

Educação Profissional Paulista

Técnico em
**Desenvolvimento
de Sistemas**

Estruturas de decisão compostas

Aninhamento de estruturas de decisão

Aula 2

[SIS]ANO1C1B2S12A2

Exposição



Objetivos da Aula

- Conhecer **estruturas de decisão compostas** por meio do fluxo de execução de programas;
- Compreender exemplos de **aninhamentos em estruturas de decisão**.



Competências da Unidade (Técnicas e Socioemocionais)

- Desenvolver **sistemas computacionais** utilizando ambientes de desenvolvimento;
- Migrar sistemas, implementando **rotinas e estruturas de dados mais eficazes**;
- Trabalhar a **curiosidade e a resiliência** em resolução de problemas computacionais.



Recursos Didáticos

- Recursos audiovisuais para exibição de vídeos e imagens;
- Caderno, canetas e lápis.



Duração da Aula

50 minutos

Desenvolvimento da aula

Olá! Hoje, exploraremos um conceito importante em programação chamado **aninhamento de estruturas de decisão**. Entenderemos como ele funciona e como ele pode nos ajudar a escrever programas mais complexos e poderosos.

- ✓ Conceito e **utilidade do aninhamento de estruturas de decisão**;
- ✓ Análise do **fluxo de execução de um programa com estruturas de decisão aninhadas**;
- ✓ Habilidade de **escrever programas com estruturas de decisão aninhadas**.

Vamos
fazer uma
atividade

Situação-problema

Imagine que você faz parte de um grupo que trabalha em um **departamento de tecnologia** de uma grande empresa que possui negócios em diferentes países.

Surgiu um desafio para o seu departamento: o desenvolvimento de um sistema que verifique em qual país você está para fornecer os acessos específicos para sua localidade.

O sistema são todos desenvolvidos em *Python* e **seu objetivo é identificar qual parâmetro pode ser usado para identificar o país de origem para o acesso e quais funções devem ser liberadas.**

Siga os passos do **próximo slide** para realizar a atividade.

Vamos
fazer uma
atividade

O relatório final deverá ser enviado
para o meio indicado pelo professor.

 15 minutos

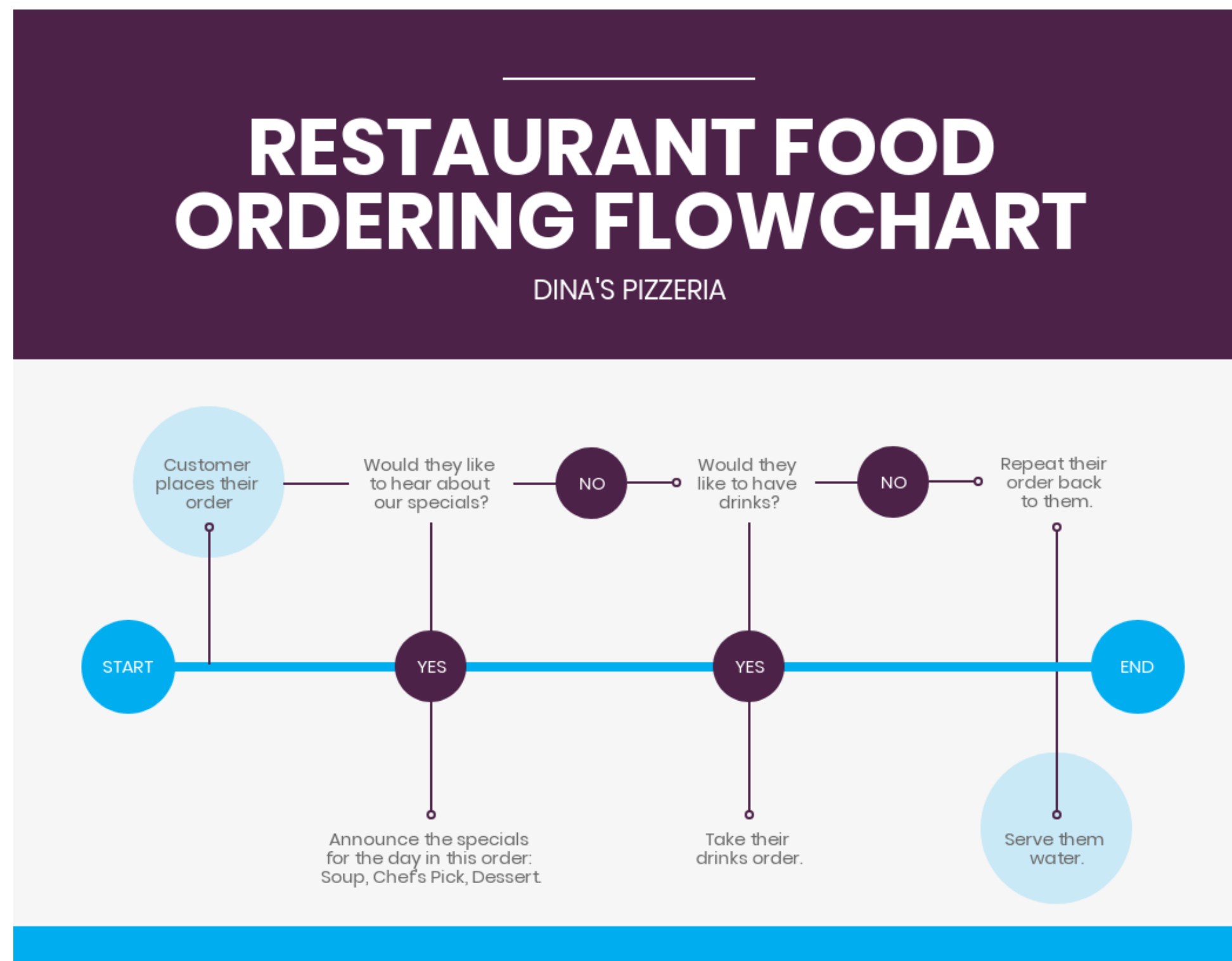
 Em grupos de até 5 pessoas

Situação-problema

- 1** Sem a necessidade de desenvolvimento prático, **monte um fluxograma** que ajude seu time a resolver esse problema;
- 2** **Qual foi a solução** que seu time pensou para solucionar esse grande problema?
- 3** **Desenvolva um fluxograma simples** que represente sua ideia para enviar **pelo meio que o professor indicar**.

Vamos
fazer uma
atividade

Situação-problema – Modelo de fluxograma



Reprodução – GASKIN, J. 7 tipos de fluxograma e como escolher o ideal. Venngage, 2023.
Disponível em: <https://venngage.com/blog/types-of-flowcharts/> Acesso em: 10 mar. 2024.

Exposição

Aninhamento de estruturas

O **aninhamento de estruturas de decisão** em Python significa que você pode ter uma instrução **if-elif-else** dentro de outra instrução **if-elif-else**.

Por exemplo, considere o seguinte cenário:

Você está escrevendo um programa para decidir se **deve ou não usar um guarda-chuva quando sair de casa**. Você pode tomar essa decisão com base em duas variáveis:

- Se está chovendo;
- Se você possui um guarda-chuva.



© Getty Images

Exposição

Exemplo

Neste exemplo, a primeira condição **verifica se está chovendo**. Se estiver chovendo, entra em uma nova estrutura de decisão para **verificar se você possui um guarda-chuva**.

```
esta_chovendo = True
possui_guarda_chuva = False

if esta_chovendo:
    if possui_guarda_chuva:
        print("Está chovendo, mas você tem um guarda-chuva. Pode sair.")
    else:
        print("Está chovendo e você não tem um guarda-chuva. Melhor ficar em casa.")
else:
    print("Não está chovendo. Pode sair.")
```

Casos de borda

Os casos de borda **são situações especiais que ocorrem nos extremos das condições que estamos testando.**

Precisam ser cuidadosamente considerados ao escrever programas, especialmente ao aninhar estruturas de decisão.

Por exemplo, vamos considerar um **programa que determina se uma pessoa é elegível para se aposentar.**

Em muitos lugares, a idade mínima para aposentadoria é 65 anos.

No entanto, **o que acontece quando a pessoa tem exatamente 65 anos? Este é um caso de borda.**



© Getty Images

Exposição

Exemplo

Neste exemplo, usamos o **operador** `>=` para tratar o caso de borda **em que a idade é exatamente 65**.

```
idade = 65

if idade < 65:
    print("Você ainda não é elegível para se aposentar.")
elif idade >= 65:
    print("Você é elegível para se aposentar.")
```


Exemplo prático

Escreva um programa que peça ao usuário para inserir a **temperatura atual**.

- Se a temperatura for **maior que 30 graus**, o programa deve avisar que **está quente**;
- Se ela estiver **entre 20 e 30 graus**, o programa deve informar que a temperatura **está agradável**;
- Caso contrário, o programa deve dizer que **está frio**.



Tome nota

Dentro de cada uma dessas condições, o programa deve verificar se também está chovendo e informar ao usuário se ele deve levar um guarda-chuva.

Exposição

Exemplo prático

```
temperatura = int(input("Qual é a temperatura atual? "))
esta_chovendo = input("Está chovendo? (sim/não) ") == "sim"

if temperatura > 30:
    print("Está quente.")
    if esta_chovendo:
        print("E também está chovendo. Você deve ficar dentro de casa.")
elif 20 <= temperatura <= 30:
    print("A temperatura está agradável.")
    if esta_chovendo:
        print("Mas está chovendo. Leve um guarda-chuva.")
else:
    print("Está frio.")
    if esta_chovendo:
        print("E também está chovendo. Fique quente e seco dentro de casa.")
```

Exemplo prático – Utilizando *AND*

```
temperatura = int(input("Qual é a temperatura atual? "))
esta_chovendo = input("Está chovendo? (sim/não) ") == "sim"

if temperatura > 30 and esta_chovendo:
    print("Está quente. E também está chovendo. Você deve ficar dentro de casa.")
elif temperatura > 30:
    print("Está quente.")
elif 20 <= temperatura <= 30 and esta_chovendo:
    print("A temperatura está agradável. Mas está chovendo. Leve um guarda-chuva.")
elif 20 <= temperatura <= 30:
    print("A temperatura está agradável.")
elif esta_chovendo:
    print("Está frio. E também está chovendo. Fique quente e seco dentro de casa.")
else:
    print("Está frio.")
```


Hoje desenvolvemos:

- 1 A aprendizagem de conceitos que permitem a **aplicação de decisão a partir de estruturas aninhadas**;
- 2 E praticamos **o desenho de um fluxograma** por meio de uma atividade representativa;
- 3 A visualização de **exemplos práticos** com códigos reais, que representam a **utilização das estruturas compostas**.

O que nós
**aprendemos
hoje?**

© Getty Images



Saiba mais

Agora que estamos mergulhando cada vez mais nos estudos sobre **estruturas condicionais**, que tal conhecermos um pouco mais sobre o funcionamento do **Desvio Condicional Composto** assistindo ao vídeo abaixo?

BÓSON TREINAMENTOS. *Estrutura de decisão condicional composta – SE, SENÃO* – Lógica de programação, 2021. Disponível em: <https://www.youtube.com/watch?v=TnIH4npQISs&t=1s>. Acesso em: 10 mar. 2024.

Referências da aula

BÓSON TREINAMENTOS. *Estrutura de decisão condicional composta – SE, SENÃO – Lógica de programação*, 2021. Disponível em: <https://www.youtube.com/watch?v=TnIH4npQlSs&t=1s>. Acesso em: 10 mar. 2024.

REIS, F. Estrutura de decisão condicional composta if ... else em Python. *Bóson treinamentos em Ciência e Tecnologia*, 2022. Disponível em: <https://www.bosontreinamentos.com.br/programacao-em-python/10-python-estrutura-de-decisao-condicional-composta-se-entao-senao/#:~:text=Em%20Python%2C%20outra%20estrutura%20de,a%20condi%C3%A7%C3%A3o%20n%C3%A3o%20for%20atendida> Acesso em: 10 mar. 2024.

Identidade visual: Imagens © Getty Images

Educação Profissional Paulista

Técnico em
**Desenvolvimento
de Sistemas**