < 5:
7     print(f"Contador: {contador}")
8     contador += 1
9
10    # Se contador chega a 3, alteramos o indicador
11    if contador == 3:
12        indicador = True
13
14    if indicador:
15        print("Contador atingiu 3! Indicador ativado.")
16    else:
17        print("Contador não atingiu 3.")
```

# Benefícios do Uso de Indicadores

- Controle claro do fluxo do programa.
- Melhora a legibilidade e manutenção.
- Permite a implementação de lógicas condicionais mais flexíveis.
- Evita redundância no código.

# Exercício

- Usar o **while** para criar uma contagem regressiva até 0, e mostrar quando o valor "passa" de 3.

# Exercício

- Usar o **while** para criar uma contagem regressiva até 0, e mostrar quando o valor "passa" de 3.

```
1 valor = int(input("Entre com Valor:"))
2 while valor >= 0:
3     if valor > 3:
4         print(f"Contagem: {valor}")
5     elif valor == 3:
6         print(f"Contagem: {valor}")
7         print("Agora, passou de 3!")
8     valor -= 1
```

# Exercício

- Crie um laço while que continue pedindo um número ao usuário até que ele digite um número maior que 10. Utilize um indicador para controlar quando o número foi aceito.

# Exercício

- Crie um laço while que continue pedindo um número ao usuário até que ele digite um número maior que 10. Utilize um indicador para controlar quando o número foi aceito.

```
1 # Indicador para controlar se o número foi aceito
2 numero_aceito = False
3
4 # Laço while para pedir um número ao usuário
5 while not numero_aceito:
6     # Solicita um número ao usuário
7     numero = int(input("Digite um número maior que 10: "))
8
9     # Verifica se o número é maior que 10
10    if numero > 10:
11        numero_aceito = True # Atualiza o indicador para True
12        print("Número aceito! Você digitou:", numero)
13    else:
14        print("Número inválido. Tente novamente.")
```

# Exercício

- Criar um laço while que faça uma contagem regressiva de 10 até 0 e, ao final, imprima a mensagem "Feliz Ano Novo!".
  - O programa deve começar com o número 10 e ir diminuindo até 0, imprimindo o número a cada iteração.
  - Quando a contagem chegar a 0, o programa imprime a mensagem "Feliz Ano Novo!".

# Exercício

- Criar um laço while que faça uma contagem regressiva de 10 até 0 e, ao final, imprima a mensagem "Feliz Ano Novo!".

```
1 contador = 10
2
3 while contador >= 0:
4     print(contador)
5     contador -= 1 # Decrementa 1 a cada iteração
6
7 print("Feliz Ano Novo!")
```

# Exercício

- Criar um laço while que peça um número ao usuário e verifique se ele é par ou ímpar.
  - O laço continuará pedindo um número ao usuário até que ele digite um número positivo.
  - Quando um número positivo for digitado, o programa deve informar se o número é par ou ímpar.

# Exercício

- Criar um laço while que peça um número ao usuário e verifique se ele é par ou ímpar.

```
1 # Indicador de continuidade
2 numero_positivo = False
3
4 while not numero_positivo:
5     numero = int(input("Digite um número positivo: "))
6
7     if numero > 0:
8         numero_positivo = True # Atualiza o indicador
9         if numero % 2 == 0:
10             print(f"O número {numero} é par.")
11         else:
12             print(f"O número {numero} é ímpar.")
13     else:
14         print("Por favor, digite um número positivo.")
```

# Exercício

- Criar um jogo simples onde o usuário deve adivinhar um número secreto entre 1 e 10.
  - O programa escolhe aleatoriamente um número entre 1 e 10.
  - O laço while deve continuar até o usuário adivinhar o número correto.
  - Quando o número for adivinhado corretamente, o programa deve imprimir "Parabéns, você acertou!".

```
1 import random
2
3 numero_secreto = random.randint(1, 10) # Gera um número aleatório entre 1 e 10
```

# Exercício

- Criar um jogo simples onde o usuário deve adivinhar um número secreto entre 1 e 10.

```
1 import random
2
3 numero_secreto = random.randint(1, 10) # Gera um número aleatório entre 1 e 10
4 numero_adivinhado = False
5
6 while not numero_adivinhado:
7     palpite = int(input("Adivinhe o número secreto entre 1 e 10: "))
8
9     if palpite == numero_secreto:
10         numero_adivinhado = True
11         print("Parabéns, você acertou!")
12     elif palpite < numero_secreto:
13         print("O número secreto é maior.")
14     else:
15         print("O número secreto é menor.")
```

# Exercício

- Criar um laço while que continue pedindo números ao usuário e calcule a soma deles. O programa deve parar quando o usuário digitar o número 0.
  - O programa deve solicitar números e somá-los até que o usuário digite 0.
  - Quando 0 for digitado, o programa deve imprimir a soma total dos números inseridos.

# Exercício

- Criar um laço while que continue pedindo números ao usuário e calcule a soma deles. O programa deve parar quando o usuário digitar o número 0.

```
1 soma = 0
2
3 while True:
4     numero = int(input("Digite um número (digite 0 para parar): "))
5
6     if numero == 0:
7         break # Interrompe o laço se o número for 0
8
9     soma += numero # Adiciona o número à soma
10
11 print("A soma dos números digitados é:", soma)
```

# Exercício

- Criar um laço while que ofereça ao usuário um menu de operações matemáticas (soma, subtração, multiplicação, divisão). O programa deve continuar pedindo operações até que o usuário escolha a opção de sair.
  - O programa apresenta um menu de opções para o usuário escolher a operação que deseja fazer.
  - O laço continua até que o usuário escolha a opção de sair.

# Exercício

- Criar um laço while que ofereça ao usuário um menu de operações matemáticas (soma, subtração, multiplicação, divisão). O programa deve continuar pedindo operações até que o usuário escolha a opção de sair.

```
1 while True:  
2     print("\nEscolha uma operação:")  
3     print("1. Soma")  
4     print("2. Subtração")  
5     print("3. Multiplicação")  
6     print("4. Divisão")  
7     print("5. Sair")  
8  
9     opcao = int(input("Digite o número da operação: "))  
10  
11    if opcao == 5:  
12        print("Saindo da calculadora. Até logo!")  
13        break # Sai do laço se o usuário escolher a opção 5 (sair)  
14  
15    num1 = float(input("Digite o primeiro número: "))  
16    num2 = float(input("Digite o segundo número: "))
```

```
17  
18    if opcao == 1:  
19        print(f"{num1} + {num2} = {num1 + num2}")  
20    elif opcao == 2:  
21        print(f"{num1} - {num2} = {num1 - num2}")  
22    elif opcao == 3:  
23        print(f"{num1} * {num2} = {num1 * num2}")  
24    elif opcao == 4:  
25        if num2 != 0:  
26            print(f"{num1} / {num2} = {num1 / num2}")  
27        else:  
28            print("Erro: Divisão por zero não permitida.")  
29    else:  
30        print("Opção inválida. Tente novamente.")
```

# Exercício

- Criar um laço while para verificar se a senha digitada pelo usuário está correta.
  - O programa deve pedir ao usuário para digitar uma senha.
  - O laço deve continuar até que o usuário digite a senha correta.
  - Quando a senha correta for digitada, o programa deve imprimir "Acesso permitido".

# Exercício

- Criar um laço while para verificar se a senha digitada pelo usuário está correta.
  - O programa deve pedir ao usuário para digitar uma senha.
  - O laço deve continuar até que o usuário digite a senha correta.
  - Quando a senha correta for digitada, o programa deve imprimir "Acesso permitido".

```
1 senha_correta = "1234"
2 senha_digitada = ""
3
4 while senha_digitada != senha_correta:
5     senha_digitada = input("Digite a senha: ")
6
7 if senha_digitada == senha_correta:
8     print("Acesso permitido!")
9 else:
10    print("Senha incorreta. Tente novamente.")
```

# Dúvidas

