



Introdução à Programação

Aula 06: Repetição (parte 1): instrução while

Prof. Eduardo Corrêa

Data 26/03/2024

Tópicos da Aula

- Temas desta aula:
 - O que é repetição?
 - Comando de Repetição **while**
 - Exercícios propostos

Exercício

- Faça um programa que leia 4 números inteiros via teclado e que descubra e imprima **qual é o maior** e **qual é o menor** dos 4 números digitados.

Exercício - Solução

```
n = int(input('digite o primeiro número: '))  
maior = n; menor = n
```

```
n = int(input('digite o segundo número: '))  
if n > maior: maior = n  
elif n < menor: menor = n
```

```
n = int(input('digite o terceiro número: '))  
if n > maior: maior = n  
elif n < menor: menor = n
```

```
n = int(input('digite o quarto número: '))  
if n > maior: maior = n  
elif n < menor: menor = n
```

```
print('O menor número é:', menor)  
print('O maior número é:', maior)
```

Exercício - Solução

```
n = int(input('digite o primeiro número: ')) ←  
maior = n; menor = n
```

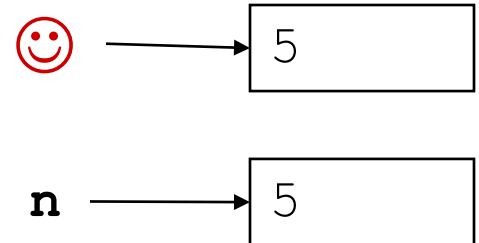
```
n = int(input('digite o segundo número: '))  
if n > maior: maior = n  
elif n < menor: menor = n
```

```
n = int(input('digite o terceiro número: '))  
if n > maior: maior = n  
elif n < menor: menor = n
```

```
n = int(input('digite o quarto número: '))  
if n > maior: maior = n  
elif n < menor: menor = n
```

```
print('O menor número é:', menor)  
print('O maior número é:', maior)
```

- Imagine que o usuário 😊 digitou 5 no primeiro número.



Exercício - Solução

```
n = int(input('digite o primeiro número: '))  
maior = n; menor = n
```

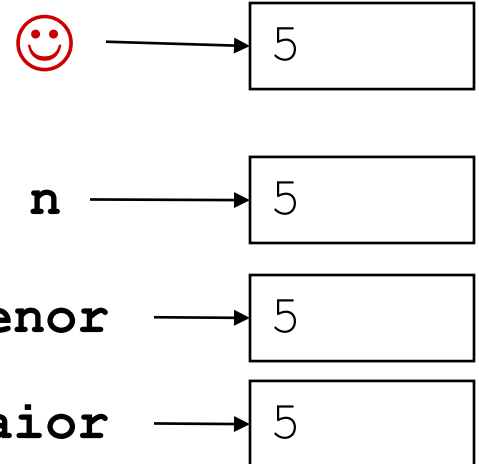
```
n = int(input('digite o segundo número: '))  
if n > maior: maior = n  
elif n < menor: menor = n
```

```
n = int(input('digite o terceiro número: '))  
if n > maior: maior = n  
elif n < menor: menor = n
```

```
n = int(input('digite o quarto número: '))  
if n > maior: maior = n  
elif n < menor: menor = n
```

```
print('O menor número é:', menor)  
print('O maior número é:', maior)
```

- Como até agora só um número foi digitado, o **maior** e o **menor** correspondem a este único número.



Exercício - Solução

```
n = int(input('digite o primeiro número: '))  
maior = n; menor = n
```

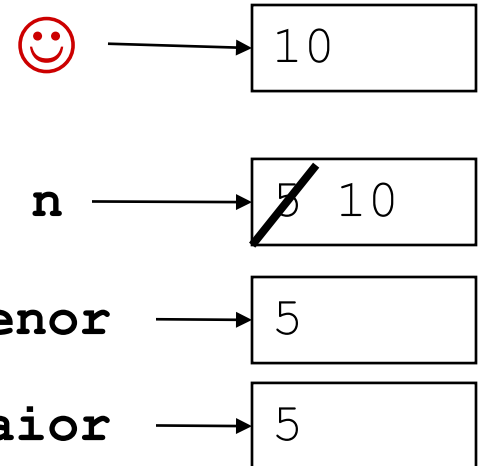
```
n = int(input('digite o segundo número: ')) ←  
if n > maior: maior = n  
elif n < menor: menor = n
```

```
n = int(input('digite o terceiro número: '))  
if n > maior: maior = n  
elif n < menor: menor = n
```

```
n = int(input('digite o quarto número: '))  
if n > maior: maior = n  
elif n < menor: menor = n
```

```
print('O menor número é:', menor)  
print('O maior número é:', maior)
```

■ Agora o usuário 😊 digitou 10.



Exercício - Solução

```
n = int(input('digite o primeiro número: '))
maior = n; menor = n
```

```
n = int(input('digite o segundo número: '))
if n > maior: maior = n ←
elif n < menor: menor = n
```

```
n = int(input('digite o terceiro número: '))
if n > maior: maior = n
elif n < menor: menor = n
```

```
n = int(input('digite o quarto número: '))
if n > maior: maior = n
elif n < menor: menor = n
```

```
print('O menor número é:', menor)
print('O maior número é:', maior)
```

■ 10 é maior do que o maior anterior.

■ Conteúdo da variável **maior** é alterado, pois teste do **if** dá **True**

😊 → 10

n → 10

menor → 5

maior → ~~5~~ 10

Exercício - Solução

```
n = int(input('digite o primeiro número: '))
maior = n; menor = n
```

10 não é menor do que o menor anterior.

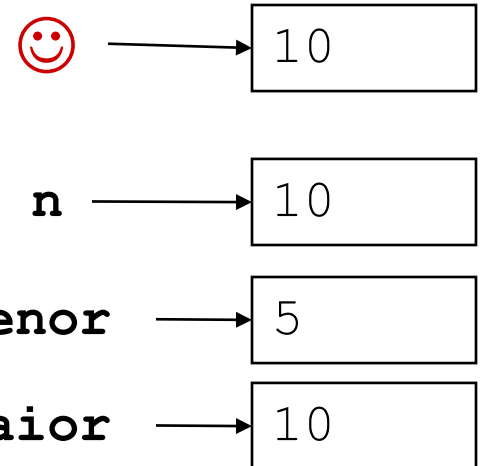
```
n = int(input('digite o segundo número: '))
if n > maior: maior = n
elif n < menor: menor = n
```

Conteúdo da variável **menor** é preservado.

```
n = int(input('digite o terceiro número: '))
if n > maior: maior = n
elif n < menor: menor = n
```

```
n = int(input('digite o quarto número: '))
if n > maior: maior = n
elif n < menor: menor = n
```

```
print('O menor número é:', menor)
print('O maior número é:', maior)
```



Exercício - Solução

```
n = int(input('digite o primeiro número: '))  
maior = n; menor = n
```

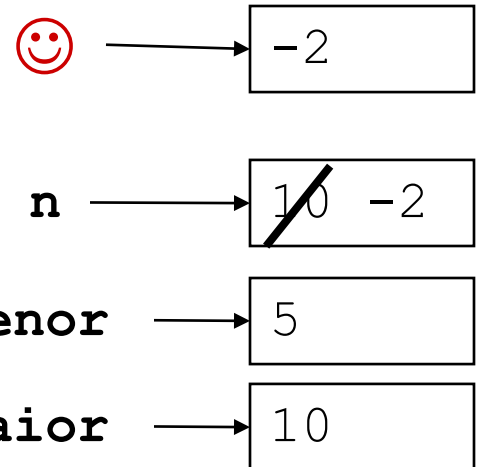
```
n = int(input('digite o segundo número: '))  
if n > maior: maior = n  
elif n < menor: menor = n
```

```
n = int(input('digite o terceiro número: ')) ←  
if n > maior: maior = n  
elif n < menor: menor = n
```

```
n = int(input('digite o quarto número: '))  
if n > maior: maior = n  
elif n < menor: menor = n
```

```
print('O menor número é:', menor)  
print('O maior número é:', maior)
```

■ Usuário 😊
digitou -2



Exercício - Solução

```
n = int(input('digite o primeiro número: '))  
maior = n; menor = n
```

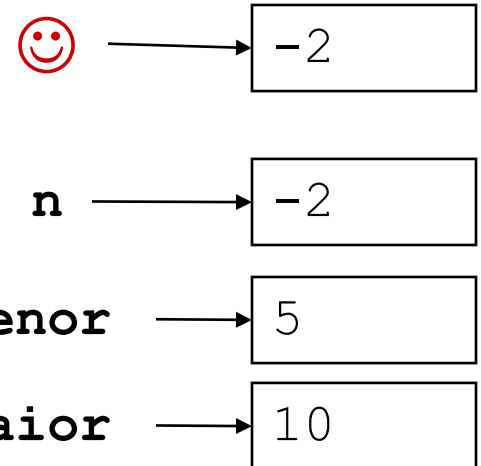
```
n = int(input('digite o segundo número: '))  
if n > maior: maior = n  
elif n < menor: menor = n
```

```
n = int(input('digite o terceiro número: '))  
if n > maior: maior = n  
elif n < menor: menor = n
```

```
n = int(input('digite o quarto número: '))  
if n > maior: maior = n  
elif n < menor: menor = n
```

```
print('O menor número é:', menor)  
print('O maior número é:', maior)
```

- -2 não é maior do que o valor atual de **maior**. Nada é feito...



Exercício - Solução

```
n = int(input('digite o primeiro número: '))
maior = n; menor = n
```

```
n = int(input('digite o segundo número: '))
if n > maior: maior = n
elif n < menor: menor = n
```

```
n = int(input('digite o terceiro número: '))
if n > maior: maior = n
elif n < menor: menor = n ←
```

```
n = int(input('digite o quarto número: '))
if n > maior: maior = n
elif n < menor: menor = n
```

```
print('O menor número é:', menor)
print('O maior número é:', maior)
```

■ -2 é menor do que o valor atual de **menor**.

■ Conteúdo de **menor** é alterado, pois teste do **elif** dá **True**

😊 → -2

n → -2

menor → ~~5~~ -2

maior → 10

Exercício - Solução

```
n = int(input('digite o primeiro número: '))  
maior = n; menor = n
```

```
n = int(input('digite o segundo número: '))  
if n > maior: maior = n  
elif n < menor: menor = n
```

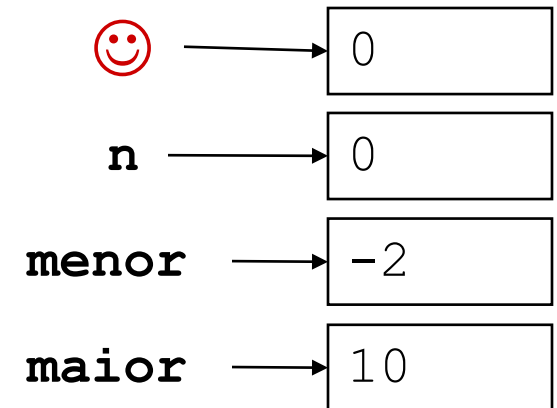
```
n = int(input('digite o terceiro número: '))  
if n > maior: maior = n  
elif n < menor: menor = n
```

```
n = int(input('digite o quarto número: '))  
if n > maior: maior = n  
elif n < menor: menor = n
```

```
print('O menor número é:', menor)  
print('O maior número é:', maior)
```

■ Por fim, imagine que usuário digitou 0. 😊

■ Não dá **True** no **if** nem no **elif**.



Exercício - Solução

```
n = int(input('digite o primeiro número: '))  
maior = n; menor = n
```

```
n = int(input('digite o segundo número: '))  
if n > maior: maior = n  
elif n < menor: menor = n
```

```
n = int(input('digite o terceiro número: '))  
if n > maior: maior = n  
elif n < menor: menor = n
```

```
n = int(input('digite o quarto número: '))  
if n > maior: maior = n  
elif n < menor: menor = n
```

```
print('O menor número é:', menor) ←  
print('O maior número é:', maior) ←
```

■ O resultado final é impresso na tela.



→ 0

n

→ 0

menor

→ -2

maior

→ 10

```
>>> %Run maior_menor.py  
digite o primeiro número: 5  
digite o segundo número: 10  
digite o terceiro número: -2  
digite o quarto número: 0  
O menor número é: -2  
O maior número é: 10
```


Comandos de Repetição

- Imagine se eu quisesse descobrir o maior e o menor entre **50 números** digitados.
- Uma solução similar a que acabou de ser apresentada resultaria em um programa muito grande, com 50 instruções `input()` e 50 comandos **if-elif** !
- Para que seja possível elaborar uma solução que não resulte em um programa enorme (em muitos casos, impossível de ser criado) são usados os comandos de repetição:
 - **while**
 - **for** (será abordado em outra aula)

Comandos de Repetição

- Um comando de repetição permite que uma ou mais instruções sejam executadas repetidas vezes até que uma determinada condição seja atingida.
- Na Aula 01 (que abordou algoritmos) apresentamos este tipo de comando.
- Examine o seguinte problema prático, que só pode ser resolvida com o uso de um comando de repetição:
 - Calcular o valor da seguinte série, onde N é escolhido pelo usuário:

$$H = 1 + (1 / 2) + (1 / 3) + \dots + (1 / N).$$

-  -> Como resolver este problema se o usuário pode colocar qualquer valor para N?????

Comandos de Repetição

- $H = 1 + (1 / 2) + (1 / 3) + \dots + (1 / N)$
- Algoritmo em pseudocódigo (*Obs.: fizemos esse exercício na Aula01*).

início

ler(N)

$H := 0$

$D := 1$

enquanto $D \leq N$ **faça:**

$H := H + 1/D$

$D := D + 1$

imprimir(H)

fim.

Comandos de Repetição

- Exemplos de problemas que podem ser resolvidos com os comandos de repetição:
 - Cálculo de séries numéricas.
 - Descobrir qual é o maior entre 10000 números lidos via arquivo.
 - Contabilizar o total de votos para eliminar um participante do BBB, com a participação milhões de votantes.
 - Contabilizar a renda média do brasileiro a partir dos dados do censo.
 - *E inúmeras outras...*

Comando while

- **while** (enquanto) é o comando do Python que implementa a “repetição com teste **no início** do laço”
- Laço (ou Loop): bloco de código que é executado repetidas vezes.
- Funciona de modo idêntico ao apresentado no exemplo do pseudocódigo:
 - **enquanto** (condição for verdadeiro) **faça**

Comando while

- Ler a nota final de 10 alunos de uma turma e calcular e imprimir a média da turma.

```
tot_alunos = 10
```

```
i = 1
```

```
soma = 0
```

```
while (i <= tot_alunos):  
    print('Digite a nota ', i, ': ')  
    nota = float(input())  
    soma = soma + nota  
    i = i + 1
```

```
media = soma / tot_alunos
```

```
print('Media Turma:', round(media, 1))
```

- Princípio básico de funcionamento do programa:
- **“O laço while é executado enquanto a variável i não ultrapassa o valor 10”.**
- Dentro do laço a nota de cada aluno vai sendo somada.

Comando while

- Ler a nota final de 10 alunos de uma turma e calcular e imprimir a média da turma.

```
tot_alunos = 10
```

```
i = 1 ←
```

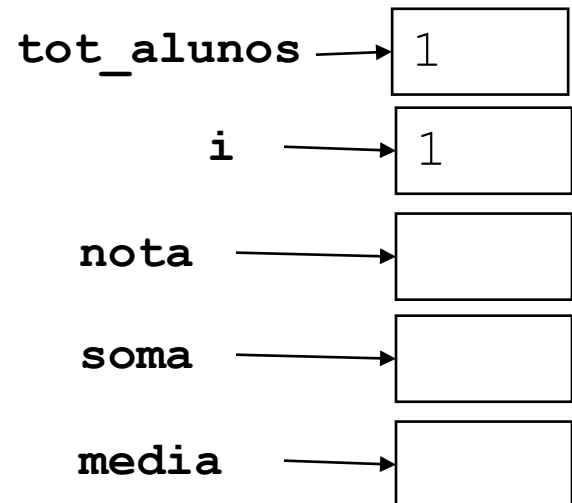
```
soma = 0
```

```
while (i <= tot_alunos):  
    print('Digite a nota ', i, ': ')  
    nota = float(input())  
    soma = soma + nota  
    i = i + 1
```

```
media = soma / tot_alunos
```

```
print('Media Turma:', round(media, 1))
```

- *i* é uma variável importantíssima!!
- Ela é usada para controlar o loop.
- O loop será mantido enquanto *i* for menor ou igual a 10.



Comando while

- Ler a nota final de 10 alunos de uma turma e calcular e imprimir a média da turma.

```
tot_alunos = 10
```

```
i = 1
```

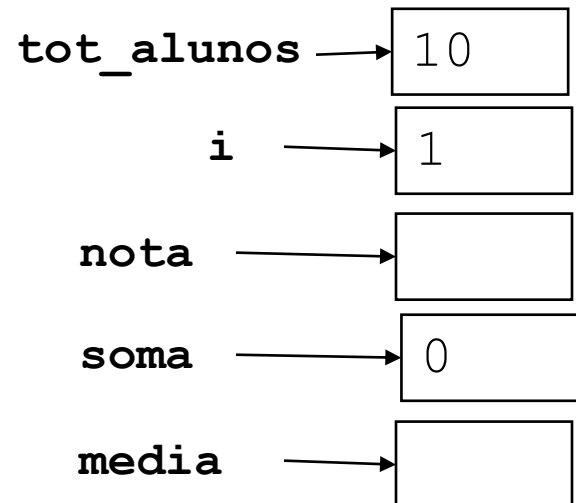
```
soma = 0
```

```
while (i <= tot_alunos):  
    print('Digite a nota ', i, ': ')  
    nota = float(input())  
    soma = soma + nota  
    i = i + 1
```

```
media = soma / tot_alunos
```

```
print('Media Turma:', round(media, 1))
```

- Inicializo a variável **soma** com o valor 0. Dentro do laço receberei a nota de cada aluno e atualizarei esta variável.
- No final ela terá a soma das notas de todos os alunos.



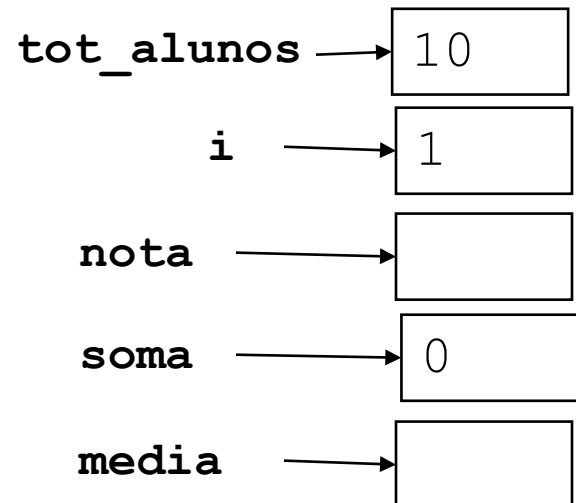
Comando while

- Ler a nota final de 10 alunos de uma turma e calcular e imprimir a média da turma.

```
tot_alunos = 10
i = 1
soma = 0
while (i <= tot_alunos): ←
    print('Digite a nota ', i, ': ')
    nota = float(input())
    soma = soma + nota
    i = i + 1
```

```
media = soma / tot_alunos
print('Media Turma:', round(media, 1))
```

- Início do laço!!!
- O conteúdo de **i** é menor ou igual do que o de **tot_alunos** (10) ?
- **SIM** (**True**), então entra no laço!!!



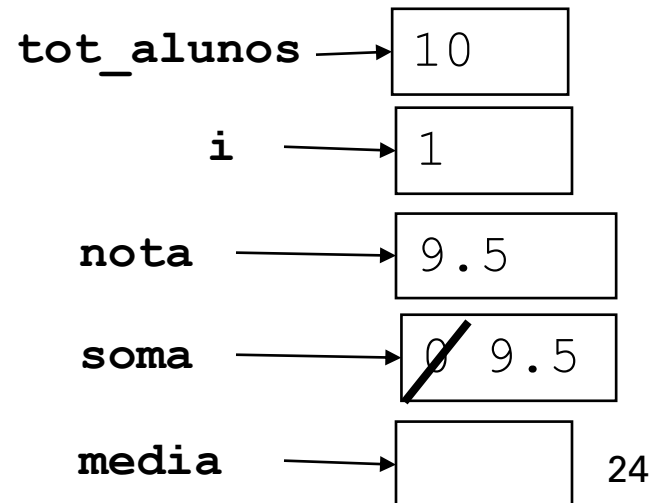
Comando while

- Ler a nota final de 10 alunos de uma turma e calcular e imprimir a média da turma.

```
tot_alunos = 10
i = 1
soma = 0
while (i <= tot_alunos):
    print('Digite a nota ', i, ': ')
    nota = float(input()) ←
    soma = soma + nota ←
    i = i + 1
```

```
media = soma / tot_alunos
print('Media Turma:', round(media, 1))
```

- Suponha que a primeira nota digitada seja 9.5
- **soma** é igual a zero + 9.5 = 9.5



Comando while

- Ler a nota final de 10 alunos de uma turma e calcular e imprimir a média da turma.

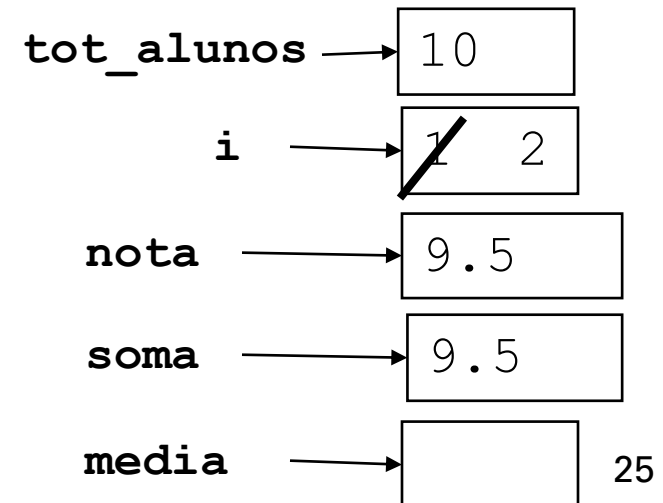
```
tot_alunos = 10
i = 1
soma = 0
while (i <= tot_alunos):
    print('Digite a nota ', i, ': ')
    nota = float(input())
    soma = soma + nota
    i = i + 1 ←
```

```
media = soma / tot_alunos
print('Media Turma:', round(media, 1))
```

- IMPORTANTÍSSIMO!

- É preciso incrementar *i*, pois ela é que controla meu loop!

- Quando *i* chegar em 11, o laço tem que parar (pois 10 notas terão sido digitadas)



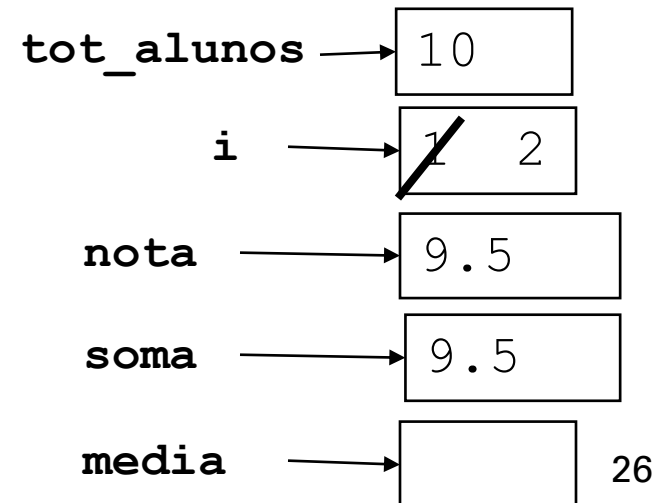
Comando while

- Ler a nota final de 10 alunos de uma turma e calcular e imprimir a média da turma.

```
tot_alunos = 10
i = 1
soma = 0
→ while (i <= tot_alunos):
    ↑ print('Digite a nota ', i, ': ')
    nota = float(input())
    soma = soma + nota
    ← i = i + 1
```

```
media = soma / tot_alunos
print('Media Turma:', round(media, 1))
```

- Quando encontra o último comando do bloco, o controle da execução do programa retorna ao **while** para testar se condição de encerramento do laço foi atingida.



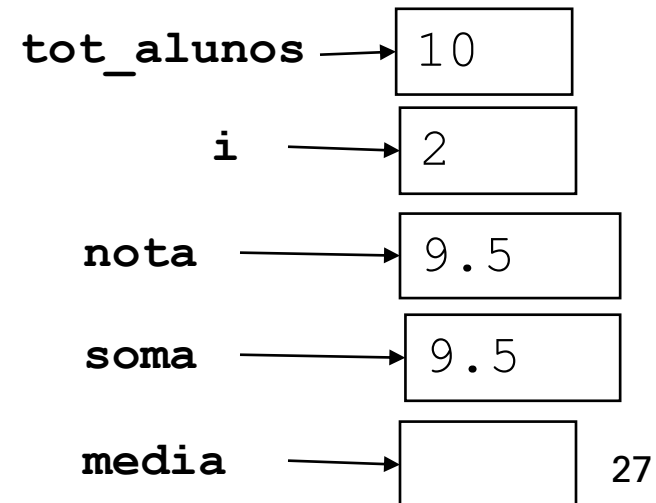
Comando while

- Ler a nota final de 10 alunos de uma turma e calcular e imprimir a média da turma.

```
tot_alunos = 10
i = 1
soma = 0
while (i <= tot_alunos): ←
    print('Digite a nota ', i, ': ')
    nota = float(input())
    soma = soma + nota
    i = i + 1
```

```
media = soma / tot_alunos
print('Media Turma:', round(media, 1))
```

- Voltamos ao teste do **while**
- O conteúdo de i é menor ou igual a 10?
- **SIM** (True), então comandos do laço devem ser executados novamente.



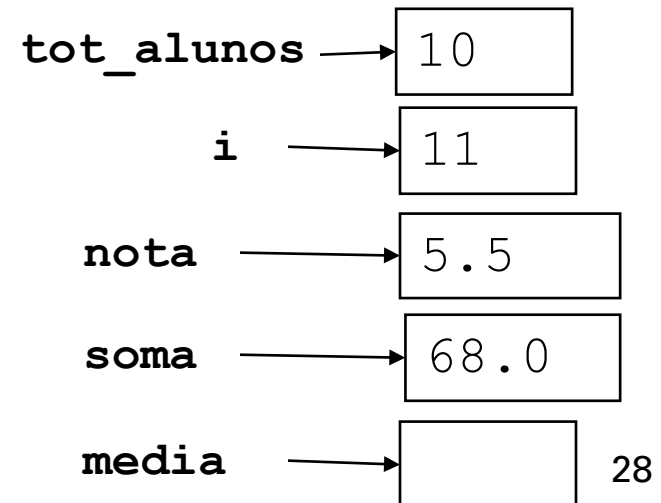
Comando while

- Ler a nota final de 10 alunos de uma turma e calcular e imprimir a média da turma.

```
tot_alunos = 10
i = 1
soma = 0
while (i <= tot_alunos): ←
    print('Digite a nota ', i, ': ')
    nota = float(input())
    soma = soma + nota
    i = i + 1
```

```
media = soma / tot_alunos
print('Media Turma:', round(media, 1))
```

- Tudo isso vai se repetir até **i** atingir o valor 11...
- Neste momento o teste vai dar **False** e o controle do programa irá para a 1ª linha após (fora) do bloco de código do **while**.



Comando while

- Ler a nota final de 10 alunos de uma turma e calcular e imprimir a média da turma.

```
tot_alunos = 10
```

```
i = 1
```

```
soma = 0
```

```
while (i <= tot_alunos):  
    print('Digite a nota ', i, ': ')  
    nota = float(input())  
    soma = soma + nota  
    i = i + 1
```

```
media = soma / tot_alunos  
print('Media Turma:', round(media, 1))
```

- Tudo isso vai se repetir até **i** atingir o valor 11...
- Neste momento o teste vai dar **False** e o controle do programa irá para a 1ª linha após (fora) do bloco de código do **while**.

tot_alunos	→	10
i	→	11
nota	→	5.5
soma	→	68.0
media	→	6.8

Comando while

- **Em resumo:**
- O comando **while** permite que um bloco de instruções sejam executadas repetidas vezes enquanto uma determinada condição não for atingida.
- As instruções que serão repetidas devem ser colocadas **indentadas** (alinhadas) em um bloco de código **abaixo da linha do while**.
- A linha do while termina com dois pontos ":"
- O teste da condição é feito no início (pode acontecer que que os comandos da repetição não sejam executados nenhuma vez).

while (condição):

C_1

C_2

...

C_n

Exercícios

- (1) Faça uma simulação em papel e indique qual será valor da variável **s** ao final da execução do programa (*não digite e nem rode o programa... Faça em papel !!!*)

```
x = 1
s = 3
while (x < 8) :
    s = s + 5
    x = x * 2

print (s)
```

- (2) Modifique o programa mostrado no slide 4, empregando o comando **while** para eliminar as linhas de código repetidas.
- (3) Analise e entenda o algoritmo apresentado no slide 17 e, em seguida, converta-o para um programa Python.