

Computação I - Python

Aula 7 - Estrutura de Repetição com Teste de Parada: while

Simulação de execução

Apresentado por: Anamaria Martins Moreira

Produção DCC-UFRJ

Metodologia de referência <https://doi.org/10.5753/wei.2016.9683>



funcao f, que ainda nao sabemos o que faz

$() \rightarrow \text{int}$

def f():

 i = 0

 j = 8

 while i < j:

 i = i + 1

 j = j - 1

 return i

funcao f, que ainda nao sabemos o que faz

$() \rightarrow \text{int}$

def f():

 i = 0 

 j = 8

 while i < j:

 i = i + 1

 j = j - 1

 return i

funcao f, que ainda nao sabemos o que faz

$() \rightarrow \text{int}$

def f():

 i ← 0

 j = 8

 while i < j:

 i = i + 1

 j = j - 1

 return i

iteração\variável	i
antes do while	0

funcao f, que ainda nao sabemos o que faz

$() \rightarrow \text{int}$

def f():

 i = 0

 j = 8 

 while i < j:

 i = i + 1

 j = j - 1

 return i


iteração\variável	i
antes do while	0

funcao f, que ainda nao sabemos o que faz

$() \rightarrow \text{int}$

def f():

 i = 0

 j = 8 

 while i < j:

 i = i + 1

 j = j - 1

 return i

iteração\variável	i	j
antes do while	0	8

funcao f, que ainda nao sabemos o que faz

$() \rightarrow \text{int}$

def f():

 i = 0

 j = 8

 while i < j:

 i = i + 1

 j = j - 1

 return i

iteração\variável	i	j	i<j
antes do while	0	8	0<8

funcao f, que ainda nao sabemos o que faz

$() \rightarrow \text{int}$

def f():

 i = 0

 j = 8

 while i < j:

 i = i + 1

 j = j - 1

 return i

iteração\variável	i	j	i<j
antes do while	0	8	True

funcao f, que ainda nao sabemos o que faz

() \rightarrow int

def f():

 i = 0

 j = 8

 while i < j:

 i

 j = j - 1

 return i

iteração\variável	i	j	i < j
antes do while	0	8	True
1a iteração			

funcao f, que ainda nao sabemos o que faz

() \rightarrow int

def f():

 i = 0

 j = 8

 while i < j:

 i

 j = j - 1

 return i

iteração\variável	i	j	i < j
antes do while	0	8	True
1a iteração	0+1		

funcao f, que ainda nao sabemos o que faz

() \rightarrow int

def f():

 i = 0

 j = 8

 while i < j:

 i

 j = j - 1

 return i

iteração\variável	i	j	i < j
antes do while	0	8	True
1a iteração	1		

funcao f, que ainda nao sabemos o que faz

() → int

def f():

 i = 0

 j = 8

 while i < j:

 i = i + 1

 j = j - 1

 return i

iteração\variável	i	j	i<j
antes do while	0	8	True
1a iteração	1		

funcao f, que ainda nao sabemos o que faz

() → int

def f():

 i = 0

 j = 8

 while i < j:

 i = i + 1

 j = j - 1

 return i

iteração\variável	i	j	i<j
antes do while	0	8	True
1a iteração	1	8-1	

funcao f, que ainda nao sabemos o que faz

() → int

def f():

 i = 0

 j = 8

 while i < j:

 i = i + 1

 j = j - 1

 return i

iteração\variável	i	j	i<j
antes do while	0	8	True
1a iteração	1	7	

funcao f, que ainda nao sabemos o que faz

() \rightarrow int

def f():

 i = 0

 j = 8

 while i < j

 i = i + 1

 j = j - 1

 return i

iteração\variável	i	j	i<j
antes do while	0	8	True
1a iteração	1	7	1<7

funcao f, que ainda nao sabemos o que faz

() \rightarrow int

def f():

 i = 0

 j = 8

 while i < j

 i = i + 1

 j = j - 1

 return i

iteração\variável	i	j	i<j
antes do while	0	8	True
1a iteração	1	7	True

funcao f, que ainda nao sabemos o que faz

() \rightarrow int

def f():

 i = 0

 j = 8

 while i < j:

 i ← i + 1

 j = j - 1

 return i

iteração\variável	i	j	i < j
antes do while	0	8	True
1a iteração	1	7	True
2a iteração			

funcao f, que ainda nao sabemos o que faz

() \rightarrow int

def f():

 i = 0

 j = 8

 while i < j:

 i ← i + 1

 j = j - 1

 return i

iteração\variável	i	j	i < j
antes do while	0	8	True
1a iteração	1	7	True
2a iteração	1+1		

funcao f, que ainda nao sabemos o que faz

() \rightarrow int

def f():

 i = 0

 j = 8

 while i < j:

 i ← i + 1

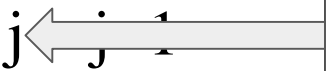
 j = j - 1

 return i

iteração\variável	i	j	i < j
antes do while	0	8	True
1a iteração	1	7	True
2a iteração	2		

```
# funcao f, que ainda nao sabemos o que faz
# () -> int
def f():
```

```
    i = 0
    j = 8
    while i<j:
        i = i + 1
        j = j - 1
    return i
```



iteração\variável	i	j	i<j
antes do while	0	8	True
1a iteração	1	7	True
2a iteração	2		

funcao f, que ainda nao sabemos o que faz

() → int

def f():

 i = 0

 j = 8

 while i < j:

 i = i + 1

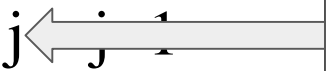
 j ← j - 1

 return i

iteração\variável	i	j	i<j
antes do while	0	8	True
1a iteração	1	7	True
2a iteração	2	7-1	

```
# funcao f, que ainda nao sabemos o que faz
# () -> int
def f():
```

```
    i = 0
    j = 8
    while i<j:
        i = i + 1
        j = j - 1
    return i
```



iteração\variável	i	j	i<j
antes do while	0	8	True
1a iteração	1	7	True
2a iteração	2	6	

funcao f, que ainda nao sabemos o que faz

$() \rightarrow \text{int}$

def f():

 i = 0

 j = 8

 while i < j:

 i = i + 1

 j = j - 1

 return i

iteração\variável	i	j	i<j
antes do while	0	8	True
1a iteração	1	7	True
2a iteração	2	6	

funcao f, que ainda nao sabemos o que faz

$() \rightarrow \text{int}$

def f():

 i = 0

 j = 8

 while i < j:

 i = i + 1

 j = j - 1

 return i

iteração\variável	i	j	i<j
antes do while	0	8	True
1a iteração	1	7	True
2a iteração	2	6	2<6

funcao f, que ainda nao sabemos o que faz

() → int

def f():

 i = 0

 j = 8

 while i < j:

 i ← i + 1

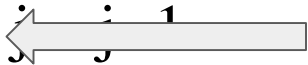
 j = j - 1

 return i

iteração\variável	i	j	i < j
antes do while	0	8	True
1a iteração	1	7	True
2a iteração	2	6	True
3a iteração			

```
# funcao f, que ainda nao sabemos o que faz
# () -> int
def f():
```

```
    i = 0
    j = 8
    while i<j:
        i = i + 1
    return i
```



iteração\variável	i	j	i<j
antes do while	0	8	True
1a iteração	1	7	True
2a iteração	2	6	True
3a iteração	3		

funcao f, que ainda nao sabemos o que faz

() → int

def f():

 i = 0

 j = 8

 while i < j:

 i = i + 1

 j = j - 1

 return i

iteração\variável	i	j	i<j
antes do while	0	8	True
1a iteração	1	7	True
2a iteração	2	6	True
3a iteração	3	5	

funcao f, que ainda nao sabemos o que faz

() \rightarrow int

def f():

 i = 0

 j = 8

 while i < j:

 i ← i + 1

 j = j - 1

 return i

iteração\variável	i	j	i<j
antes do while	0	8	True
1a iteração	1	7	True
2a iteração	2	6	True
3a iteração	3	5	True
4a iteração			

funcao f, que ainda nao sabemos o que faz

() → int

def f():

 i = 0

 j = 8

 while i < j:

 i = i + 1

 j = j - 1

 return i

iteração\variável	i	j	i<j
antes do while	0	8	True
1a iteração	1	7	True
2a iteração	2	6	True
3a iteração	3	5	True
4a iteração	4		

funcao f, que ainda nao sabemos o que faz

() \rightarrow int

def f():

 i = 0

 j = 8

 while i

 i = i + 1

 j = j - 1

 return i

iteração\variável	i	j	i<j
antes do while	0	8	True
1a iteração	1	7	True
2a iteração	2	6	True
3a iteração	3	5	True
4a iteração	4	4	

funcao f, que ainda nao sabemos o que faz

() → int

def f():

 i = 0

 j = 8

 while i < j

 i = i + 1

 j = j - 1

 return i

iteração\variável	i	j	i<j
antes do while	0	8	True
1a iteração	1	7	True
2a iteração	2	6	True
3a iteração	3	5	True
4a iteração	4	4	4<4

funcao f, que ainda nao sabemos o que faz

() \rightarrow int

def f():

 i = 0

 j = 8

 while i < j

 i = i + 1

 j = j - 1

 return i

iteração\variável	i	j	i<j
antes do while	0	8	True
1a iteração	1	7	True
2a iteração	2	6	True
3a iteração	3	5	True
4a iteração	4	4	False

funcao f, que ainda nao sabemos o que faz

() → int

def f():

 i = 0

 j = 8

 while i < j:

 i = i + 1

 j = j - 1

 return i

f() devolve 4

iteração\variável	i	j	i < j
antes do while	0	8	True
1a iteração	1	7	True
2a iteração	2	6	True
3a iteração	3	5	True
4a iteração	4	4	False

Agora podemos melhorar o nome e a descrição da função

funcao quatro, que retorna o inteiro 4

() → int

def quatro():

 i = 0

 j = 10

 while i < j:

 i = i + 1

 j = j - 1

 return i



iteração\variável	i	j	i<j
antes do while	0	8	True
1a iteração	1	7	True
2a iteração	2	6	True
3a iteração	3	5	True
4a iteração	4	4	False

quatro() devolve 4

funcao f, que ainda nao sabemos o que faz

(int, int) \rightarrow int

```
def f (i, j):
```

```
    while i<j:
```

```
        i = i + 1
```

```
        j = j -1
```

```
    return i
```

Simular a execução de (sempre importante testar situações bem variadas, em especial, zero e números negativos, no caso de inteiros)

f (0, 8)

f (1, 8)

f (-1, 5)

f (-5, -2)

f (10, 14)

f (1, 0)

f (-5, -1)

f devolve i quando $i \geq j$ e, quando $i < j$, a média de i e j, quando $i + j$ é par, arredondando para o inteiro superior, quando $i + j$ é ímpar.

(int, int) \rightarrow int

def f (i, j):

while $i < j$:

$i = i + 1$

$j = j - 1$

return i

f não possui uma descrição simples. Normalmente isso é sinal de que poderia ser repensada. Dificilmente queremos disponibilizar uma função cujo comportamento não pode ser descrito simplesmente.