**Texto base**

**5**

**Estrutura de Seleção Encadeada**

Gilberto Alves Pereira

***Resumo***

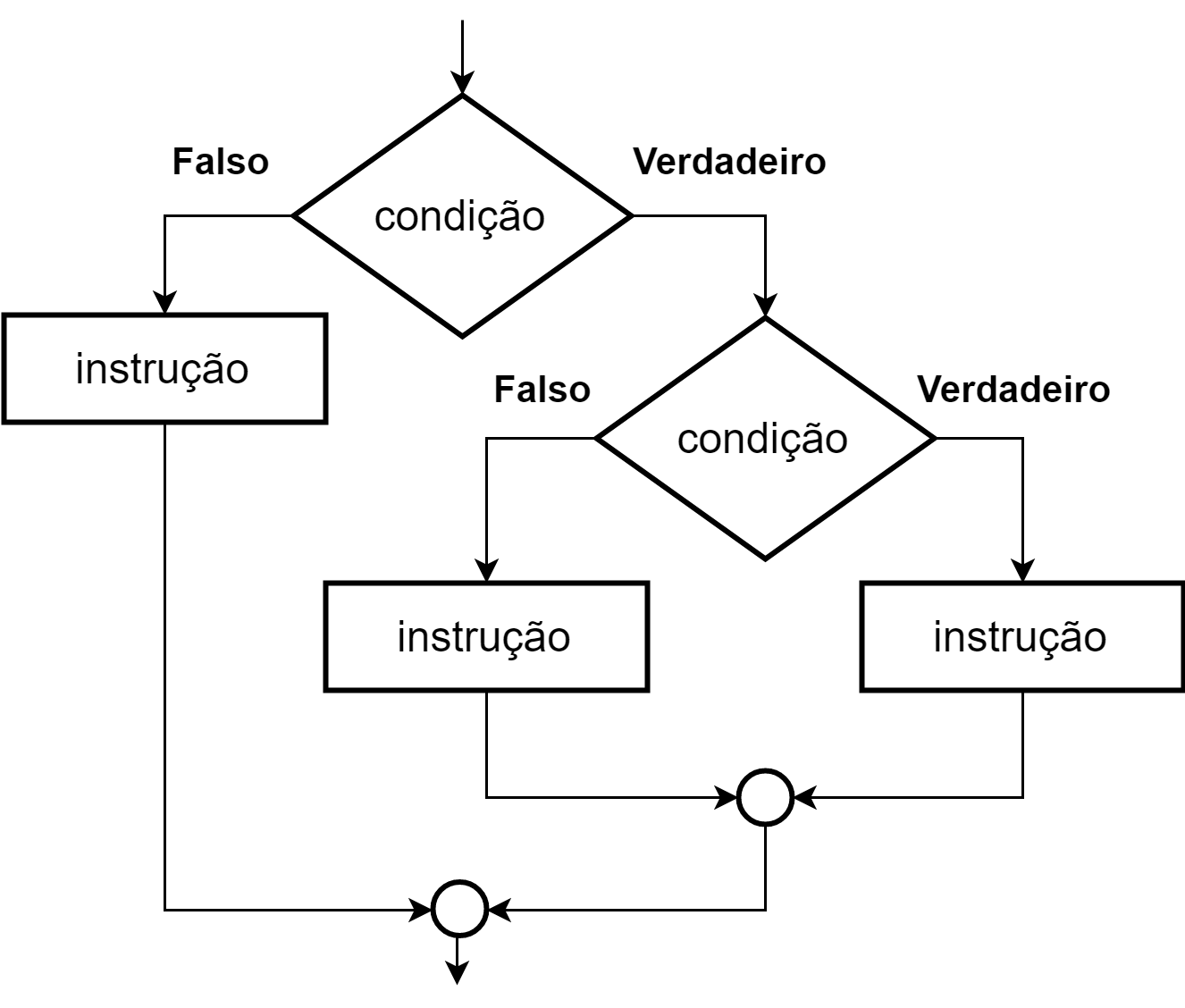
*Neste texto é discutido o conceito de estrutura de seleção encadeada ou aninhada. Como exemplificação mostra-se uma implementação de um exemplo usando fluxogramas, um segundo exemplo usando fluxogramas juntamente com uma simulação. Ao final é mostrado um terceiro exemplo usando fluxograma com sua conversão para Python.*

# 1.1. Estrutura de seleção encadeada

Tanto na estrutura de seleção simples quanto na seleção composta há apenas uma condição para ser avaliada e que condicionará o fluxo de execução do programa. Portanto, há no máximo uma bifurcação para um caminho, entre dois possíveis.

E quando é necessário resolver um problema que possui mais de dois caminhos possíveis? Para lidar com situações como essa, podemos usar uma estrutura de seleção encadeada (também conhecida como estrutura de seleção aninhada).

**Figura 5.1. Fluxograma de uma estrutura de seleção encadeada. Fonte Autor.**

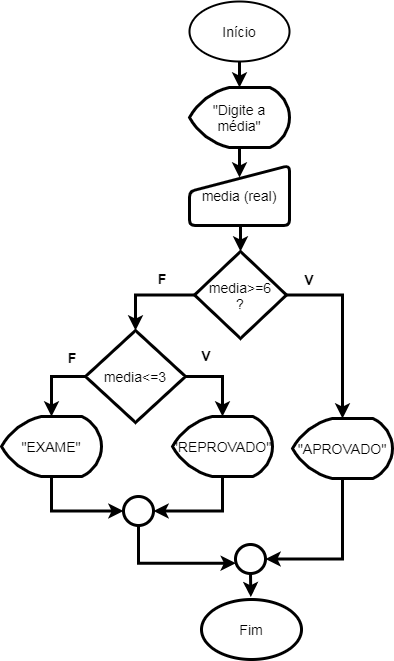
A estrutura de seleção encadeada ocorre quando uma seleção tem como ação uma outra seleção ou quando temos mais de duas possibilidades para uma decisão. A seleção encadeada ou aninhada é o agrupamento de uma ou várias seleções internas a uma seleção externa.

Exemplo 1:

Desenhar um fluxograma que obtém a média de um aluno e exibe a sua situação segundo a tabela:

**Tabela 5.1. Situações de aprovação de acordo com as médias, segundo o exemplo 1.**

| **Média** | **Situação** |
| --- | --- |
| maior ou igual a 6 | APROVADO |
| entre 3 e 6 | EXAME |
| menor que 3 | REPROVADO |



**Figura 5.2. Fluxograma referente ao exemplo 1. Fonte Autor.**

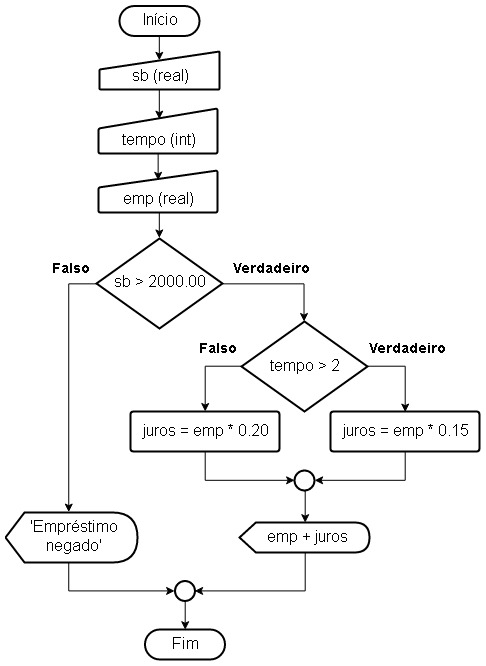
Perceba que nesse fluxograma temos 3 possibilidades de caminhos a serem percorridos, uma para cada status APROVADO, REPROVADO e EXAME. Na primeira decisão media >=6 definimos dois caminhos. No caminho da direita (verdadeiro), temos apenas uma possibilidade (media>=6 portanto aluno aprovado). Já no caminho da esquerda (falso), temos duas possibilidades: média <=3, ou seja, reprovado e média entre 3 e 6 (como ele caiu no caminho da esquerda, o valor da média é menor que 6). Precisamos, portanto, de mais de uma estrutura de seleção para que possamos decidir em qual dessas duas possibilidades a média testada se encontra. Perceba, ainda, que essa outra estrutura de seleção está dentro da primeira (no ramo esquerdo dela) e, portanto, é uma estrutura encadeada ou aninhada (ninho).

Exemplo 2:

Considere a seguinte situação: um banco oferece uma modalidade de empréstimo de parcela única para qualquer cliente cujo salário bruto ultrapasse R$ 2000,00. Essa modalidade também considera que, caso o cliente tenha mais de dois anos de contrato, serão cobrados juros de 15% sobre o valor emprestado, caso contrário os juros serão de 20%. Se o salário bruto não atingir o mínimo estipulado, o empréstimo será negado. O gerente do banco quer um programa que dados como entrada o salário bruto (em reais e centavos), o tempo de contrato do cliente com o banco (em anos) e o valor solicitado de empréstimo (em reais e centavos), exibe o valor da dívida do cliente, isto é, o valor emprestado acrescido de juros.

Note que com apenas os recursos das estruturas de seleção simples e composta não seria simples modelar esta situação, pois o enunciado evidencia que existem mais de dois possíveis fluxos de execução no algoritmo, que dependem dos dados passados como entrada.

Um fluxograma adequado para esse problema é ilustrado a seguir:

****

**Figura 5.3. Esquema do fluxograma e testes de simulação referentes ao exemplo 2. Fonte Autor.**

**1º Teste de mesa**

**Tabela 5.2. Primeiro teste de mesa referente ao exemplo 2.**

| **sb** | **tempo** | **emp** | **juros** | **sb > 2000.00** | **tempo > 2** | **Tela** |
| --- | --- | --- | --- | --- | --- | --- |
| 2000.00 | 5 | 100.00 | ‒ | 2000.00 > 2000.00 (False) | ‒ | 'empréstimo negado' |

**2º Teste de mesa**

**Tabela 5.3. Segundo teste de mesa referente ao exemplo 2.**

| **sb** | **tempo** | **emp** | **juros** | **sb > 2000.00** | **tempo > 2** | **Tela** |
| --- | --- | --- | --- | --- | --- | --- |
| 3000.00 | 1 | 500.00 | 100.00 | 3000.00 > 2000.00 (True) | 1 > 2 (False) | 600.00 |

**3º Teste de mesa**

**Tabela 5.4. Terceiro teste de mesa referente ao exemplo 2.**

| **sb** | **tempo** | **emp** | **juros** | **sb > 2000.00** | **tempo > 2** | **Tela** |
| --- | --- | --- | --- | --- | --- | --- |
| 2001.00 | 4 | 1000.00 | 150.00 | 2001.00 > 2000.00 (True) | 4 > 2 (True) | 1150.00 |

**1.2. Comparativo entre fluxograma e Python**

Abaixo podemos identificar um fluxograma e sua conversão para python.

Exemplo 3:

Desenhar um fluxograma que obtém uma idade e exibe uma mensagem segundo a tabela abaixo:

**Tabela 5.5. Classificações de acordo com a idade, segundo o exemplo 3.**

| **Idade** | **Classificação** |
| --- | --- |
| menor que 10 | INFANTIL |
| entre 10 e 18 | TEEN |
| maior ou igual a 18 | SENIOR |

| **Fluxograma** | **Python** |
| --- | --- |
|  | print (“Digite a Idade”)  idade=int (input ( ) )  if idade < 10:  print (“Infatil”)  else:  if idade<18:  print (“Teen”)  else:  print (“Senior”) |

**Figura 5.4. Esquema do fluxograma e seu respectivo código em Python referentes ao exemplo 3. Fonte Autor.**

**1.3. Referências**

DIERBACH, C. “Introduction to Computer Science Using Python: A Computational Problem-Solving Focus”1st Edition, New York: Wiley, 2012.