

Educação Profissional Paulista

Técnico em
**Desenvolvimento
de Sistemas**

Estruturas de decisão compostas

Aninhamento de estruturas de decisão

Aula 4

[SIS]ANO1C1B2S12A4

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

- ✓ Demonstração de exemplos práticos **utilizando estruturas compostas por meio da construção de um jogo;**
- ✓ Exemplos práticos da aplicação de **estruturas compostas solucionando problemas cotidianos;**
- ✓ Aplicação prática dos conceitos compreendidos por meio da **construção de um esquema solucionando uma situação-problema apresentada.**



Que tal relembrarmos o conceito do jogo de adivinhação, agora incluindo também a aplicação das estruturas de decisão compostas?



© Getty Images

Exposição



ALURA. *Python: começando com a linguagem*. Testando valores. 01 A condição elif, [s.d.]. Disponível em: <https://cursos.alura.com.br/course/python-introducao-a-linguagem/task/22811>. Acesso em: 10 mar. 2024. Acesso para link externo.

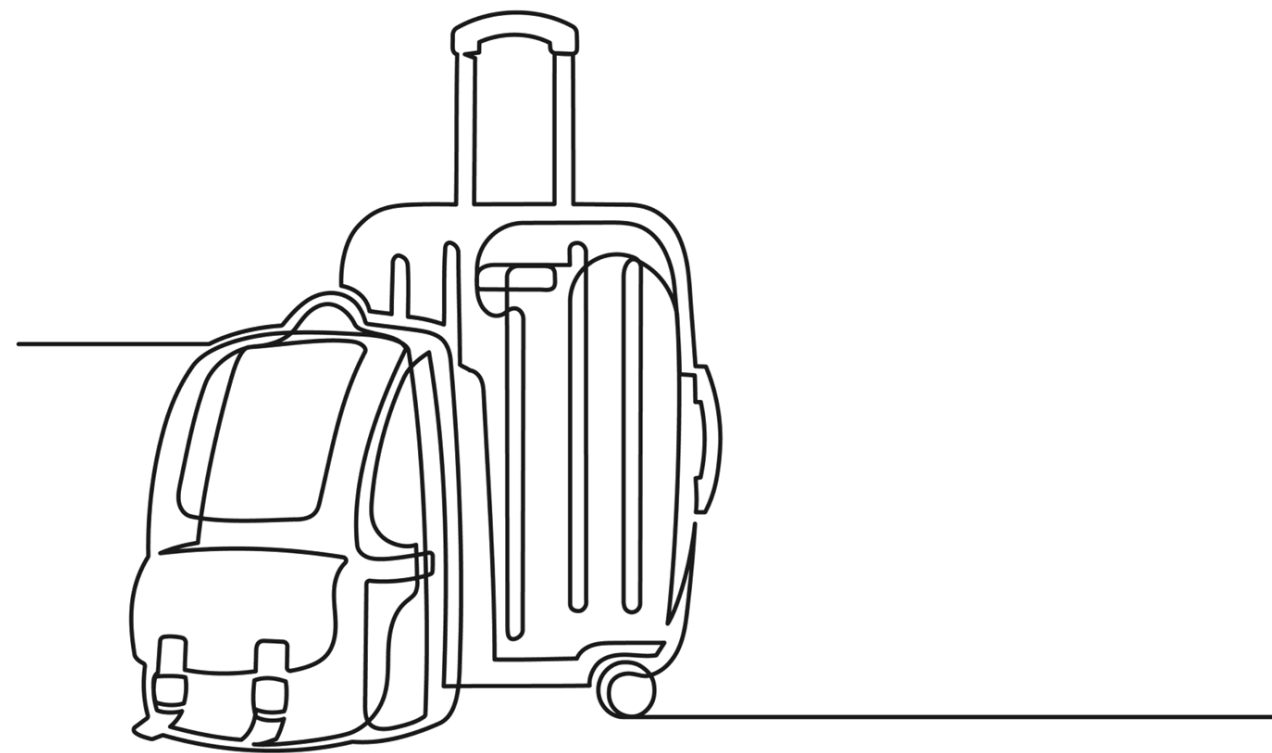
Exposição

Exemplos práticos

Vamos conhecer alguns exemplos cotidianos que exigem a resolução de problemas pela estrutura de decisão composta e compreender como esse processo pode ser feito a partir dos aprendizados em Python.

Exposição

Exemplos práticos



Situação-problema 1:

Planejamento de viagens com múltiplas condições

Contexto:

Um aplicativo de planejamento de viagens precisa decidir o **tipo de transporte que um usuário deve utilizar** com base na distância da viagem, no orçamento do usuário e na preferência por rapidez.

© Getty Images

Exemplos práticos

Situação-problema 1: planejamento de viagens com múltiplas condições

```
distancia = float(input("Digite a distância da viagem em km: "))
orcamento = float(input("Digite o seu orçamento em R$: "))
prioriza_rapidez = input("Você prioriza rapidez? (sim/não): ").lower() == "sim"
```

```
if distancia > 1000:
    if prioriza_rapidez:
        meio_transporte = "Avião"
    else:
        meio_transporte = "Ônibus" if orcamento < 300 else "Avião"
elif distancia > 300:
    if orcamento < 100:
        meio_transporte = "Ônibus"
    else:
        meio_transporte = "Carro" if prioriza_rapidez else "Trem"
else:
    meio_transporte = "Carro" if orcamento >= 50 else "Bicicleta"

print(f"Recomendamos utilizar: {meio_transporte}")
```


Exposição

Exemplos práticos



Situação-problema 2:

Sistema de recomendação de cursos com base em interesses e experiência prévia

Contexto:

Um *website* educacional quer **recomendar cursos aos seus usuários com base em seus interesses específicos** (como tecnologia, artes e ciências) **e no nível de experiência prévia** (iniciante, intermediário e avançado).

Exposição

Exemplos práticos

Situação-problema 2: sistema de recomendação de cursos com base em interesses e experiência prévia

```
interesse = input("Qual é o seu interesse principal? (tecnologia/artes/ciências): ").lower()
experiencia = input("Qual é o seu nível de experiência? (iniciante/intermediario/avancado): ").lower()

if interesse == "tecnologia":
    if experiencia == "iniciante":
        curso_recomendado = "Introdução à programação"
    elif experiencia == "intermediario":
        curso_recomendado = "Desenvolvimento Web"
    else:
        curso_recomendado = "Inteligência Artificial avançada"
elif interesse == "artes":
    if experiencia == "iniciante":
        curso_recomendado = "Fundamentos do desenho"
    elif experiencia == "intermediario":
        curso_recomendado = "Técnicas de pintura"
    else:
        curso_recomendado = "História da Arte Moderna"
else: # ciências
    if experiencia == "iniciante":
        curso_recomendado = "Ciências para todos"
    elif experiencia == "intermediario":
        curso_recomendado = "Química experimental"
    else:
        curso_recomendado = "Física teórica"

Printf("Curso recomendado: {curso_recomendado}")
```

Vamos
fazer uma
atividade

Situação-problema – Decisão de modo de transporte para o trabalho

Objetivo:

Criar um esquema ou fluxograma que ajude a decidir qual meio de transporte utilizar com base nas condições dadas.

Você está planejando seu trajeto diário para o trabalho e precisa **decidir qual modo de transporte utilizará**.

A decisão depende de vários fatores, incluindo o **clima** do dia, a **distância** do seu trabalho e se você tem **reuniões importantes**.

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

Vamos
fazer uma
atividade



25 minutos



Em grupos de até 5 pessoas

Situação-problema – Decisão de modo de transporte para o trabalho

1 Início: comece avaliando as condições do dia.
Verificar o clima: primeiro, verifique o clima. Se estiver chovendo, vá para o passo 2; se estiver ensolarado, vá para o passo 3.

2 Clima chuvoso:
Se a distância até o trabalho for curta (menos de 5 km), escolha o transporte público.
Se for uma distância mais longa, use o carro.

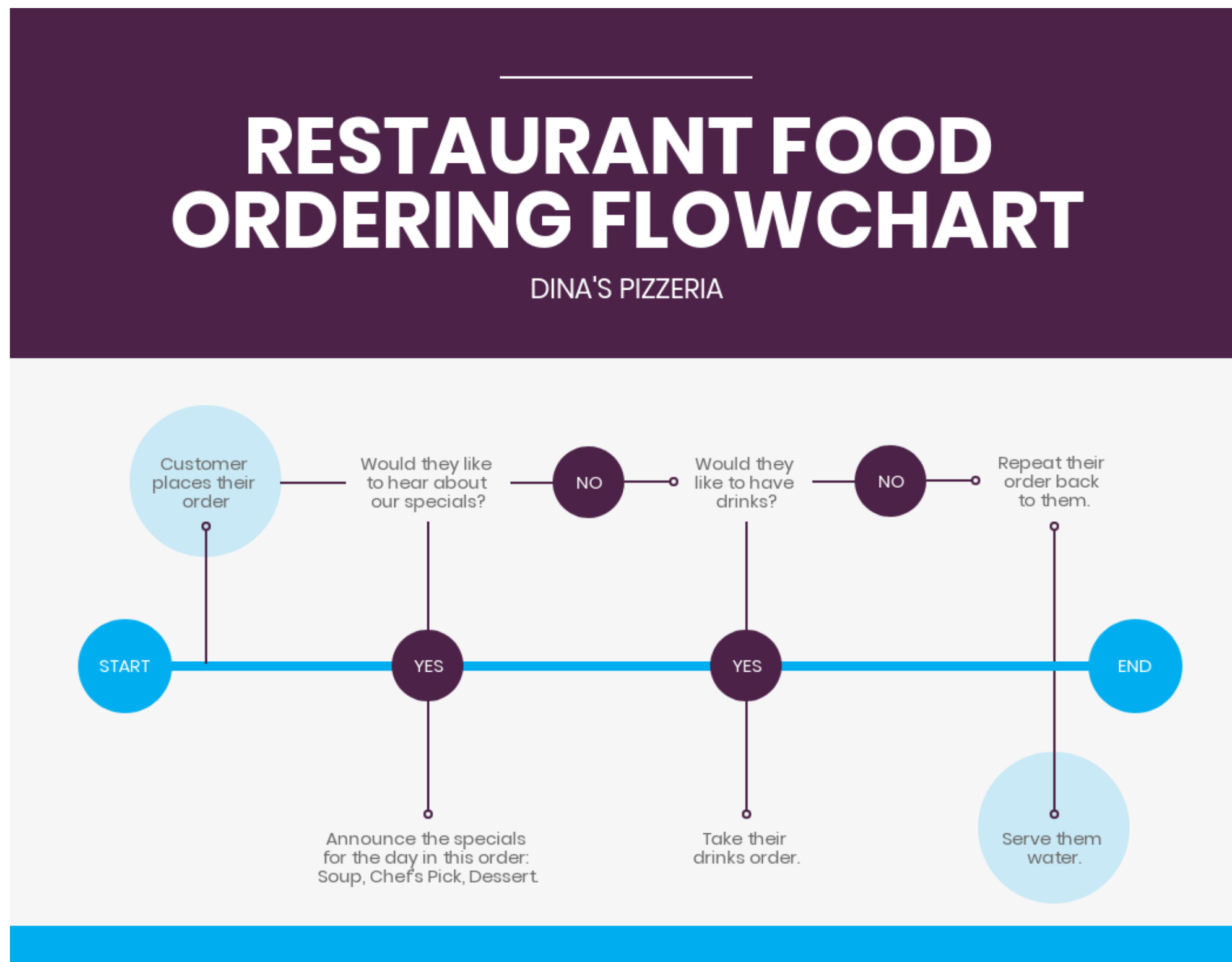
3 Clima ensolarado:
Verifique se há reuniões importantes no trabalho. Se sim, use o carro ou o transporte público, dependendo da distância.
3 Se não, considere usar uma bicicleta ou ir a pé se a distância for curta. Para distâncias maiores, use o transporte público ou o carro.

Decisão final: escolha o meio de transporte com base na análise das condições.

Fim: finalize o processo de decisão.

Vamos
fazer uma
atividade

Situação-problema – Modelo de fluxograma



Hoje desenvolvemos:

- 1 O **conceito de *elif*** dentro do universo de programação Python por meio da aplicação em um exemplo prático de **jogo com Python**;
- 2 E trabalhamos a compreensão de exemplos práticos de **estruturas compostas de decisão** utilizando os conceitos da linguagem **Python**;
- 3 Um **fluxograma** que utiliza conceitos compostos de estrutura de decisão para solução de problemas cotidianos.

O que nós
**aprendemos
hoje?**

© Getty Images



Saiba mais

Uma vez que estamos descobrindo como o **Python pode nos ajudar a resolver problemas cotidianos**, que tal aprender um pouco mais sobre sua aplicação na **resolução de problemas financeiros** por meio do vídeo abaixo?

PROGRAMANDO COM ANDERSON SOUZA. *Juros compostos com Python* – Crie seu próprio programa, [s.d.]. Disponível em: <https://www.youtube.com/watch?v=9mrkAYPgXTw>. Acesso em: 10 mar. 2024.

Referências da aula

ALURA. Python: começando com a linguagem. Testando valores. 01 A condição elif, [s.d.]. Disponível em: <https://cursos.alura.com.br/course/python-introducao-a-linguagem/task/22811> Acesso em: 10 mar. 2024.

PROGRAMANDO COM ANDERSON SOUZA. *Juros compostos com Python* - Crie seu próprio programa. Disponível em: <https://www.youtube.com/watch?v=9mrkAYPgXTw> 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**