

Programação em Python

Estruturas de repetição (while)

Prof. Daniel Di Domenico

https://github.com/danidomenico/gex003_algprog

Slides cortesia da profa. Andrea Charão (UFSM) e do prof. João V. F. Lima (UFSM)

Estruturas de repetição

- **Problema:** elabore um algoritmo para somar 10 números

Estruturas de repetição

- Programa do problema anterior:

```
n1 = int(input("Digite um numero: "))
n2 = int(input("Digite um numero: "))
n3 = int(input("Digite um numero: "))
n4 = int(input("Digite um numero: "))
n5 = int(input("Digite um numero: "))
n6 = int(input("Digite um numero: "))
n7 = int(input("Digite um numero: "))
n8 = int(input("Digite um numero: "))
n9 = int(input("Digite um numero: "))
n10 = int(input("Digite um numero: "))

soma = n1 + n2 + n3 + n4 + n5 + n6 + n7 + n8 + n9 + n10

print(soma)
```

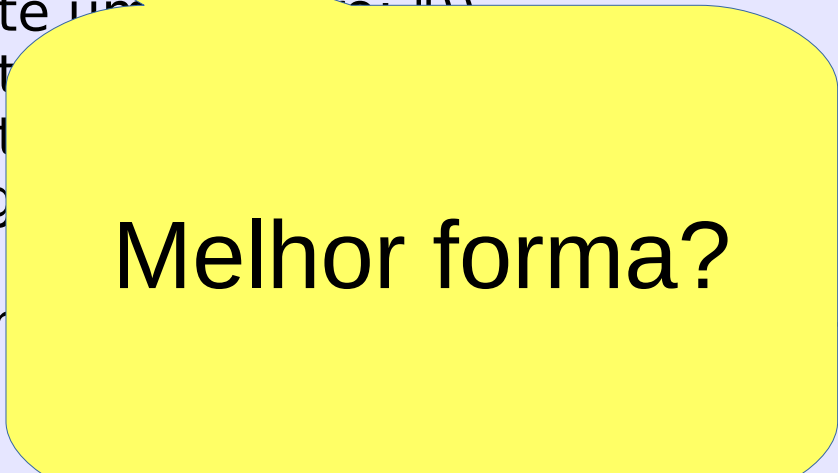
Estruturas de repetição

- Programa do problema anterior:

```
n1 = int(input("Digite um numero: "))
n2 = int(input("Digite um numero: "))
n3 = int(input("Digite um numero: "))
n4 = int(input("Digite um numero: "))
n5 = int(input("Digite um numero: "))
n6 = int(input("Digite um numero: "))
n7 = int(input("Digite um numero: "))
n8 = int(input("Digite um numero: "))
n9 = int(input("Digite um numero: "))
n10 = int(input("Digite um numero: "))

soma = n1 + n2 + n3 + n4 + n5 + n6 + n7 + n8 + n9 + n10

print(soma)
```



Melhor forma?

Estruturas de repetição

- **Solução:** ideal seria repetir o trecho abaixo 10x

```
n1 = int(input("Digite um numero: "))  
soma = soma + n1
```

Estruturas de repetição

- Em Python:

while condição:

bloco de comando(s)
a repetir

- Exemplo:

```
num = 1
```

```
while num <= 10:
```

```
# repete os comandos 10 vezes  
num = num + 1
```

Estruturas de repetição

- **Solução:** somar 10 números

```
conta = 0
soma = 0
while conta < 10:
    n1 = int(input("Digite um numero: "))
    soma = soma + n1
    conta = conta + 1

print(soma, conta)
```

Estruturas de repetição

- **Problema:** elabore um algoritmo para calcular a média individual de cada aluno, para uma turma de 10 alunos.
- Considere:
 - Média geral: $(\text{nota1} + \text{nota2})/2$
- Informe se o aluno foi aprovado ou reprovado. Para o aluno ser aprovado, sua média deve ser ≥ 6 .

Estruturas de repetição

- Algoritmo Python:

```
numAlunos = 1
while numAlunos <= 10:
    N1 = float(input("Qual a primeira nota: "))
    N2 = float(input("Qual a segunda nota: "))
    media = (N1 + N2) / 2
    print("A media é:", media)
    if media >= 7:
        print("Aluno aprovado")
    else:
        print("Aluno reprovado")
    numAlunos = numAlunos + 1
```

Estruturas de repetição

- Atenção ao **recuo** (endentação!!!!)
- Não esquecer de recuar à direita

while condição:
bloco de comando(s)

CERTO!

while condição:
bloco de comando(s)

ERRADO!
Deveria ter recuo!

Estruturas de repetição

- Atenção ao **recuo** (endentaçãoooo!!!)
- Usar sempre o mesmo recuo em todo programa

```
while condição1:  
    # comando(s)  
while condição2:  
    # comando(s)
```

CERTO!

```
while condição1:  
    # comando(s)  
while condição2:  
    # comando(s)
```

ERRADO!
Recuos diferentes!

Estruturas de repetição

- **ATENÇÃO** a condição !!!
- Para usar corretamente esta estrutura de repetição, é importante formular uma condição adequada.
- Se a condição for sempre **verdadeira**, a repetição nunca vai **parar** (laço infinito).

Estruturas de repetição

- **Problema:** elabore um programa que leia repetidamente uma **senha**, até que ela esteja correta. Quando isso acontecer, imprimir a mensagem "Seja bem vindo!".
- **DICA:** a condição para término da repetição está relacionada à entrada do programa.

Estruturas de repetição

- Programa em Python:

```
senha = input("Digite a senha: ")

while senha != "python":
    senha = input("Digite a senha: ")

print("Seja bem-vindo!")
```

Estruturas de repetição

- Programa em Python:

```
senha = input("Digite a senha: ")  
  
while senha != "python":  
    senha = input("Digite a senha: ")  
  
print("Seja bem-vindo")
```

Término da repetição:
senha != "python"

Estruturas de repetição

- **Problema:** elabore um programa que leia 10 valores e imprima **o maior deles** no final.
- Usar uma variável para guardar o maior número lido. Essa variável é testada e possivelmente atualizada a cada repetição.

Estruturas de repetição

- Programa em Python:

```
n = 0
maior = int(input("Digite um valor: "))
while n < 9:
    valor = int(input("Digite um valor: "))
    if valor > maior:
        maior = valor
    n = n + 1

print("Maior valor =", maior)
```

Guia para programar repetição

- Determine as variáveis/valores/comandos que precisam estar **ANTES** do laço
 - é o caso sem repetição
- Determine a condição do laço
 - precisa ter um operador lógico (>, <, ==, !=, etc)
 - simples ou composto (and/or) ?
- Dentro do laço
 - Não esqueça do recuo à direita
 - Atualize as variáveis verificadas na condição
 - Assim você evita **laços infinitos**

Exercícios

- 1) Faça um programa que calcule a soma dos 100 primeiros termos da série harmônica:
$$1/1 + 1/2 + 1/3 + 1/4 + 1/5 + \dots$$
- 2) Faça um programa que solicite números até que o usuário digite 0. Após isso, informe a quantidade de números pares e ímpares lidos.