

FIAP GRADUAÇÃO

ANÁLISE E DESENVOLVIMENTO DE SISTEMAS

Computational Thinking

PROF. EDUARDO GONDO

I Problema 1

PROBLEMA 1: Escreva um algoritmo que calcula o aumento salarial de um trabalhador.

- ▶ Como resolvemos este problema no papel?
- ▶ Quais informações o usuário precisa fornecer para resolvermos este problema?
- ▶ Depois de respondidas as questões, estamos prontos para escrever o algoritmo em Python...

I Problema 2

PROBLEMA 2: Escreva um algoritmo que calcula o aumento salarial de um trabalhador. Se o trabalhador for um gerente, ele ganha um bônus de 1,5% a mais no salário.

Comando de decisão/seleção - if



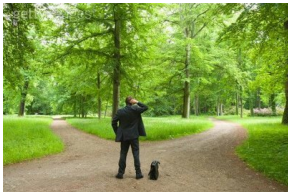
```
1  if combustivel acabando:
2      paro no posto
3      encho o tanque
4  sigo minha viagem

1  #comando if
2  if <condicao>:
3      instrucao_1
4      instrucao_2
5      ...
6      instrucao_n
7
8  instrucao_fora_do_bloco
```

I Problema 3

PROBLEMA 3: Escreva um algoritmo que calcula a idade de uma pessoa com base no ano de nascimento e verifica se ela têm mais de 18 anos.

Comando de decisão/seleção - if/else



```
1  if <condicao>:  
2      vou pela direita  
3  else:  
4      vou pela esquerda  
5  ...
```

```
1  #comando if/else  
2  if <condicao>:  
3      instr_if_1  
4      ...  
5      instr_if_n  
6  else:  
7      instr_else_1  
8      ...  
9      instr_else_n  
10  
11  instr_fora_bloco
```

I Problema 4

PROBLEMA 4: Escreva um algoritmo que calcula a idade de uma pessoa com base no ano de nascimento e verifica se ela têm mais de 18 anos ou menos de 18 anos.

Problema 5

PROBLEMA 5: A raiz quadrada é uma operação que apenas aceita números positivos. Escreva um algoritmo que lê um número qualquer e retorna a raiz quadrada desse número se possível. Use a função `math.sqrt(<numero>)` para calcular a raiz quadrada em Python. Note que, para usar essa função, você terá que importar o módulo **math** antes.

Operadores condicionais

- ▶ uma <condicao> é uma expressão booleana que pode retornar dois valores: True ou False
- ▶ para criar expressões booleanas usamos os seguintes operadores:

significado	Python
igual	<code>==</code>
menor	<code><</code>
maior	<code>></code>
diferente	<code>!=</code>
menor ou igual	<code><=</code>
maior ou igual	<code>>=</code>

Tabela: Operadores condicionais

Comandos de decisão encadeados

- ▶ algumas vezes nos deparamos com problemas em que nem o `if` e nem o `if/else` são suficientes
- ▶ quando isso acontece, devemos encadear os comandos de decisão `else + if` ou `elif`

PROBLEMA 6: Suponha que você está escrevendo um programa para verificar o vencedor de uma partida de futebol. A entrada do seu programa são 2 números inteiros não negativos representando a quantidade de gols do Time A e a quantidade de gols do Time B, respectivamente. Você deverá exibir na tela o time ganhador da partida ou se houve empate.

Problema 7

PROBLEMA 7: Escreva um algoritmo que lê o salário de um funcionário e mostra qual o percentual de desconto que será aplicado para sua contribuição ao INSS. Use a tabela abaixo para calcular o desconto:

salário contribuição	alíquota/valor
até R\$ 1.693,72	8%
de R\$ 1.693,73 até R\$ 2.822,90	9%
de R\$ 2.822,91 até R\$ 5.645,80	11%
acima de R\$ 5.645,80	11% sobre R\$ 5.645,80

Tabela: Contribuição INSS 2019

Por exemplo, um trabalhador com salário de R\$ 2.000,00 o percentual de desconto será de 9%. Quem ganha R\$ 8.000,00 terá um desconto de 11% sobre o teto da aposentadoria.

I Problema 8

PROBLEMA 8: Escreva um algoritmo que recebe dois números e um caractere (representando uma das operações matemáticas(+,-,*,/)) e calcula o valor da operação matemática, ou seja, se a entrada for 5, * e 6 então seu programa deverá mostrar 30.

Problema 9

PROBLEMA 9: Escreva um algoritmo que calcule o que deve ser pago por um produto, considerando o preço normal de etiqueta e a escolha da condição de pagamento. Utilize os códigos da tabela a seguir para ler qual a condição de pagamento escolhida e efetuar o cálculo adequado.

código	condição de pagamento
1	A vista em dinheiro ou cheque, recebe 10% de desconto
2	A vista no cartão de crédito, recebe 5% de desconto
3	Em duas vezes, preço normal de etiqueta sem juros
4	Em quatro vezes, preço normal de etiqueta mais juros de 7%

Conectores lógicos

- ▶ algumas vezes precisamos combinar expressões lógicas
- ▶ os conectores lógicos **and** e **or** são os responsáveis por essa combinação no Python
- ▶ note que o resultado da expressão lógica combinada continua sendo True ou False
- ▶ por exemplo, na situação da estrada com posto, qual outra situação me levaria a parar no posto?

Tabela verdade

- ▶ sejam x e y duas expressões lógicas
- ▶ cada uma dessas expressões pode assumir o valor **verdadeiro** ou **falso**
- ▶ para os dois valores possíveis de x e y vamos construir uma tabela com x *and* y e x *or* y

x	y	x <i>and</i> y	x <i>or</i> y	<i>not</i> x
V	V	V	V	F
V	F	F	V	F
F	V	F	V	V
F	F	F	F	V

Tabela: Tabela verdade

I Problema 10

PROBLEMA 10: Escreva um algoritmo que recebe três números inteiros e imprime eles em ordem crescente.

Problema 11

PROBLEMA 11: Desenvolva um algoritmo que informe se uma data é válida ou não. O algoritmo deverá ler 2 números inteiros, que representem o dia e o mês e informar se é um dia do mês válido. Desconsidere os casos de ano bissexto, ou seja, fevereiro têm 28 dias.

I Problema 12

PROBLEMA 12: Agora, vamos acrescentar na verificação de data os casos de ano bissexto, ou seja, o ano que fevereiro tem 29 dias. Um ano é bissexto se:

1. o ano for divisível por 4
2. anos múltiplos de 100, não são bissextos
3. quando o ano for divisível por 400 ele é bissexto
4. as últimas regras prevalecem sobre as primeiras

Referência Bibliográfica

- ▶ Puga e Rissetti - Lógica de Programação e Estrutura de Dados
- ▶ Ascêncio e Campos - Fundamentos da Programação de Computadores
- ▶ Forbelone e Eberspacher - Lógica de programação: a construção de algoritmos e estruturas de dados
- ▶ Documentação do Python - <https://docs.python.org/3.8/>
- ▶ Python Programming For Beginners: Learn The Basics Of Python Programming (Python Crash Course, Programming for Dummies) (English Edition). Kindle
- ▶ Python: 3 Manuscripts in 1 book: - Python Programming For Beginners - Python Programming For Intermediates - Python Programming for Advanced (English Edition). Kindle

| Copyleft

Copyleft © 2021 Prof. Eduardo Gondo Todos direitos liberados.
Reprodução ou divulgação total ou parcial deste documento é liberada.