

Educação Profissional Paulista

Técnico em
**Ciência de
Dados**

Lógica de programação e algoritmos

Estruturas de controle: repetição

Aula 1

Código da aula: [DADOS]ANO1C3B2S12A1

Exposição



Objetivo da Aula

Explorar e apresentar, com apoio de pseudocódigos, os conceitos importantes de estruturas de controle de repetição.



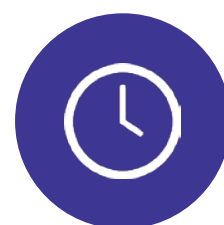
Competências da Unidade (Técnicas e Socioemocionais)

- 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.



Duração da Aula

50 minutos

Exposição

Recapitulação de algoritmos e lógica de programação

```
if tar_word in list_word and seed_word_p in list_wo
    count_word = len(list_word)

if count_word <= NEAR: # if number of word in e
    count_p += 1
else:
    positive_tar = []
    positive_sw = []
    for l_index, l_word in enumerate(list_word)
        if tar_word == l_word:
            positive_tar.append(l_index)
        elif seed_word_p == l_word:
            positive_sw.append(l_index)

    breaker = False
    for index_tar in positive_tar:
```

© Getty Images

- ✓ **Definição:** algoritmos são conjuntos de passos organizados, semelhantes a receitas, que guiam a resolução de problemas de forma eficiente;
- ✓ **Uso:** fundamentais na computação, algoritmos são "mapas" lógicos que possibilitam desde cálculos simples até inovações tecnológicas avançadas, como reconhecimento de voz;
- ✓ **Importância:** são a base de todos os softwares, permitindo a automação de tarefas e o desenvolvimento de soluções complexas.

Exposição

Recapitulação de algoritmos e lógica de programação

A importância de fluxogramas e de pseudocódigos

Eles otimizam o desenvolvimento, proporcionando uma compreensão visual e lógica do processo, que facilita a identificação de melhorias e acelera a transição para a implementação prática, independentemente da linguagem de programação ou do software usado para implementação.



© Getty Images

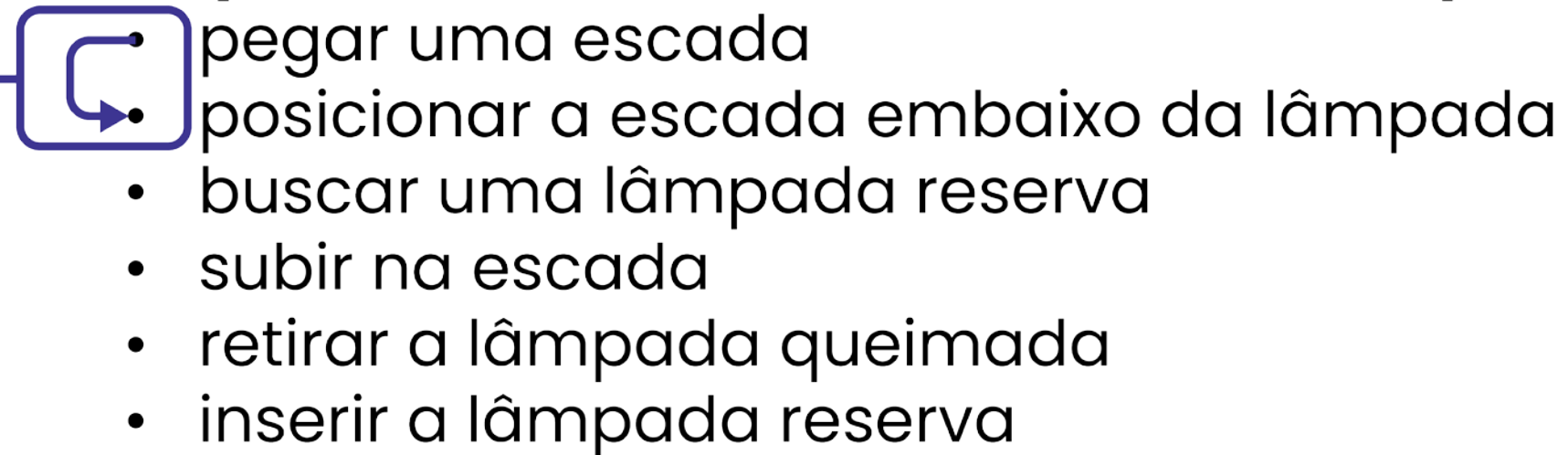
Recapitulação de estrutura sequencial

O que é uma estrutura sequencial?

- ✓ É a forma mais simples de **controle de fluxo em programação**;
- ✓ Consiste em uma **sequência linear de instruções**, em que uma ação ocorre após a outra, de maneira ordenada;
- ✓ É fundamental para executar **tarefas passo a passo**, uma após a outra, sem desvios controlados.

Recapitulação de pseudocódigos

Exemplo visto anteriormente (troca de lâmpada)

- 
- pegar uma escada
 - posicionar a escada embaixo da lâmpada
 - buscar uma lâmpada reserva
 - subir na escada
 - retirar a lâmpada queimada
 - inserir a lâmpada reserva

Estrutura
sequencial dos
pseudocódigos

Elaborado especialmente para o curso.



Tome nota

Os pseudocódigos permitem que a visualização das estruturas sequenciais sejam compreendidas exatamente como será no programa real. Elas seguem linha a linha, assim como nós fazemos leitura, da esquerda para a direita, de cima para baixo.

Estruturas de repetição

O que são?

- ✓ São conhecidas também como **laços** ou **loops**,
- ✓ São recursos fundamentais em programação, que permitem a execução repetida de um bloco de código enquanto uma condição específica for atendida.

Existem duas principais estruturas de repetição:

Laço "Enquanto" (While): repete um bloco de código enquanto uma condição é verdadeira.

Laço "Para" (For): executa um bloco de código em um número específico de vezes.

Estruturas de repetição

Importância das estruturas de repetição:



Eficiência

Permitem a automação de tarefas repetitivas, economizando tempo e esforço no desenvolvimento de software;



Flexibilidade

Adaptam-se a situações em que a quantidade de iterações não é conhecida antecipadamente, garantindo que o código execute o número adequado de vezes;



Manutenção

Simplificam a manutenção do código, pois reduzem a duplicação de instruções e tornam as atualizações mais consistentes;



Solução para problemas específicos

São essenciais para resolver problemas que exigem processamento repetitivo, como processamento de listas, cálculos iterativos e interação com conjuntos de dados;



Redução de erros

Ao evitar a repetição manual de código, as estruturas de repetição diminuem a probabilidade de erros.

Estruturas de repetição – ENQUANTO/FAÇA

O que é?

A **estrutura de repetição "Enquanto" (While)** é uma construção fundamental em programação que permite a execução de um bloco de código **enquanto uma condição específica permanece verdadeira**.



Na prática

- O bloco de código dentro do "Enquanto" é executado repetidamente enquanto a condição especificada permanecer verdadeira;
- A condição é avaliada antes da execução do bloco e, se for falsa inicialmente, o bloco não será executado.

Estruturas de repetição – ENQUANTO/FAÇA

Como utilizar?

ENQUANTO (condição) FAÇA

// Bloco de código a ser repetido

FIM ENQUANTO



Atenção!

Evite: é importante que essa condição seja sempre trabalhada dentro da estrutura de repetição, caso contrário ela estará sempre em execução e isso é um problema grave. Portanto, atente aos laços que possam ser infinitos.

Estruturas de repetição – ENQUANTO/FAÇA

Enunciado 1:

1. Escreva um programa em pseudocódigo que utilize a estrutura de repetição "Enquanto" para realizar a contagem de 1 a 5.
2. Crie um pseudocódigo para essa situação.

Requisitos:

- necessário iniciar a variável usada no "Enquanto" com valor antes da estrutura;
- alterar a variável da condição de repetição para evitar laços infinitos.



Momento
de **reflexão**

© Pexels

Estruturas de repetição – ENQUANTO/FAÇA

Solução em pseudocódigo:

1. contador \leftarrow 1
2. ENQUANTO contador \leq 5 FAÇA
3. Escrever "Contagem: ", contador
4. contador \leftarrow contador + 1
5. FIM ENQUANTO



Momento
de **reflexão**

© Pexels

Estruturas de repetição – ENQUANTO/FAÇA

Enunciado 2:

1. Escreva um programa em pseudocódigo que utilize a estrutura de repetição "Enquanto" para realizar uma contagem regressiva de 5 a 1.
2. Crie um pseudocódigo para essa situação.

Requisitos:

- necessário iniciar a variável usada no “Enquanto” com valor antes da estrutura;
- alterar a variável da condição de repetição para evitar laços infinitos.



Momento
de **reflexão**

© Pexels

Estruturas de repetição – ENQUANTO/FAÇA

Solução em pseudocódigo:

1. contador \leftarrow 5
2. ENQUANTO contador \geq 1 FAÇA
3. Escrever "Contagem: ", contador
4. contador \leftarrow contador - 1
5. FIM ENQUANTO



Momento
de **reflexão**

© Pexels

Estruturas de repetição – ENQUANTO/FAÇA

Enunciado 3:

1. Crie um programa em pseudocódigo que utilize a estrutura de repetição "Enquanto" para solicitar ao usuário a entrada de uma senha.
2. Enquanto a senha digitada não for igual a "1234", o programa deve continuar pedindo a senha. Quando a senha correta for inserida, exiba a mensagem "Acesso concedido".

Requisitos:

- necessário iniciar a variável senhaDigitada e a constante senhaCorreta.



Momento
de **reflexão**

© Pexels

Estruturas de repetição – ENQUANTO/FAÇA

Solução em pseudocódigo:

1. senhaCorreta <- "1234"
2. senhaDigitada <- ""
3. ENQUANTO senhaDigitada <> senhaCorreta FAÇA
4. Escrever "Digite a senha:"
5. Ler senhaDigitada
6. FIM ENQUANTO
7. Escrever "Acesso concedido"



Momento
de **reflexão**

© Pexels



Vamos
fazer um
quiz

Qual é o principal objetivo das estruturas de repetição em programação?

Realizar operações aritméticas

Controlar o fluxo de execução de um programa

Declarar variáveis e constantes

Realizar operações lógicas



Vamos
fazer um
quiz

Qual é o principal objetivo das estruturas de repetição em programação?

Realizar operações aritméticas

Controlar o fluxo de execução de um programa ✓

Declarar variáveis e constantes

Realizar operações lógicas

RESPOSTA CORRETA!

Destaca o principal propósito das estruturas de repetição, que é controlar a repetição de um bloco de código com base em uma condição.



Vamos
fazer um
quiz

Ao utilizar a estrutura de repetição "Enquanto" (While) em um algoritmo, qual é uma prática que deve ser evitada para garantir a correta execução do loop?

Iniciar a variável de controle fora da condicional do "Enquanto"

Modificar a variável de controle dentro da condição do "Enquanto"

Utilizar critérios de saída complexos dentro da condicional do "Enquanto"

Declarar novas variáveis dentro da condicional do "Enquanto"



Vamos
fazer um
quiz

Ao utilizar a estrutura de repetição "Enquanto" (While) em um algoritmo, qual é uma prática que deve ser evitada para garantir a correta execução do loop?

Iniciar a variável de controle fora da condicional do "Enquanto"

Modificar a variável de controle dentro da condição do "Enquanto"



Utilizar critérios de saída complexos dentro da condicional do "Enquanto"

Declarar novas variáveis dentro da condicional do "Enquanto"

RESPOSTA CORRETA!

Essa alternativa é a correta porque modificar a variável de controle dentro da condição do "Enquanto" pode levar a resultados inesperados e, potencialmente, causar loops infinitos.



Vamos
fazer um
quiz

Considere o seguinte algoritmo em pseudocódigo:

```
INÍCIO  
  soma ← 0  
  contador ← 1  
  ENQUANTO contador ≤ 5 FAÇA  
    soma ← soma + contador  
    contador ← contador + 1  
  FIM ENQUANTO  
  Escrever "Soma: ", soma  
FIM
```

Soma: 15

Soma: 10

Soma: 5

Soma: 20



Vamos
fazer um
quiz

Considere o seguinte algoritmo em pseudocódigo:

```
INÍCIO  
  soma ← 0  
  contador ← 1  
  ENQUANTO contador ≤ 5 FAÇA  
    soma ← soma + contador  
    contador ← contador + 1  
  FIM ENQUANTO  
  Escrever "Soma: ", soma  
FIM
```



Soma: 15

Soma: 10

Soma: 5

Soma: 20

RESPOSTA CORRETA!

O algoritmo realiza a soma dos números de 1 a 5, resultando em 15. Enquanto as opções B, C e D representam valores incorretos da soma, considerando o algoritmo apresentado.



O que nós
**aprendemos
hoje?**

© Getty Images

Hoje desenvolvemos:

- 1** Conhecimento que a estrutura de repetição "Enquanto" (While) é uma construção fundamental em programação que permite a execução de um bloco de código enquanto uma condição específica permanece verdadeira;
- 2** Compreensão de que é importante que essa condição seja sempre trabalhada dentro da estrutura de repetição, caso contrário ela estará sempre em execução e isso é um problema grave. Portanto, atente aos laços que possam ser infinitos;
- 3** Prática com QUIZ e com exercícios sem avaliação; solução de algoritmos com Estruturas de Controle de Repetição "Enquanto", que são largamente utilizadas na indústria para a construção de algoritmos profissionais.



Saiba mais

Entendendo o básico de Python para treinar estruturas de Controle – Parte 1

ALURA. *Python para Data Science: primeiros passos*.

01. Começando com Python. Disponível em:

<https://cursos.alura.com.br/course/python-data-science-primeiros-passos/task/122382>. Acesso em:

15 mar. 2024.

Entendendo o básico de Python para treinar estruturas de Controle – Parte 2

ALURA. *Python para Data Science: primeiros passos*.

02. Manipulando dados no Python. Disponível em:

<https://cursos.alura.com.br/course/python-data-science-primeiros-passos/task/123730>. Acesso em:

15 mar. 2024.

Referências da aula

ALURA. *Python para Data Science: primeiros passos*. 01. Começando com Python. Disponível em: <https://cursos.alura.com.br/course/python-data-science-primeiros-passos/task/122382>. Acesso em: 15 mar. 2024.

ALURA. *Python para Data Science: primeiros passos*. 02. Manipulando dados no Python. Disponível em: <https://cursos.alura.com.br/course/python-data-science-primeiros-passos/task/123730>. Acesso em: 15 mar. 2024.

FORBELLONE, A. L. V.; EBERSPÄCHER, H. F. *Lógica de programação: a construção de algoritmos e estruturas de dados com aplicações em Python*. Porto Alegre: Bookman, 2022.

Identidade visual: Imagens © Getty Images

Educação Profissional Paulista

Técnico em
**Ciência de
Dados**