Fatec
Rio Claro

Linguagem de Programação I

Prof. Orlando Saraiva Júnior
orlando.nascimento@fatec.sp.gov.br

Variáveis do tipo Lógico

Variáveis do tipo Lógico

Fatec
Rio Claro

Muitas vezes, queremos armazenar um conteúdo simples: verdadeiro ou falso. Neste caso, utilizamos o tipo lógico ou booleano.

Em Python, escrevemos **True** para verdadeiro e **False** para falso.

Operadores relacionais

Fatec Rio Claro

Para realizarmos comparaçẽs lógicas, utilizamos operadores relacionais.

Operador	Operação	Símbolo Matemático
==	Igualdade	=
>	Maior que	>
<	Menor que	<
!=	diferente	≠
>=	Maior ou igual	≥
<=	Menor ou igual	≤

Operadores relacionais

Fatec
Rio Claro

O resultado de uma comparação com esses operadores é um valor do tipo lógico, ou seja, **True** ou **False**.

O que a operação a seguir retorna ?

a = 1

b = 5

c = 2

d = 1

Operadores relacionais

Fatec Rio Claro

O resultado de uma comparação com esses operadores é um valor do tipo lógico, ou seja, **True** ou **False**.

O que a operação a seguir retorna ?

a = 1

b = 5

c = 2

d = 1

```
a == b

b > a

a < b

a == d

c <= b

d != a
```

Operadores relacionais

Fatec
Rio Claro

O resultado de uma comparação com esses operadores é um valor do tipo lógico, ou seja, **True** ou **False**.

O que a operação a seguir retorna ?

a = 1

b = 5

c = 2

d = 1

a == b **False**

b > a **True**

a < b **True**

a == d **True**

c <= b **True**

d != a **False**

Operadores Lógicos

Fatec Rio Claro

Para agrupar operações lógicas booleana, utilizaremos operadores lógicos. Python suporta três operadores básicos:

Operador Python	Operação
not	não
and	e
or	ou

Operador not

Fatec Rio Claro

O operador not (não) é o mais simples, pois precisa apenas de um operando. A operação de negação é chamada inversão, pois um valor verdadeiro se torna falso.

V	Not V
True (verdadeiro)	**False** (Falso)
False (Falso)	**True** (verdadeiro)

Operador and

Fatec Rio Claro

O operador and (e) tem sua tabela verdade representada a seguir. O operador and resulta verdadeiro apenas quando seus dois operandos forem verdadeiros:

V1	V2	V1 and V2
True (verdadeiro)	True (verdadeiro)	True (verdadeiro)
True (verdadeiro)	False (Falso)	False (Falso)
False (Falso)	True (verdadeiro)	False (Falso)
False (Falso)	False (Falso)	False (Falso)

Operador or

Fatec Rio Claro

O operador or (e) tem sua tabela verdade representada a seguir. Se apenas um dos seus operandos for verdadeiro, ou se os dois forem, o resultado será verdadeiro

V1	V2	V1 or V2
True (verdadeiro)	**True** (verdadeiro)	**True** (verdadeiro)
True (verdadeiro)	False (Falso)	**True** (verdadeiro)
False (Falso)	**True** (verdadeiro)	**True** (verdadeiro)
False (Falso)	False (Falso)	False (Falso)

Expressões Lógicas

Fatec
Rio Claro

Os operadores lógicos podem ser combinados em expressões lógicas mais complexas. Quando uma expressão tiver mais de um operador lógico, avalia-se o operador **not** primeiro, seguindo do operador **and** e finalmente o operador **or.**

```
True or False and not True
```

Expressões Lógicas

Fatec
Rio Claro

Os operadores lógicos podem ser combinados em expressões lógicas mais complexas. Quando uma expressão tiver mais de um operador lógico, avalia-se o operador **not** primeiro, seguindo do operador **and** e finalmente o operador **or.**

True or False and not True
True or False and False
True or False
True

Expressões Lógicas

Fatec
Rio Claro

Os operadores relacionais também podem ser utilizados em expressões com operadores lógicos.

```
salario > 1000 and idade > 18
```

Expressões Lógicas

Fatec Rio Claro

Os operadores relacionais também podem ser utilizados em expressões com operadores lógicos.

salario = 100

idade = 20

```
salario > 1000 and idade > 18
```

Expressões Lógicas

Fatec Rio Claro

Os operadores relacionais também podem ser utilizados em expressões com operadores lógicos.

salario = 100

idade = 20

salario > 1000 **and** idade > 18
100 > 1000 and 20 > 18
False and **True**
False

Atividade

Fatec Rio Claro

Observe as variáveis abaixo.

salario = 100

idade = 20

Registre se o retorno é verdadeiro ou falso:

salario > 1000 and idade > 18 (x)True ( ) False

salario != 2000 and idade <= 25 (x)True ( ) False

salario != 2000 and not idade <= 25 ( )True (x) False

Condição

Condição

Fatec
Rio Claro

Nem sempre todas as linhas de nosso programa serão executadas. Muitas vezes, sermá mais interessante descobrir que partes do programa devem ser executadas com base no resultado de uma condição. A base dessas condições consistirá em expressões lógicas que permitam representar escolhas em programas.

A execução condicional de parte do programa é um dos fundamentos da programação de computadores.

DICA

Fatec
Rio Claro

Python é uma das poucas linguagens de programação que utiliza o deslocamento do texto à direita (recuo ou avanço) para marcar o início e o fim do bloco.

Outras linguagens contam com palavras especiais, como BEGIN e END, em Pascal, ou as chaves ({ }), em C/C++ ou Java.

Embora o deslocamento à direita usando espaços em branco deixe nosso código mais elegante, ele exige maiores cuidados ao digitar o programa.

Verificação de Número Par ou Ímpar

Fatec Rio Claro

Crie um programa que peça ao usuário para digitar um número inteiro e verifique se ele é par ou ímpar.

Regras:

- Se o número for divisível por 2, exiba a mensagem:

"O número X é par."

- Caso contrário, exiba a mensagem: "O número X é ímpar."

Tarefa:

1. Solicite ao usuário um número inteiro.
2. Use a estrutura if para verificar se o número é par ou ímpar.
3. Exiba o resultado de acordo com a regra acima.

Resolução

Cláusula else

Fatec Rio Claro

A cláusula **else** é usada para definir o que deve acontecer quando a condição do if for falsa. Ela funciona como um plano B:

Se a condição no if for verdadeira, o bloco do if é executado. Caso contrário, o bloco do else é executado.

```
idade = int(input("Digite sua idade: "))

if idade >= 18:

    print("Você é maior de idade.")

else:

    print("Você é menor de idade.")
```

Estruturas aninhadas

Estruturas aninhadas

Fatec
Rio Claro

Nem sempre nossos programas serão tão simples. Muitas vezes, precisaremos aninhar vários if para obter o comportamento desejado do programa. Aninhar, nesse caso, é utilizar um if dentro do outro.

Vejamos um exemplo: Os planos da empresa telefônica XYZ. Abaixo de 200 minutos, a empresa cobra R$0,20 por minuto. Entre 200 e 400 minutos, o preço é de R$0,18. Acima de 400 minutos, o preço por minuto é de R$0,15.

Estruturas aninhadas

Fatec
Rio Claro

```
minutos = int(input("Quantos minutos você utilizou este mês: "))
if minutos < 200:
    preco = 0.20
else:
    if minutos < 400:
        preco = 0.18
    else:
        preco = 0.15
total = minutos * preco
print("Você vai pagar este mês R$", total)
```

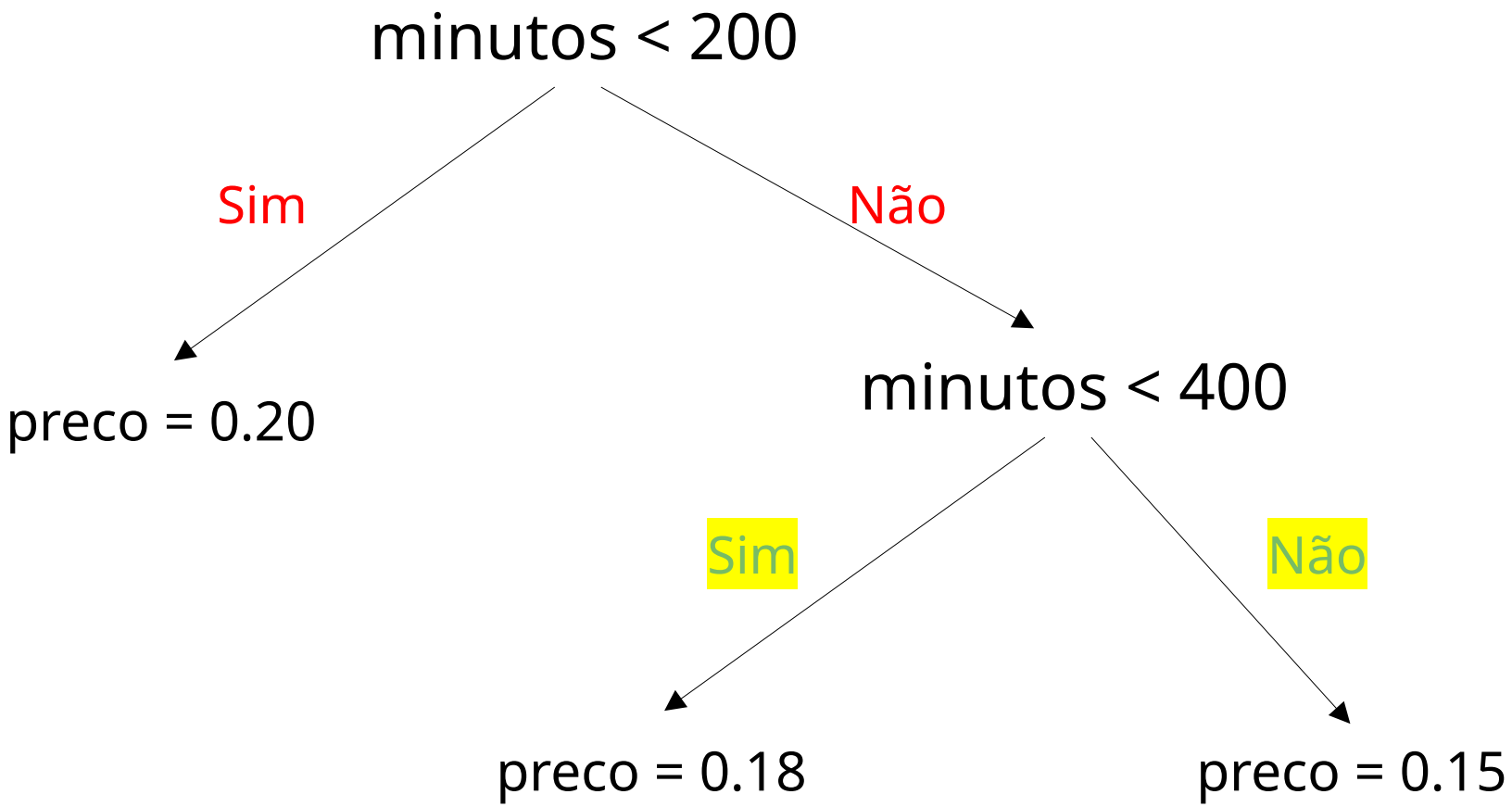
Estruturas aninhadas

Fatec
Rio Claro

```
minutos = int(input("Quantos minutos você utilizou este mês: "))
if minutos < 200:
    preco = 0.20
else:
    if minutos < 400:
        preco = 0.18
    else:
        preco = 0.15
total = minutos * preco
print("Você vai pagar este mês R$", total)
```

Estruturas aninhadas

Fatec
Rio Claro



Cláusula elif

Fatec
Rio Claro

Python apresenta uma solução muito interessate ao problema de múltiplos ifs aninhados. A cláusula elif substitui um par else if, mas sem criar outro nível de estrutura, evitano problemas de deslocamento desnecessário à direita.

No exemplo a seguir, substituímos o if-else aninhado por um elif, tornando o código mais limpo e fácil de entender.

Cláusula elif

Fatec
Rio Claro

```
minutos = int(input("Quantos minutos você utilizou este mês: "))
if minutos < 200:
    preco = 0.20
elif minutos < 400:
    preco = 0.18
else:
    preco = 0.15

total = minutos * preco
print("Você vai pagar este mês R$", total)
```

Fatec
Rio Claro

Dúvidas

Prof. Orlando Saraiva Júnior
orlando.nascimento@fatec.sp.gov.br

Exercício de Fixação

Fatec Rio Claro

Realize a lista de exercícios do dia.

Disponível no github do professor.

Tempo para Atividade