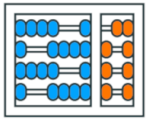


Instituto de Computação

UNIVERSIDADE ESTADUAL DE CAMPINAS

UNICAMP

MC102 - Aula 21

Exemplos sobre Recursão (parte 2)

Algoritmos e Programação de Computadores

Turmas

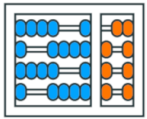
OVXZ

Prof. Lise R. R. Navarrete

lrommel@ic.unicamp.br

Terça-feira, 07 de junho de 2022

21:00h - 23:00h (CB06)



Instituto de Computação

UNIVERSIDADE ESTADUAL DE CAMPINAS

UNICAMP

MC102 – Algoritmos e Programação de Computadores

Turmas

OVXZ

https://ic.unicamp.br/~mc102/

Site da Coordenação de MC102

Aulas teóricas:

Terça-feira, 21:00h - 23:00h (CB06)

Quinta-feira, 19:00h - 21:00h (CB06)

Conteúdo

Exemplo 9

Cálculo do valor de k^n (sendo n um inteiro não negativo)

$$k^n = \begin{cases} 1, & \text{se } n = 0 \\ k \times k^{n-1}, & \text{caso contrário} \end{cases}$$

$$k^n = \begin{cases} 1, & \text{se } n = 0 \\ k^{n/2} \times k^{n/2}, & \text{se } n \text{ for positivo e par} \\ k \times k^{\lfloor n/2 \rfloor} \times k^{\lfloor n/2 \rfloor}, & \text{se } n \text{ for positivo e ímpar} \end{cases}$$

Implementação 1

$$k^n = \begin{cases} 1, & \text{se } n = 0 \\ k \times k^{n-1}, & \text{caso contrário} \end{cases}$$

Python 3.6
(known limitations)

```
1  def potencia1(k, n):
2      if n == 0:
3          return 1
4      else:
5          return k * potencia1(k, n - 1)
6
7  def potencia2(k, n):
8      if n == 0:
9          return 1
10     elif n % 2 == 0:
11         return potencia2(k, n//2) * potencia2(k, n//2)
12     else:
13         return k * potencia2(k, n//2) * potencia2(k, n//2)
14
15 print(potencia1(2,5))
```

Edit this code

line that just executed
next line to execute

<< First | < Prev | Next > | Last >>

Step 1 of 27

Print output (drag lower right corner to resize)

Frames Objects

$$k^n = \begin{cases} 1, & \text{se } n = 0 \\ k \times k^{n-1}, & \text{caso contrário} \end{cases}$$

https://pythontutor.com
https://tinyurl.com/5csdp9mx

Python 3.6
(known limitations)

```
1  def potencia1(k, n):
2      if n == 0:
3          return 1
4      else:
5          return k * potencia1(k, n - 1)
6
7  def potencia2(k, n):
8      if n == 0:
9          return 1
10     elif n % 2 == 0:
11         return potencia2(k, n//2) * potencia2(k, n//2)
12     else:
13         return k * potencia2(k, n//2) * potencia2(k, n//2)
14
15 print(potencia1(2,5))
```

Edit this code

line that just executed
next line to execute

<< First | < Prev | Next > | Last >>

Step 2 of 27

Print output (drag lower right corner to resize)

Frames Objects

Global frame
potencia1

function
potencia1(k, n)

$$k^n = \begin{cases} 1, & \text{se } n = 0 \\ k \times k^{n-1}, & \text{caso contrário} \end{cases}$$

https://pythontutor.com
https://tinyurl.com/5csdp9mx

Python 3.6
(known limitations)

```
1  def potencia1(k, n):
2      if n == 0:
3          return 1
4      else:
5          return k * potencia1(k, n - 1)
6
7  def potencia2(k, n):
8      if n == 0:
9          return 1
10     elif n % 2 == 0:
11         return potencia2(k, n//2) * potencia2(k, n//2)
12     else:
13         return k * potencia2(k, n//2) * potencia2(k, n//2)
14
15 print(potencia1(2,5))
```

Edit this code

line that just executed
next line to execute

<< First | < Prev | Next > | Last >>

Step 3 of 27

Print output (drag lower right corner to resize)

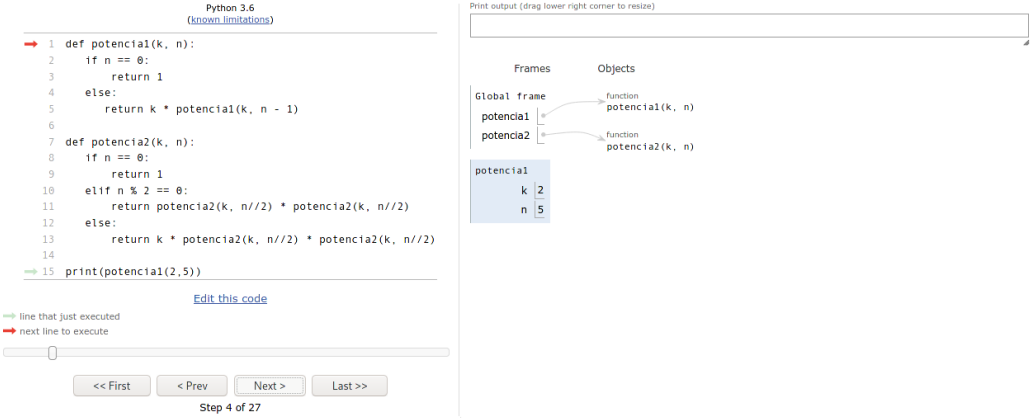
Frames Objects

Global frame
potencia1
potencia2

function potencia1(k, n)
function potencia2(k, n)

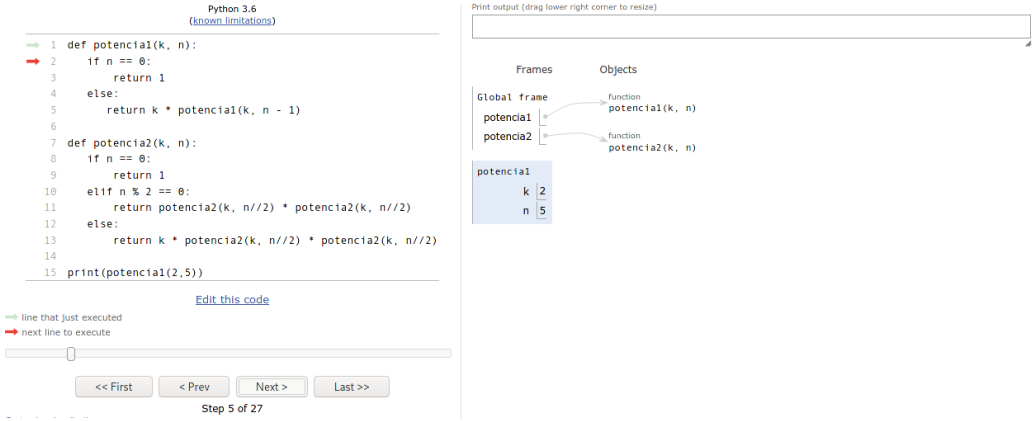
$$k^n = \begin{cases} 1, & \text{se } n = 0 \\ k \times k^{n-1}, & \text{caso contrário} \end{cases}$$

https://pythontutor.com
https://tinyurl.com/5csdp9mx



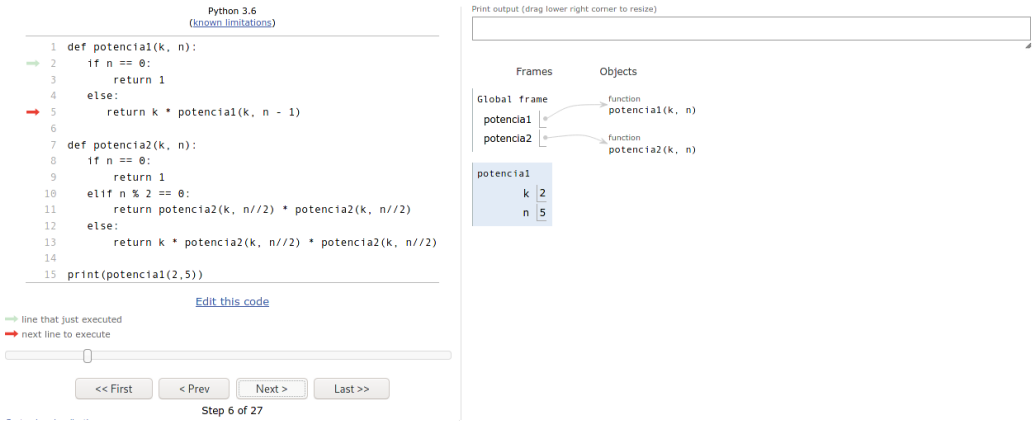
$$k^n = \begin{cases} 1, & \text{se } n = 0 \\ k \times k^{n-1}, & \text{caso contrário} \end{cases}$$

https://pythontutor.com
https://tinyurl.com/5csdp9mx



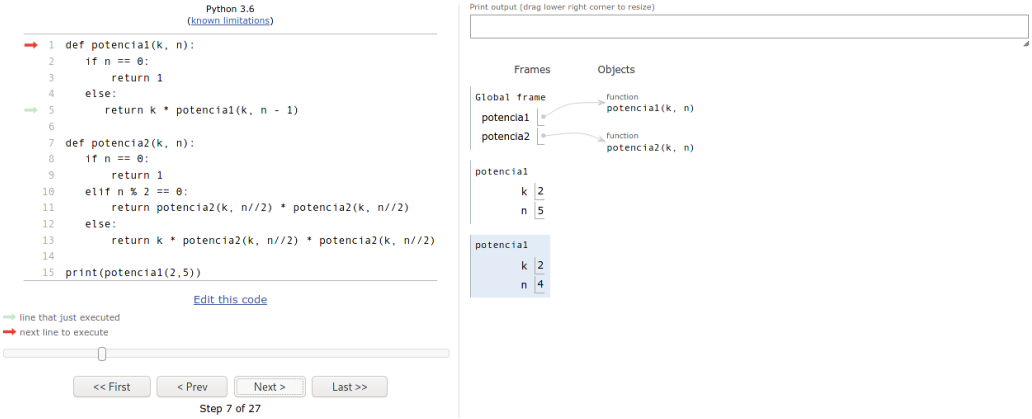
$$k^n = \begin{cases} 1, & \text{se } n = 0 \\ k \times k^{n-1}, & \text{caso contrário} \end{cases}$$

https://pythontutor.com
https://tinyurl.com/5csdp9mx



$$k^n = \begin{cases} 1, & \text{se } n = 0 \\ k \times k^{n-1}, & \text{caso contrário} \end{cases}$$

https://pythontutor.com
https://tinyurl.com/5csdp9mx



$$k^n = \begin{cases} 1, & \text{se } n = 0 \\ k \times k^{n-1}, & \text{caso contrário} \end{cases}$$

https://pythontutor.com

https://tinyurl.com/5csdp9mx

Python 3.6
(known limitations)

```
1  def potencia1(k, n):
2      if n == 0:
3          return 1
4      else:
5          return k * potencia1(k, n - 1)
6
7  def potencia2(k, n):
8      if n == 0:
9          return 1
10     elif n % 2 == 0:
11         return potencia2(k, n//2) * potencia2(k, n//2)
12     else:
13         return k * potencia2(k, n//2) * potencia2(k, n//2)
14
15 print(potencia1(2,5))
```

Edit this code

line that just executed
next line to execute

<< First | < Prev | Next > | Last >>

Step 8 of 27

Print output (drag lower right corner to resize)

Frames Objects

Global frame
potencia1
potencia2

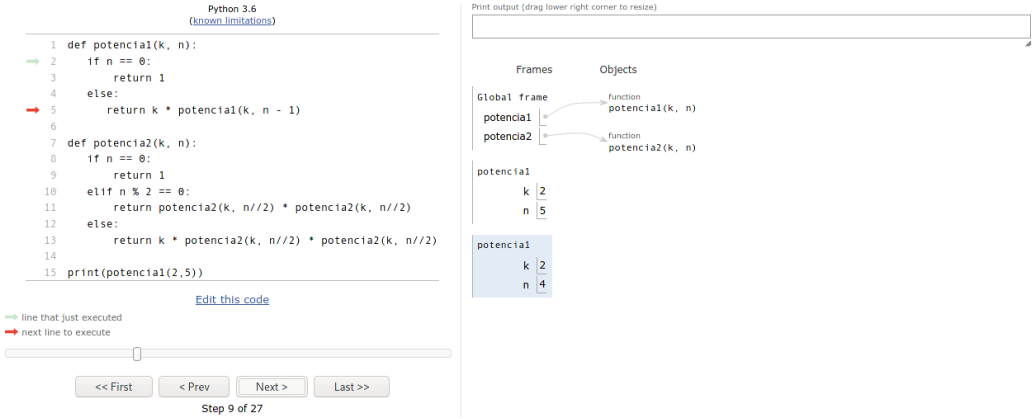
function potencia1(k, n)
function potencia2(k, n)

potencia1
k 2
n 5

potencia1
k 2
n 4

$$k^n = \begin{cases} 1, & \text{se } n = 0 \\ k \times k^{n-1}, & \text{caso contrário} \end{cases}$$

https://pythontutor.com
https://tinyurl.com/5csdp9mx



$$k^n = \begin{cases} 1, & \text{se } n = 0 \\ k \times k^{n-1}, & \text{caso contrário} \end{cases}$$

https://pythontutor.com

https://tinyurl.com/5csdp9mx

Python 3.6
(known limitations)

```
1  def potencia1(k, n):
2      if n == 0:
3          return 1
4      else:
5          return k * potencia1(k, n - 1)
6
7  def potencia2(k, n):
8      if n == 0:
9          return 1
10     elif n % 2 == 0:
11         return potencia2(k, n//2) * potencia2(k, n//2)
12     else:
13         return k * potencia2(k, n//2) * potencia2(k, n//2)
14
15 print(potencia1(2,5))
```

Edit this code

line that just executed
next line to execute

<< First | < Prev | Next > | Last >>

Step 10 of 27

Print output (drag lower right corner to resize)

Frames Objects

Global frame
potencia1
potencia2

function potencia1(k, n)
function potencia2(k, n)

potencia1
k 2
n 5

potencia1
k 2
n 4

potencia1
k 2
n 3

$$k^n = \begin{cases} 1, & \text{se } n = 0 \\ k \times k^{n-1}, & \text{caso contrário} \end{cases}$$

https://pythontutor.com
https://tinyurl.com/5csdp9mx

Python 3.6
(known limitations)

```
1  def potencia1(k, n):
2      if n == 0:
3          return 1
4      else:
5          return k * potencia1(k, n - 1)
6
7  def potencia2(k, n):
8      if n == 0:
9          return 1
10     elif n % 2 == 0:
11         return potencia2(k, n//2) * potencia2(k, n//2)
12     else:
13         return k * potencia2(k, n//2) * potencia2(k, n//2)
14
15 print(potencia1(2,5))
```

Edit this code

line that just executed
next line to execute

<< First | < Prev | Next > | Last >>

Step 11 of 27

Print output (drag lower right corner to resize)

Frames Objects

Global frame
potencia1
potencia2

function potencia1(k, n)
function potencia2(k, n)

potencia1
k 2
n 5

potencia1
k 2
n 4

potencia1
k 2
n 3

$$k^n = \begin{cases} 1, & \text{se } n = 0 \\ k \times k^{n-1}, & \text{caso contrário} \end{cases}$$

https://pythontutor.com
https://tinyurl.com/5csdp9mx

Python 3.6
(known limitations)

```
1  def potencia1(k, n):
2      if n == 0:
3          return 1
4      else:
5          return k * potencia1(k, n - 1)
6
7  def potencia2(k, n):
8      if n == 0:
9          return 1
10     elif n % 2 == 0:
11         return potencia2(k, n//2) * potencia2(k, n//2)
12     else:
13         return k * potencia2(k, n//2) * potencia2(k, n//2)
14
15 print(potencia1(2,5))
```

Edit this code

line that just executed
next line to execute

<< First | < Prev | Next > | Last >>

Step 12 of 27

Print output (drag lower right corner to resize)

Frames Objects

Global frame
potencia1
potencia2

function potencia1(k, n)
function potencia2(k, n)

potencia1
k 2
n 5

potencia1
k 2
n 4

potencia1
k 2
n 3

$$k^n = \begin{cases} 1, & \text{se } n = 0 \\ k \times k^{n-1}, & \text{caso contrário} \end{cases}$$

https://pythontutor.com
https://tinyurl.com/5csdp9mx

Python 3.6
(known limitations)

```
1  def potencia1(k, n):
2      if n == 0:
3          return 1
4      else:
5          return k * potencia1(k, n - 1)
6
7  def potencia2(k, n):
8      if n == 0:
9          return 1
10     elif n % 2 == 0:
11         return potencia2(k, n//2) * potencia2(k, n//2)
12     else:
13         return k * potencia2(k, n//2) * potencia2(k, n//2)
14
15 print(potencia1(2,5))
```

Edit this code

line that just executed
next line to execute

<< First | < Prev | Next > | Last >>

Step 13 of 27

Print output (drag lower right corner to resize)

Frames Objects

Global frame
potencia1
potencia2

function potencia1(k, n)
function potencia2(k, n)

potencia1
k 2
n 5

potencia1
k 2
n 4

potencia1
k 2
n 3

potencia1
k 2
n 2

$$k^n = \begin{cases} 1, & \text{se } n = 0 \\ k \times k^{n-1}, & \text{caso contrário} \end{cases}$$

https://pythontutor.com
https://tinyurl.com/5csdp9mx

Python 3.6
(known limitations)

```
1  def potencia1(k, n):
2      if n == 0:
3          return 1
4      else:
5          return k * potencia1(k, n - 1)
6
7  def potencia2(k, n):
8      if n == 0:
9          return 1
10     elif n % 2 == 0:
11         return potencia2(k, n//2) * potencia2(k, n//2)
12     else:
13         return k * potencia2(k, n//2) * potencia2(k, n//2)
14
15 print(potencia1(2,5))
```

Edit this code

line that just executed
next line to execute

<< First | < Prev | Next > | Last >>

Step 14 of 27

Print output (drag lower right corner to resize)

Frames Objects

Global frame
potencia1 → function potencia1(k, n)
potencia2 → function potencia2(k, n)

potencia1
k 2
n 5

potencia1
k 2
n 4

potencia1
k 2
n 3

potencia1
k 2
n 2

$$k^n = \begin{cases} 1, & \text{se } n = 0 \\ k \times k^{n-1}, & \text{caso contrário} \end{cases}$$

https://pythontutor.com
https://tinyurl.com/5csdp9mx

Python 3.6
(known limitations)

```
1  def potencia1(k, n):
2      if n == 0:
3          return 1
4      else:
5          return k * potencia1(k, n - 1)
6
7  def potencia2(k, n):
8      if n == 0:
9          return 1
10     elif n % 2 == 0:
11         return potencia2(k, n//2) * potencia2(k, n//2)
12     else:
13         return k * potencia2(k, n//2) * potencia2(k, n//2)
14
15 print(potencia1(2,5))
```

Edit this code

line that just executed
next line to execute

<< First | < Prev | Next > | Last >>

Step 15 of 27

Print output (drag lower right corner to resize)

Frames Objects

Global frame
potencia1
potencia2

function potencia1(k, n)
function potencia2(k, n)

potencia1
k 2
n 5

potencia1
k 2
n 4

potencia1
k 2
n 3

potencia1
k 2
n 2

$$k^n = \begin{cases} 1, & \text{se } n = 0 \\ k \times k^{n-1}, & \text{caso contrário} \end{cases}$$

https://pythontutor.com
https://tinyurl.com/5csdp9mx

Python 3.6
(known limitations)

```
1  def potencia1(k, n):
2      if n == 0:
3          return 1
4      else:
5          return k * potencia1(k, n - 1)
6
7  def potencia2(k, n):
8      if n == 0:
9          return 1
10     elif n % 2 == 0:
11         return potencia2(k, n//2) * potencia2(k, n//2)
12     else:
13         return k * potencia2(k, n//2) * potencia2(k, n//2)
14
15 print(potencia1(2,5))
```

Edit this code

line that just executed
next line to execute

<< First | < Prev | Next > | Last >>

Step 16 of 27

Print output (drag lower right corner to resize)

Frames Objects

Global frame
potencia1
potencia2

function
potencia1(k, n)

function
potencia2(k, n)

potencia1
k 2
n 5

potencia1
k 2
n 4

potencia1
k 2
n 3

potencia1
k 2
n 2

potencia1
k 2
n 1

$$k^n = \begin{cases} 1, & \text{se } n = 0 \\ k \times k^{n-1}, & \text{caso contrário} \end{cases}$$

https://pythontutor.com
https://tinyurl.com/5csdp9mx

Python 3.6
(known limitations)

```
1  def potencia1(k, n):
2      if n == 0:
3          return 1
4      else:
5          return k * potencia1(k, n - 1)
6
7  def potencia2(k, n):
8      if n == 0:
9          return 1
10     elif n % 2 == 0:
11         return potencia2(k, n//2) * potencia2(k, n//2)
12     else:
13         return k * potencia2(k, n//2) * potencia2(k, n//2)
14
15 print(potencia1(2,5))
```

Edit this code

line that just executed
next line to execute

<< First | < Prev | Next > | Last >>

Step 17 of 27

Print output (drag lower right corner to resize)

Frames Objects

Global frame
potencia1
potencia2

function
potencia1(k, n)

function
potencia2(k, n)

potencia1
k 2
n 5

potencia1
k 2
n 4

potencia1
k 2
n 3

potencia1
k 2
n 2

potencia1
k 2
n 1

$$k^n = \begin{cases} 1, & \text{se } n = 0 \\ k \times k^{n-1}, & \text{caso contrário} \end{cases}$$

https://pythontutor.com
https://tinyurl.com/5csdp9mx

Python 3.6
(known limitations)

```
 1  def potencia1(k, n):
 2      if n == 0:
 3          return 1
 4      else:
 5          return k * potencia1(k, n - 1)
 6
 7  def potencia2(k, n):
 8      if n == 0:
 9          return 1
10      elif n % 2 == 0:
11          return potencia2(k, n//2) * potencia2(k, n//2)
12      else:
13          return k * potencia2(k, n//2) * potencia2(k, n//2)
14
15  print(potencia1(2,5))
```

Edit this code

line that just executed
next line to execute

<< First | < Prev | Next > | Last >>

Step 18 of 27

Print output (drag lower right corner to resize)

Frames Objects

Global frame
potencia1
potencia2

function potencia1(k, n)
function potencia2(k, n)

potencia1
k 2
n 5

potencia1
k 2
n 4

potencia1
k 2
n 3

potencia1
k 2
n 2

potencia1
k 2
n 1

$$k^n = \begin{cases} 1, & \text{se } n = 0 \\ k \times k^{n-1}, & \text{caso contrário} \end{cases}$$

https://pythontutor.com
https://tinyurl.com/5csdp9mx

Python 3.6
(known limitations)

```
1  def potencia1(k, n):
2      if n == 0:
3          return 1
4      else:
5          return k * potencia1(k, n - 1)
6
7  def potencia2(k, n):
8      if n == 0:
9          return 1
10     elif n % 2 == 0:
11         return potencia2(k, n//2) * potencia2(k, n//2)
12     else:
13         return k * potencia2(k, n//2) * potencia2(k, n//2)
14
15 print(potencia1(2,5))
```

Edit this code

line that just executed
next line to execute

<< First | < Prev | Next > | Last >>

Step 19 of 27

Print output (drag lower right corner to resize)

Frames Objects

Global frame
potencia1 → function potencia1(k, n)
potencia2 → function potencia2(k, n)

potencia1
k 2
n 5

potencia1
k 2
n 4

potencia1
k 2
n 3

potencia1
k 2
n 2

potencia1
k 2
n 1

potencia1
k 2
n 0

$$k^n = \begin{cases} 1, & \text{se } n = 0 \\ k \times k^{n-1}, & \text{caso contrário} \end{cases}$$

https://pythontutor.com
https://tinyurl.com/5csdp9mx

Python 3.6
(known limitations)

```
1  def potencia1(k, n):
2      if n == 0:
3          return 1
4      else:
5          return k * potencia1(k, n - 1)
6
7  def potencia2(k, n):
8      if n == 0:
9          return 1
10     elif n % 2 == 0:
11         return potencia2(k, n//2) * potencia2(k, n//2)
12     else:
13         return k * potencia2(k, n//2) * potencia2(k, n//2)
14
15 print(potencia1(2,5))
```

Edit this code

line that just executed
next line to execute

<< First | < Prev | Next > | Last >>

Step 20 of 27

Print output (drag lower right corner to resize)

Frames Objects

Global frame
potencia1 → function potencia1(k, n)
potencia2 → function potencia2(k, n)

potencia1
k 2
n 5

potencia1
k 2
n 4

potencia1
k 2
n 3

potencia1
k 2
n 2

potencia1
k 2
n 1

potencia1
k 2
n 0

$$k^n = \begin{cases} 1, & \text{se } n = 0 \\ k \times k^{n-1}, & \text{caso contrário} \end{cases}$$

https://pythontutor.com
https://tinyurl.com/5csdp9mx

Python 3.6
(known limitations)

```
1  def potencia1(k, n):
2      if n == 0:
3          return 1
4      else:
5          return k * potencia1(k, n - 1)
6
7  def potencia2(k, n):
8      if n == 0:
9          return 1
10     elif n % 2 == 0:
11         return potencia2(k, n//2) * potencia2(k, n//2)
12     else:
13         return k * potencia2(k, n//2) * potencia2(k, n//2)
14
15 print(potencia1(2,5))
```

Edit this code

line that just executed
next line to execute

<< First | < Prev | Next > | Last >>

Step 21 of 27

Print output (drag lower right corner to resize)

Frames Objects

Global frame
potencia1 → function potencia1(k, n)
potencia2 → function potencia2(k, n)

potencia1
k 2
n 5

potencia1
k 2
n 4

potencia1
k 2
n 3

potencia1
k 2
n 2

potencia1
k 2
n 1

potencia1
k 2
n 0

$$k^n = \begin{cases} 1, & \text{se } n = 0 \\ k \times k^{n-1}, & \text{caso contrário} \end{cases}$$

https://pythontutor.com
https://tinyurl.com/5csdp9mx

Python 3.6
(known limitations)

```
1  def potencia1(k, n):
2      if n == 0:
3          return 1
4      else:
5          return k * potencia1(k, n - 1)
6
7  def potencia2(k, n):
8      if n == 0:
9          return 1
10     elif n % 2 == 0:
11         return potencia2(k, n//2) * potencia2(k, n//2)
12     else:
13         return k * potencia2(k, n//2) * potencia2(k, n//2)
14
15 print(potencia1(2,5))
```

Edit this code

line that just executed
next line to execute

<< First | < Prev | Next > | Last >>

Step 22 of 27

Print output (drag lower right corner to resize)

Frames Objects

Global frame
potencia1 → function potencia1(k, n)
potencia2 → function potencia2(k, n)

potencia1
k 2
n 5

potencia1
k 2
n 4

potencia1
k 2
n 3

potencia1
k 2
n 2

potencia1
k 2
n 1

potencia1
k 2
n 0
Return value 1

$$k^n = \begin{cases} 1, & \text{se } n = 0 \\ k \times k^{n-1}, & \text{caso contrário} \end{cases}$$

https://pythontutor.com
https://tinyurl.com/5csdp9mx

Python 3.6
(known limitations)

```
1  def potencia1(k, n):
2      if n == 0:
3          return 1
4      else:
5          return k * potencia1(k, n - 1)
6
7  def potencia2(k, n):
8      if n == 0:
9          return 1
10     elif n % 2 == 0:
11         return potencia2(k, n//2) * potencia2(k, n//2)
12     else:
13         return k * potencia2(k, n//2) * potencia2(k, n//2)
14
15 print(potencia1(2,5))
```

Edit this code

line that just executed
next line to execute

<< First | < Prev | Next > | Last >>

Step 23 of 27

Print output (drag lower right corner to resize)

Frames Objects

Global frame
potencia1
potencia2

function potencia1(k, n)
function potencia2(k, n)

potencia1
k 2
n 5

potencia1
k 2
n 4

potencia1
k 2
n 3

potencia1
k 2
n 2

potencia1
k 2
n 1
Return value 2

$$k^n = \begin{cases} 1, & \text{se } n = 0 \\ k \times k^{n-1}, & \text{caso contrário} \end{cases}$$

https://pythontutor.com
https://tinyurl.com/5csdp9mx

Python 3.6
(known limitations)

```
1  def potencia1(k, n):
2      if n == 0:
3          return 1
4      else:
5          return k * potencia1(k, n - 1)
6
7  def potencia2(k, n):
8      if n == 0:
9          return 1
10     elif n % 2 == 0:
11         return potencia2(k, n//2) * potencia2(k, n//2)
12     else:
13         return k * potencia2(k, n//2) * potencia2(k, n//2)
14
15 print(potencia1(2,5))
```

Edit this code

line that just executed
next line to execute

<< First | < Prev | Next > | Last >>

Step 24 of 27

Print output (drag lower right corner to resize)

Frames Objects

Global frame
potencia1
potencia2

function potencia1(k, n)
function potencia2(k, n)

potencia1
k 2
n 5

potencia1
k 2
n 4

potencia1
k 2
n 3

potencia1
k 2
n 2
Return value 4

$$k^n = \begin{cases} 1, & \text{se } n = 0 \\ k \times k^{n-1}, & \text{caso contrário} \end{cases}$$

https://pythontutor.com
https://tinyurl.com/5csdp9mx

Python 3.6
(known limitations)

```
1  def potencia1(k, n):
2      if n == 0:
3          return 1
4      else:
5          return k * potencia1(k, n - 1)
6
7  def potencia2(k, n):
8      if n == 0:
9          return 1
10     elif n % 2 == 0:
11         return potencia2(k, n//2) * potencia2(k, n//2)
12     else:
13         return k * potencia2(k, n//2) * potencia2(k, n//2)
14
15 print(potencia1(2,5))
```

Edit this code

line that just executed
next line to execute

<< First | < Prev | Next > | Last >>

Step 25 of 27

Print output (drag lower right corner to resize)

Frames Objects

Global frame
potencia1
potencia2

function potencia1(k, n)
function potencia2(k, n)

potencia1
k 2
n 5

potencia1
k 2
n 4

potencia1
k 2
n 3
Return value 8

$$k^n = \begin{cases} 1, & \text{se } n = 0 \\ k \times k^{n-1}, & \text{caso contrário} \end{cases}$$

https://pythontutor.com
https://tinyurl.com/5csdp9mx

Python 3.6
(known limitations)

```
1  def potencia1(k, n):
2      if n == 0:
3          return 1
4      else:
5          return k * potencia1(k, n - 1)
6
7  def potencia2(k, n):
8      if n == 0:
9          return 1
10     elif n % 2 == 0:
11         return potencia2(k, n//2) * potencia2(k, n//2)
12     else:
13         return k * potencia2(k, n//2) * potencia2(k, n//2)
14
15 print(potencia1(2,5))
```

Edit this code

line that just executed
next line to execute

<< First | < Prev | Next > | Last >>

Step 26 of 27

Print output (drag lower right corner to resize)

Frames Objects

Global frame
potencia1 → function potencia1(k, n)
potencia2 → function potencia2(k, n)

potencia1
k 2
n 5

potencia1
k 2
n 4
Return value 16

$$k^n = \begin{cases} 1, & \text{se } n = 0 \\ k \times k^{n-1}, & \text{caso contrário} \end{cases}$$

https://pythontutor.com
https://tinyurl.com/5csdp9mx

Python 3.6
(known limitations)

```
1  def potencia1(k, n):
2      if n == 0:
3          return 1
4      else:
5          return k * potencia1(k, n - 1)
6
7  def potencia2(k, n):
8      if n == 0:
9          return 1
10     elif n % 2 == 0:
11         return potencia2(k, n//2) * potencia2(k, n//2)
12     else:
13         return k * potencia2(k, n//2) * potencia2(k, n//2)
14
15 print(potencia1(2,5))
```

Edit this code

line that just executed
next line to execute

<< First | < Prev | Next > | Last >>

Step 27 of 27

Print output (drag lower right corner to resize)

Frames Objects

Global frame
potencia1
potencia2

function potencia1(k, n)
function potencia2(k, n)

potencia1
k 2
n 5
Return value 32

$$k^n = \begin{cases} 1, & \text{se } n = 0 \\ k \times k^{n-1}, & \text{caso contrário} \end{cases}$$

https://pythontutor.com
https://tinyurl.com/5csdp9mx

Python 3.6
(known limitations)

```
1  def potencia1(k, n):
2      if n == 0:
3          return 1
4      else:
5          return k * potencia1(k, n - 1)
6
7  def potencia2(k, n):
8      if n == 0:
9          return 1
10     elif n % 2 == 0:
11         return potencia2(k, n//2) * potencia2(k, n//2)
12     else:
13         return k * potencia2(k, n//2) * potencia2(k, n//2)
14
15 print(potencia1(2,5))
```

Edit this code

line that just executed
next line to execute

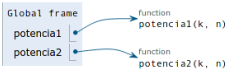
<< First | < Prev | Next > | Last >>

Done running (27 steps)

Print output (drag lower right corner to resize)

```
32
```

Frames Objects



$$k^n = \begin{cases} 1, & \text{se } n = 0 \\ k \times k^{n-1}, & \text{caso contrário} \end{cases}$$

https://pythontutor.com
https://tinyurl.com/5csdp9mx

Implementação 2

$$k^n = \begin{cases} 1, & \text{se } n = 0 \\ k^{n/2} \times k^{n/2}, & \text{se } n \text{ for positivo e par} \\ k \times k^{\lfloor n/2 \rfloor} \times k^{\lfloor n/2 \rfloor}, & \text{se } n \text{ for positivo e ímpar} \end{cases}$$

Python 3.6
(known limitations)

```
1  def potencia1(k, n):
2      if n == 0:
3          return 1
4      else:
5          return k * potencia1(k, n - 1)
6
7  def potencia2(k, n):
8      if n == 0:
9          return 1
10     elif n % 2 == 0:
11         return potencia2(k, n//2) * potencia2(k, n//2)
12     else:
13         return k * potencia2(k, n//2) * potencia2(k, n//2)
14
15 print(potencia2(2,5))
```

Edit this code

line that just executed
next line to execute

<< First | < Prev | Next > | Last >>

Step 1 of 70

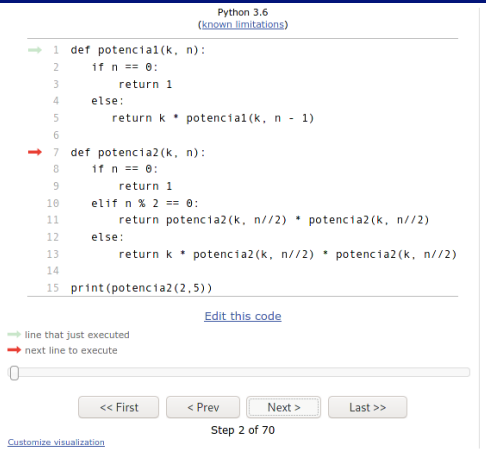
Customize visualization

Print output (drag lower right corner to resize)

Frames Objects

$$k^n = \begin{cases} 1, & \text{se } n = 0 \\ k^{n/2} \times k^{n/2}, & \text{se } n \text{ for positivo e par} \\ k \times k^{\lfloor n/2 \rfloor} \times k^{\lfloor n/2 \rfloor}, & \text{se } n \text{ for positivo e ímpar} \end{cases}$$

https://pythontutor.com
https://tinyurl.com/y6emx5yr



$$k^n = \begin{cases} 1, & \text{se } n = 0 \\ k^{n/2} \times k^{n/2}, & \text{se } n \text{ for positivo e par} \\ k \times k^{\lfloor n/2 \rfloor} \times k^{\lfloor n/2 \rfloor}, & \text{se } n \text{ for positivo e ímpar} \end{cases}$$

https://pythontutor.com

https://tinyurl.com/y6emx5yr

Python 3.6
(known limitations)

```
1  def potencia1(k, n):
2      if n == 0:
3          return 1
4      else:
5          return k * potencia1(k, n - 1)
6
7  def potencia2(k, n):
8      if n == 0:
9          return 1
10     elif n % 2 == 0:
11         return potencia2(k, n//2) * potencia2(k, n//2)
12     else:
13         return k * potencia2(k, n//2) * potencia2(k, n//2)
14
15 print(potencia2(2,5))
```

Edit this code

line that just executed
next line to execute

<< First | < Prev | Next > | Last >>

Step 3 of 70

Customize visualization

Print output (drag lower right corner to resize)

Frames Objects

Global frame
potencia1
potencia2

function
potencia1(k, n)

function
potencia2(k, n)

$$k^n = \begin{cases} 1, & \text{se } n = 0 \\ k^{n/2} \times k^{n/2}, & \text{se } n \text{ for positivo e par} \\ k \times k^{\lfloor n/2 \rfloor} \times k^{\lfloor n/2 \rfloor}, & \text{se } n \text{ for positivo e ímpar} \end{cases}$$

https://pythontutor.com
https://tinyurl.com/y6emx5yr

Python 3.6
(known limitations)

```
1  def potencia1(k, n):
2      if n == 0:
3          return 1
4      else:
5          return k * potencia1(k, n - 1)
6
7  def potencia2(k, n):
8      if n == 0:
9          return 1
10     elif n % 2 == 0:
11         return potencia2(k, n//2) * potencia2(k, n//2)
12     else:
13         return k * potencia2(k, n//2) * potencia2(k, n//2)
14
15 print(potencia2(2,5))
```

Edit this code

line that just executed
next line to execute

<< First | < Prev | Next > | Last >>

Step 4 of 70

Customize visualization

Print output (drag lower right corner to resize)

Frames Objects

Global frame
potencia1 → function potencia1(k, n)
potencia2 → function potencia2(k, n)

potencia2
k 2
n 5

$$k^n = \begin{cases} 1, & \text{se } n = 0 \\ k^{n/2} \times k^{n/2}, & \text{se } n \text{ for positivo e par} \\ k \times k^{\lfloor n/2 \rfloor} \times k^{\lfloor n/2 \rfloor}, & \text{se } n \text{ for positivo e ímpar} \end{cases}$$

https://pythontutor.com
https://tinyurl.com/y6emx5yr

Python 3.6
(known limitations)

```
1  def potencia1(k, n):
2      if n == 0:
3          return 1
4      else:
5          return k * potencia1(k, n - 1)
6
7  def potencia2(k, n):
8      if n == 0:
9          return 1
10     elif n % 2 == 0:
11         return potencia2(k, n//2) * potencia2(k, n//2)
12     else:
13         return k * potencia2(k, n//2) * potencia2(k, n//2)
14
15 print(potencia2(2,5))
```

Edit this code

line that just executed
next line to execute

<< First | < Prev | Next > | Last >>

Step 5 of 70

Customize visualization

Print output (drag lower right corner to resize)

Frames Objects

Global frame
potencia1 → function potencia1(k, n)
potencia2 → function potencia2(k, n)

potencia2
k 2
n 5

$$k^n = \begin{cases} 1, & \text{se } n = 0 \\ k^{n/2} \times k^{n/2}, & \text{se } n \text{ for positivo e par} \\ k \times k^{\lfloor n/2 \rfloor} \times k^{\lfloor n/2 \rfloor}, & \text{se } n \text{ for positivo e ímpar} \end{cases}$$

https://pythontutor.com
https://tinyurl.com/y6emx5yr

Python 3.6
(known limitations)

```
1  def potencia1(k, n):
2      if n == 0:
3          return 1
4      else:
5          return k * potencia1(k, n - 1)
6
7  def potencia2(k, n):
8      if n == 0:
9          return 1
10     elif n % 2 == 0:
11         return potencia2(k, n//2) * potencia2(k, n//2)
12     else:
13         return k * potencia2(k, n//2) * potencia2(k, n//2)
14
15 print(potencia2(2,5))
```

Edit this code

line that just executed
next line to execute

<< First | < Prev | Next > | Last >>

Step 6 of 70

Customize visualization

Print output (drag lower right corner to resize)

Frames Objects

Global frame
potencia1 → function potencia1(k, n)
potencia2 → function potencia2(k, n)

potencia2
k 2
n 5

$$k^n = \begin{cases} 1, & \text{se } n = 0 \\ k^{n/2} \times k^{n/2}, & \text{se } n \text{ for positivo e par} \\ k \times k^{\lfloor n/2 \rfloor} \times k^{\lfloor n/2 \rfloor}, & \text{se } n \text{ for positivo e ímpar} \end{cases}$$

https://pythontutor.com
https://tinyurl.com/y6emx5yr

Python 3.6
(known limitations)

```
1  def potencia1(k, n):
2      if n == 0:
3          return 1
4      else:
5          return k * potencia1(k, n - 1)
6
7  def potencia2(k, n):
8      if n == 0:
9          return 1
10     elif n % 2 == 0:
11         return potencia2(k, n//2) * potencia2(k, n//2)
12     else:
13         return k * potencia2(k, n//2) * potencia2(k, n//2)
14
15 print(potencia2(2,5))
```

Edit this code

line that just executed
next line to execute

<< First | < Prev | Next > | Last >>

Step 7 of 70

Customize visualization

Print output (drag lower right corner to resize)

Frames Objects

Global frame
potencia1
potencia2

function
potencia1(k, n)

function
potencia2(k, n)

potencia2
k 2
n 5

$$k^n = \begin{cases} 1, & \text{se } n = 0 \\ k^{n/2} \times k^{n/2}, & \text{se } n \text{ for positivo e par} \\ k \times k^{\lfloor n/2 \rfloor} \times k^{\lfloor n/2 \rfloor}, & \text{se } n \text{ for positivo e ímpar} \end{cases}$$

https://pythontutor.com
https://tinyurl.com/y6emx5yr

Python 3.6
(known limitations)

```
1  def potencia1(k, n):
2      if n == 0:
3          return 1
4      else:
5          return k * potencia1(k, n - 1)
6
7  def potencia2(k, n):
8      if n == 0:
9          return 1
10     elif n % 2 == 0:
11         return potencia2(k, n//2) * potencia2(k, n//2)
12     else:
13         return k * potencia2(k, n//2) * potencia2(k, n//2)
14
15 print(potencia2(2,5))
```

Edit this code

line that just executed
next line to execute

<< First | < Prev | Next > | Last >>

Step 8 of 70

Customize visualization

Print output (drag lower right corner to resize)

Frames Objects

Global frame
potencia1
potencia2

function
potencia1(k, n)

function
potencia2(k, n)

potencia2
k 2
n 5

potencia2
k 2
n 2

$$k^n = \begin{cases} 1, & \text{se } n = 0 \\ k^{n/2} \times k^{n/2}, & \text{se } n \text{ for positivo e par} \\ k \times k^{\lfloor n/2 \rfloor} \times k^{\lfloor n/2 \rfloor}, & \text{se } n \text{ for positivo e ímpar} \end{cases}$$

https://pythontutor.com
https://tinyurl.com/y6emx5yr

Python 3.6
(known limitations)

```
1  def potencia1(k, n):
2      if n == 0:
3          return 1
4      else:
5          return k * potencia1(k, n - 1)
6
7  def potencia2(k, n):
8      if n == 0:
9          return 1
10     elif n % 2 == 0:
11         return potencia2(k, n//2) * potencia2(k, n//2)
12     else:
13         return k * potencia2(k, n//2) * potencia2(k, n//2)
14
15 print(potencia2(2,5))
```

Edit this code

line that just executed
next line to execute

<< First | < Prev | Next > | Last >>

Step 9 of 70

Customize visualization

Print output (drag lower right corner to resize)

Frames Objects

Global frame
potencia1
potencia2

function
potencia1(k, n)

function
potencia2(k, n)

potencia2
k 2
n 5

potencia2
k 2
n 2

$$k^n = \begin{cases} 1, & \text{se } n = 0 \\ k^{n/2} \times k^{n/2}, & \text{se } n \text{ for positivo e par} \\ k \times k^{\lfloor n/2 \rfloor} \times k^{\lfloor n/2 \rfloor}, & \text{se } n \text{ for positivo e ímpar} \end{cases}$$

https://pythontutor.com
https://tinyurl.com/y6emx5yr

Python 3.6
(known limitations)

```
1  def potencia1(k, n):
2      if n == 0:
3          return 1
4      else:
5          return k * potencia1(k, n - 1)
6
7  def potencia2(k, n):
8      if n == 0:
9          return 1
10     elif n % 2 == 0:
11         return potencia2(k, n//2) * potencia2(k, n//2)
12     else:
13         return k * potencia2(k, n//2) * potencia2(k, n//2)
14
15 print(potencia2(2,5))
```

Edit this code

line that just executed
next line to execute

<< First | < Prev | Next > | Last >>

Step 10 of 70

Customize visualization

Print output (drag lower right corner to resize)

Frames Objects

Global frame
potencia1 → function potencia1(k, n)
potencia2 → function potencia2(k, n)

potencia2
k 2
n 5

potencia2
k 2
n 2

$$k^n = \begin{cases} 1, & \text{se } n = 0 \\ k^{n/2} \times k^{n/2}, & \text{se } n \text{ for positivo e par} \\ k \times k^{\lfloor n/2 \rfloor} \times k^{\lfloor n/2 \rfloor}, & \text{se } n \text{ for positivo e ímpar} \end{cases}$$

https://pythontutor.com
https://tinyurl.com/y6emx5yr

Python 3.6
(known limitations)

```
1  def potencia1(k, n):
2      if n == 0:
3          return 1
4      else:
5          return k * potencia1(k, n - 1)
6
7  def potencia2(k, n):
8      if n == 0:
9          return 1
10     elif n % 2 == 0:
11         return potencia2(k, n//2) * potencia2(k, n//2)
12     else:
13         return k * potencia2(k, n//2) * potencia2(k, n//2)
14
15 print(potencia2(2,5))
```

Edit this code

line that just executed
next line to execute

<< First | < Prev | Next > | Last >>

Step 11 of 70

Customize visualization

Print output (drag lower right corner to resize)

Frames Objects

Global frame
potencia1 → function potencia1(k, n)
potencia2 → function potencia2(k, n)

potencia2
k 2
n 5

potencia2
k 2
n 2

$$k^n = \begin{cases} 1, & \text{se } n = 0 \\ k^{n/2} \times k^{n/2}, & \text{se } n \text{ for positivo e par} \\ k \times k^{\lfloor n/2 \rfloor} \times k^{\lfloor n/2 \rfloor}, & \text{se } n \text{ for positivo e ímpar} \end{cases}$$

https://pythontutor.com
https://tinyurl.com/y6emx5yr

Python 3.6
(known limitations)

```
1  def potencia1(k, n):
2      if n == 0:
3          return 1
4      else:
5          return k * potencia1(k, n - 1)
6
7  def potencia2(k, n):
8      if n == 0:
9          return 1
10     elif n % 2 == 0:
11         return potencia2(k, n//2) * potencia2(k, n//2)
12     else:
13         return k * potencia2(k, n//2) * potencia2(k, n//2)
14
15 print(potencia2(2,5))
```

Edit this code

line that just executed
next line to execute

<< First | < Prev | Next > | Last >>

Step 12 of 70

Customize visualization

Print output (drag lower right corner to resize)

Frames Objects

Global frame
potencia1
potencia2

function
potencia1(k, n)

function
potencia2(k, n)

potencia2
k 2
n 5

potencia2
k 2
n 2

potencia2
k 2
n 1

$$k^n = \begin{cases} 1, & \text{se } n = 0 \\ k^{n/2} \times k^{n/2}, & \text{se } n \text{ for positivo e par} \\ k \times k^{\lfloor n/2 \rfloor} \times k^{\lfloor n/2 \rfloor}, & \text{se } n \text{ for positivo e ímpar} \end{cases}$$

https://pythontutor.com
https://tinyurl.com/y6emx5yr

Python 3.6
(known limitations)

```
1  def potencia1(k, n):
2      if n == 0:
3          return 1
4      else:
5          return k * potencia1(k, n - 1)
6
7  def potencia2(k, n):
8      if n == 0:
9          return 1
10     elif n % 2 == 0:
11         return potencia2(k, n//2) * potencia2(k, n//2)
12     else:
13         return k * potencia2(k, n//2) * potencia2(k, n//2)
14
15 print(potencia2(2,5))
```

Edit this code

line that just executed
next line to execute

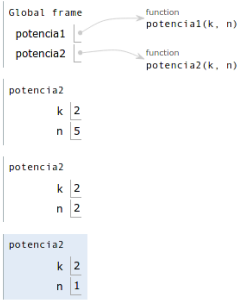
<< First | < Prev | Next > | Last >>

Step 13 of 70

Customize visualization

Print output (drag lower right corner to resize)

Frames Objects



$$k^n = \begin{cases} 1, & \text{se } n = 0 \\ k^{n/2} \times k^{n/2}, & \text{se } n \text{ for positivo e par} \\ k \times k^{\lfloor n/2 \rfloor} \times k^{\lfloor n/2 \rfloor}, & \text{se } n \text{ for positivo e ímpar} \end{cases}$$

https://pythontutor.com
https://tinyurl.com/y6emx5yr

Python 3.6
(known limitations)

```
1  def potencia1(k, n):
2      if n == 0:
3          return 1
4      else:
5          return k * potencia1(k, n - 1)
6
7  def potencia2(k, n):
8      if n == 0:
9          return 1
10     elif n % 2 == 0:
11         return potencia2(k, n//2) * potencia2(k, n//2)
12     else:
13         return k * potencia2(k, n//2) * potencia2(k, n//2)
14
15 print(potencia2(2,5))
```

Edit this code

line that just executed
next line to execute

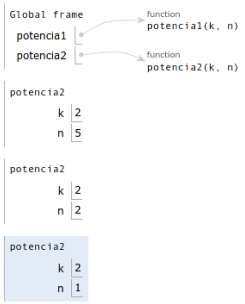
<< First | < Prev | Next > | Last >>

Step 14 of 70

Customize visualization

Print output (drag lower right corner to resize)

Frames Objects



$$k^n = \begin{cases} 1, & \text{se } n = 0 \\ k^{n/2} \times k^{n/2}, & \text{se } n \text{ for positivo e par} \\ k \times k^{\lfloor n/2 \rfloor} \times k^{\lfloor n/2 \rfloor}, & \text{se } n \text{ for positivo e ímpar} \end{cases}$$

https://pythontutor.com
https://tinyurl.com/y6emx5yr

Python 3.6
(known limitations)

```
1  def potencia1(k, n):
2      if n == 0:
3          return 1
4      else:
5          return k * potencia1(k, n - 1)
6
7  def potencia2(k, n):
8      if n == 0:
9          return 1
10     elif n % 2 == 0:
11         return potencia2(k, n//2) * potencia2(k, n//2)
12     else:
13         return k * potencia2(k, n//2) * potencia2(k, n//2)
14
15 print(potencia2(2,5))
```

Edit this code

line that just executed
next line to execute

<< First | < Prev | Next > | Last >>

Step 15 of 70

Customize visualization

Print output (drag lower right corner to resize)

Frames Objects

Global frame
potencia1
potencia2

function
potencia1(k, n)

function
potencia2(k, n)

potencia2
k 2
n 5

potencia2
k 2
n 2

potencia2
k 2
n 1

$$k^n = \begin{cases} 1, & \text{se } n = 0 \\ k^{n/2} \times k^{n/2}, & \text{se } n \text{ for positivo e par} \\ k \times k^{\lfloor n/2 \rfloor} \times k^{\lfloor n/2 \rfloor}, & \text{se } n \text{ for positivo e ímpar} \end{cases}$$

https://pythontutor.com
https://tinyurl.com/y6emx5yr

Python 3.6
(known limitations)

```
1  def potencia1(k, n):
2      if n == 0:
3          return 1
4      else:
5          return k * potencia1(k, n - 1)
6
7  def potencia2(k, n):
8      if n == 0:
9          return 1
10     elif n % 2 == 0:
11         return potencia2(k, n//2) * potencia2(k, n//2)
12     else:
13         return k * potencia2(k, n//2) * potencia2(k, n//2)
14
15 print(potencia2(2,5))
```

Edit this code

line that just executed
next line to execute

<< First | < Prev | Next > | Last >>

Step 16 of 70

Customize visualization

Print output (drag lower right corner to resize)

Frames Objects

Global frame
potencia1
potencia2

function
potencia1(k, n)

function
potencia2(k, n)

potencia2
k 2
n 5

potencia2
k 2
n 2

potencia2
k 2
n 1

potencia2
k 2
n 0

$$k^n = \begin{cases} 1, & \text{se } n = 0 \\ k^{n/2} \times k^{n/2}, & \text{se } n \text{ for positivo e par} \\ k \times k^{\lfloor n/2 \rfloor} \times k^{\lfloor n/2 \rfloor}, & \text{se } n \text{ for positivo e ímpar} \end{cases}$$

https://pythontutor.com
https://tinyurl.com/y6emx5yr

Python 3.6
(known limitations)

```
1  def potencia1(k, n):
2      if n == 0:
3          return 1
4      else:
5          return k * potencia1(k, n - 1)
6
7  def potencia2(k, n):
8      if n == 0:
9          return 1
10     elif n % 2 == 0:
11         return potencia2(k, n//2) * potencia2(k, n//2)
12     else:
13         return k * potencia2(k, n//2) * potencia2(k, n//2)
14
15 print(potencia2(2,5))
```

Edit this code

line that just executed
next line to execute

<< First | < Prev | Next > | Last >>

Step 17 of 70

Customize visualization

Print output (drag lower right corner to resize)

Frames Objects

Global frame
potencia1 → function potencia1(k, n)
potencia2 → function potencia2(k, n)

potencia2
k 2
n 5

potencia2
k 2
n 2

potencia2
k 2
n 1

potencia2
k 2
n 0

$$k^n = \begin{cases} 1, & \text{se } n = 0 \\ k^{n/2} \times k^{n/2}, & \text{se } n \text{ for positivo e par} \\ k \times k^{\lfloor n/2 \rfloor} \times k^{\lfloor n/2 \rfloor}, & \text{se } n \text{ for positivo e ímpar} \end{cases}$$

https://pythontutor.com
https://tinyurl.com/y6emx5yr

Python 3.6
(known limitations)

```
1  def potencia1(k, n):
2      if n == 0:
3          return 1
4      else:
5          return k * potencia1(k, n - 1)
6
7  def potencia2(k, n):
8      if n == 0:
9          return 1
10     elif n % 2 == 0:
11         return potencia2(k, n//2) * potencia2(k, n//2)
12     else:
13         return k * potencia2(k, n//2) * potencia2(k, n//2)
14
15 print(potencia2(2,5))
```

Edit this code

line that just executed
next line to execute

<< First | < Prev | Next > | Last >>

Step 18 of 70

Customize visualization

Print output (drag lower right corner to resize)

Frames Objects

Global frame
potencia1
potencia2

function
potencia1(k, n)

function
potencia2(k, n)

potencia2
k 2
n 5

potencia2
k 2
n 2

potencia2
k 2
n 1

potencia2
k 2
n 0

$$k^n = \begin{cases} 1, & \text{se } n = 0 \\ k^{n/2} \times k^{n/2}, & \text{se } n \text{ for positivo e par} \\ k \times k^{\lfloor n/2 \rfloor} \times k^{\lfloor n/2 \rfloor}, & \text{se } n \text{ for positivo e ímpar} \end{cases}$$

https://pythontutor.com
https://tinyurl.com/y6emx5yr

Python 3.6
(known limitations)

```
1  def potencia1(k, n):
2      if n == 0:
3          return 1
4      else:
5          return k * potencia1(k, n - 1)
6
7  def potencia2(k, n):
8      if n == 0:
9          return 1
10     elif n % 2 == 0:
11         return potencia2(k, n//2) * potencia2(k, n//2)
12     else:
13         return k * potencia2(k, n//2) * potencia2(k, n//2)
14
15 print(potencia2(2,5))
```

Edit this code

line that just executed
next line to execute

<< First | < Prev | Next > | Last >>

Step 19 of 70

Customize visualization

Print output (drag lower right corner to resize)

Frames Objects

Global frame
potencia1
potencia2

function
potencia1(k, n)

function
potencia2(k, n)

potencia2
k 2
n 5

potencia2
k 2
n 2

potencia2
k 2
n 1

potencia2
k 2
n 0
Return value 1

$$k^n = \begin{cases} 1, & \text{se } n = 0 \\ k^{n/2} \times k^{n/2}, & \text{se } n \text{ for positivo e par} \\ k \times k^{\lfloor n/2 \rfloor} \times k^{\lfloor n/2 \rfloor}, & \text{se } n \text{ for positivo e ímpar} \end{cases}$$

https://pythontutor.com
https://tinyurl.com/y6emx5yr

Python 3.6
(known limitations)

```
 1  def potencia1(k, n):
 2      if n == 0:
 3          return 1
 4      else:
 5          return k * potencia1(k, n - 1)
 6
 7  def potencia2(k, n):
 8      if n == 0:
 9          return 1
10      elif n % 2 == 0:
11          return potencia2(k, n//2) * potencia2(k, n//2)
12      else:
13          return k * potencia2(k, n//2) * potencia2(k, n//2)
14
15  print(potencia2(2,5))
```

Edit this code

line that just executed
next line to execute

<< First | < Prev | Next > | Last >>

Step 20 of 70

Customize visualization

Print output (drag lower right corner to resize)

Frames Objects

Global frame
potencia1
potencia2

function
potencia1(k, n)

function
potencia2(k, n)

potencia2
k 2
n 5

potencia2
k 2
n 2

potencia2
k 2
n 1

potencia2
k 2
n 0

$$k^n = \begin{cases} 1, & \text{se } n = 0 \\ k^{n/2} \times k^{n/2}, & \text{se } n \text{ for positivo e par} \\ k \times k^{\lfloor n/2 \rfloor} \times k^{\lfloor n/2 \rfloor}, & \text{se } n \text{ for positivo e ímpar} \end{cases}$$

https://pythontutor.com
https://tinyurl.com/y6emx5yr

Python 3.6
(known limitations)

```
1  def potencia1(k, n):
2      if n == 0:
3          return 1
4      else:
5          return k * potencia1(k, n - 1)
6
7  def potencia2(k, n):
8      if n == 0:
9          return 1
10     elif n % 2 == 0:
11         return potencia2(k, n//2) * potencia2(k, n//2)
12     else:
13         return k * potencia2(k, n//2) * potencia2(k, n//2)
14
15 print(potencia2(2,5))
```

Edit this code

line that just executed
next line to execute

<< First | < Prev | Next > | Last >>

Step 21 of 70

Customize visualization

Print output (drag lower right corner to resize)

Frames Objects

Global frame
potencia1
potencia2

function potencia1(k, n)
function potencia2(k, n)

potencia2
k 2
n 5

potencia2
k 2
n 2

potencia2
k 2
n 1

potencia2
k 2
n 0

$$k^n = \begin{cases} 1, & \text{se } n = 0 \\ k^{n/2} \times k^{n/2}, & \text{se } n \text{ for positivo e par} \\ k \times k^{\lfloor n/2 \rfloor} \times k^{\lfloor n/2 \rfloor}, & \text{se } n \text{ for positivo e ímpar} \end{cases}$$

https://pythontutor.com
https://tinyurl.com/y6emx5yr

Python 3.6
(known limitations)

```
1  def potencia1(k, n):
2      if n == 0:
3          return 1
4      else:
5          return k * potencia1(k, n - 1)
6
7  def potencia2(k, n):
8      if n == 0:
9          return 1
10     elif n % 2 == 0:
11         return potencia2(k, n//2) * potencia2(k, n//2)
12     else:
13         return k * potencia2(k, n//2) * potencia2(k, n//2)
14
15 print(potencia2(2,5))
```

Edit this code

line that just executed
next line to execute

<< First | < Prev | Next > | Last >>

Step 22 of 70

Customize visualization

Print output (drag lower right corner to resize)

Frames Objects

Global frame
potencia1
potencia2

function potencia1(k, n)
function potencia2(k, n)

potencia2
k 2
n 5

potencia2
k 2
n 2

potencia2
k 2
n 1

potencia2
k 2
n 0

$$k^n = \begin{cases} 1, & \text{se } n = 0 \\ k^{n/2} \times k^{n/2}, & \text{se } n \text{ for positivo e par} \\ k \times k^{\lfloor n/2 \rfloor} \times k^{\lfloor n/2 \rfloor}, & \text{se } n \text{ for positivo e ímpar} \end{cases}$$

https://pythontutor.com
https://tinyurl.com/y6emx5yr

Python 3.6
(known limitations)

```
1  def potencia1(k, n):
2      if n == 0:
3          return 1
4      else:
5          return k * potencia1(k, n - 1)
6
7  def potencia2(k, n):
8      if n == 0:
9          return 1
10     elif n % 2 == 0:
11         return potencia2(k, n//2) * potencia2(k, n//2)
12     else:
13         return k * potencia2(k, n//2) * potencia2(k, n//2)
14
15 print(potencia2(2,5))
```

Edit this code

line that just executed
next line to execute

<< First | < Prev | Next > | Last >>

Step 23 of 70

Customize visualization

Print output (drag lower right corner to resize)

Frames Objects

Global frame
potencia1
potencia2

function
potencia1(k, n)

function
potencia2(k, n)

potencia2
k 2
n 5

potencia2
k 2
n 2

potencia2
k 2
n 1

potencia2
k 2
n 0
Return value 1

$$k^n = \begin{cases} 1, & \text{se } n = 0 \\ k^{n/2} \times k^{n/2}, & \text{se } n \text{ for positivo e par} \\ k \times k^{\lfloor n/2 \rfloor} \times k^{\lfloor n/2 \rfloor}, & \text{se } n \text{ for positivo e ímpar} \end{cases}$$

https://pythontutor.com
https://tinyurl.com/y6emx5yr

Python 3.6
(known limitations)

```
1  def potencia1(k, n):
2      if n == 0:
3          return 1
4      else:
5          return k * potencia1(k, n - 1)
6
7  def potencia2(k, n):
8      if n == 0:
9          return 1
10     elif n % 2 == 0:
11         return potencia2(k, n//2) * potencia2(k, n//2)
12     else:
13         return k * potencia2(k, n//2) * potencia2(k, n//2)
14
15 print(potencia2(2,5))
```

Edit this code

line that just executed
next line to execute

<< First | < Prev | Next > | Last >>

Step 24 of 70

Customize visualization

Print output (drag lower right corner to resize)

Frames Objects

Global frame
potencia1
potencia2

function
potencia1(k, n)

function
potencia2(k, n)

potencia2
k 2
n 5

potencia2
k 2
n 2

potencia2
k 2
n 1
Return value 2

$$k^n = \begin{cases} 1, & \text{se } n = 0 \\ k^{n/2} \times k^{n/2}, & \text{se } n \text{ for positivo e par} \\ k \times k^{\lfloor n/2 \rfloor} \times k^{\lfloor n/2 \rfloor}, & \text{se } n \text{ for positivo e ímpar} \end{cases}$$

https://pythontutor.com
https://tinyurl.com/y6emx5yr

Python 3.6
(known limitations)

```
 1  def potencia1(k, n):
 2      if n == 0:
 3          return 1
 4      else:
 5          return k * potencia1(k, n - 1)
 6
 7  def potencia2(k, n):
 8      if n == 0:
 9          return 1
10      elif n % 2 == 0:
11          return potencia2(k, n//2) * potencia2(k, n//2)
12      else:
13          return k * potencia2(k, n//2) * potencia2(k, n//2)
14
15  print(potencia2(2,5))
```

Edit this code

line that just executed
next line to execute

<< First | < Prev | Next > | Last >>

Step 25 of 70

Customize visualization

Print output (drag lower right corner to resize)

Frames Objects

Global frame
potencia1
potencia2

function
potencia1(k, n)

function
potencia2(k, n)

potencia2
k 2
n 5

potencia2
k 2
n 2

potencia2
k 2
n 1

$$k^n = \begin{cases} 1, & \text{se } n = 0 \\ k^{n/2} \times k^{n/2}, & \text{se } n \text{ for positivo e par} \\ k \times k^{\lfloor n/2 \rfloor} \times k^{\lfloor n/2 \rfloor}, & \text{se } n \text{ for positivo e ímpar} \end{cases}$$

https://pythontutor.com
https://tinyurl.com/y6emx5yr

Python 3.6
(known limitations)

```
1  def potencia1(k, n):
2      if n == 0:
3          return 1
4      else:
5          return k * potencia1(k, n - 1)
6
7  def potencia2(k, n):
8      if n == 0:
9          return 1
10     elif n % 2 == 0:
11         return potencia2(k, n//2) * potencia2(k, n//2)
12     else:
13         return k * potencia2(k, n//2) * potencia2(k, n//2)
14
15 print(potencia2(2,5))
```

Edit this code

line that just executed
next line to execute

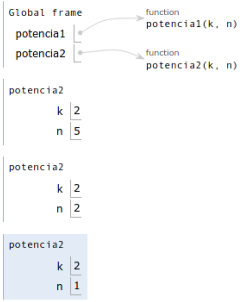
<< First | < Prev | Next > | Last >>

Step 26 of 70

Customize visualization

Print output (drag lower right corner to resize)

Frames Objects



$$k^n = \begin{cases} 1, & \text{se } n = 0 \\ k^{n/2} \times k^{n/2}, & \text{se } n \text{ for positivo e par} \\ k \times k^{\lfloor n/2 \rfloor} \times k^{\lfloor n/2 \rfloor}, & \text{se } n \text{ for positivo e ímpar} \end{cases}$$

https://pythontutor.com
https://tinyurl.com/y6emx5yr

Python 3.6
(known limitations)

```
1  def potencia1(k, n):
2      if n == 0:
3          return 1
4      else:
5          return k * potencia1(k, n - 1)
6
7  def potencia2(k, n):
8      if n == 0:
9          return 1
10     elif n % 2 == 0:
11         return potencia2(k, n//2) * potencia2(k, n//2)
12     else:
13         return k * potencia2(k, n//2) * potencia2(k, n//2)
14
15 print(potencia2(2,5))
```

Edit this code

line that just executed
next line to execute

<< First | < Prev | Next > | Last >>

Step 27 of 70

Customize visualization

Print output (drag lower right corner to resize)

Frames Objects

Global frame
potencia1
potencia2

function
potencia1(k, n)

function
potencia2(k, n)

potencia2
k 2
n 5

potencia2
k 2
n 2

potencia2
k 2
n 1

$$k^n = \begin{cases} 1, & \text{se } n = 0 \\ k^{n/2} \times k^{n/2}, & \text{se } n \text{ for positivo e par} \\ k \times k^{\lfloor n/2 \rfloor} \times k^{\lfloor n/2 \rfloor}, & \text{se } n \text{ for positivo e ímpar} \end{cases}$$

https://pythontutor.com
https://tinyurl.com/y6emx5yr

Python 3.6
(known limitations)

```
1  def potencia1(k, n):
2      if n == 0:
3          return 1
4      else:
5          return k * potencia1(k, n - 1)
6
7  def potencia2(k, n):
8      if n == 0:
9          return 1
10     elif n % 2 == 0:
11         return potencia2(k, n//2) * potencia2(k, n//2)
12     else:
13         return k * potencia2(k, n//2) * potencia2(k, n//2)
14
15 print(potencia2(2,5))
```

Edit this code

line that just executed
next line to execute

<< First | < Prev | Next > | Last >>

Step 28 of 70

Customize visualization

Print output (drag lower right corner to resize)

Frames Objects

Global frame
potencia1
potencia2

function
potencia1(k, n)

function
potencia2(k, n)

potencia2
k 2
n 5

potencia2
k 2
n 2

potencia2
k 2
n 1

$$k^n = \begin{cases} 1, & \text{se } n = 0 \\ k^{n/2} \times k^{n/2}, & \text{se } n \text{ for positivo e par} \\ k \times k^{\lfloor n/2 \rfloor} \times k^{\lfloor n/2 \rfloor}, & \text{se } n \text{ for positivo e ímpar} \end{cases}$$

https://pythontutor.com
https://tinyurl.com/y6emx5yr

Python 3.6
(known limitations)

```
 1  def potencia1(k, n):
 2      if n == 0:
 3          return 1
 4      else:
 5          return k * potencia1(k, n - 1)
 6
 7  def potencia2(k, n):
 8      if n == 0:
 9          return 1
10      elif n % 2 == 0:
11          return potencia2(k, n//2) * potencia2(k, n//2)
12      else:
13          return k * potencia2(k, n//2) * potencia2(k, n//2)
14
15  print(potencia2(2,5))
```

Edit this code

line that just executed
next line to execute

<< First | < Prev | Next > | Last >>

Step 29 of 70

Customize visualization

Print output (drag lower right corner to resize)

Frames Objects

Global frame
potencia1 → function potencia1(k, n)
potencia2 → function potencia2(k, n)

potencia2
k 2
n 5

potencia2
k 2
n 2

potencia2
k 2
n 1

potencia2
k 2
n 0

$$k^n = \begin{cases} 1, & \text{se } n = 0 \\ k^{n/2} \times k^{n/2}, & \text{se } n \text{ for positivo e par} \\ k \times k^{\lfloor n/2 \rfloor} \times k^{\lfloor n/2 \rfloor}, & \text{se } n \text{ for positivo e ímpar} \end{cases}$$

https://pythontutor.com
https://tinyurl.com/y6emx5yr

Python 3.6
(known limitations)

```
1  def potencia1(k, n):
2      if n == 0:
3          return 1
4      else:
5          return k * potencia1(k, n - 1)
6
7  def potencia2(k, n):
8      if n == 0:
9          return 1
10     elif n % 2 == 0:
11         return potencia2(k, n//2) * potencia2(k, n//2)
12     else:
13         return k * potencia2(k, n//2) * potencia2(k, n//2)
14
15 print(potencia2(2,5))
```

Edit this code

line that just executed
next line to execute

<< First | < Prev | Next > | Last >>

Step 30 of 70

Customize visualization

Print output (drag lower right corner to resize)

Frames Objects

Global frame
potencia1
potencia2

function
potencia1(k, n)

function
potencia2(k, n)

potencia2
k 2
n 5

potencia2
k 2
n 2

potencia2
k 2
n 1

potencia2
k 2
n 0

$$k^n = \begin{cases} 1, & \text{se } n = 0 \\ k^{n/2} \times k^{n/2}, & \text{se } n \text{ for positivo e par} \\ k \times k^{\lfloor n/2 \rfloor} \times k^{\lfloor n/2 \rfloor}, & \text{se } n \text{ for positivo e ímpar} \end{cases}$$

https://pythontutor.com
https://tinyurl.com/y6emx5yr

Python 3.6
(known limitations)

```
1  def potencia1(k, n):
2      if n == 0:
3          return 1
4      else:
5          return k * potencia1(k, n - 1)
6
7  def potencia2(k, n):
8      if n == 0:
9          return 1
10     elif n % 2 == 0:
11         return potencia2(k, n//2) * potencia2(k, n//2)
12     else:
13         return k * potencia2(k, n//2) * potencia2(k, n//2)
14
15 print(potencia2(2,5))
```

Edit this code

line that just executed
next line to execute

<< First | < Prev | Next > | Last >>

Step 31 of 70

Customize visualization

Print output (drag lower right corner to resize)

Frames Objects

Global frame
potencia1
potencia2

function
potencia1(k, n)

function
potencia2(k, n)

potencia2
k 2
n 5

potencia2
k 2
n 2

potencia2
k 2
n 1

potencia2
k 2
n 0

$$k^n = \begin{cases} 1, & \text{se } n = 0 \\ k^{n/2} \times k^{n/2}, & \text{se } n \text{ for positivo e par} \\ k \times k^{\lfloor n/2 \rfloor} \times k^{\lfloor n/2 \rfloor}, & \text{se } n \text{ for positivo e ímpar} \end{cases}$$

https://pythontutor.com
https://tinyurl.com/y6emx5yr

Python 3.6
(known limitations)

```
1  def potencia1(k, n):
2      if n == 0:
3          return 1
4      else:
5          return k * potencia1(k, n - 1)
6
7  def potencia2(k, n):
8      if n == 0:
9          return 1
10     elif n % 2 == 0:
11         return potencia2(k, n//2) * potencia2(k, n//2)
12     else:
13         return k * potencia2(k, n//2) * potencia2(k, n//2)
14
15 print(potencia2(2,5))
```

Edit this code

line that just executed
next line to execute

<< First | < Prev | Next > | Last >>

Step 32 of 70

Customize visualization

Print output (drag lower right corner to resize)

Frames Objects

Global frame
potencia1
potencia2

function
potencia1(k, n)

function
potencia2(k, n)

potencia2
k 2
n 5

potencia2
k 2
n 2

potencia2
k 2
n 1

potencia2
k 2
n 0
Return value 1

$$k^n = \begin{cases} 1, & \text{se } n = 0 \\ k^{n/2} \times k^{n/2}, & \text{se } n \text{ for positivo e par} \\ k \times k^{\lfloor n/2 \rfloor} \times k^{\lfloor n/2 \rfloor}, & \text{se } n \text{ for positivo e ímpar} \end{cases}$$

https://pythontutor.com
https://tinyurl.com/y6emx5yr

Python 3.6
(known limitations)

```
 1  def potencia1(k, n):
 2      if n == 0:
 3          return 1
 4      else:
 5          return k * potencia1(k, n - 1)
 6
 7  def potencia2(k, n):
 8      if n == 0:
 9          return 1
10      elif n % 2 == 0:
11          return potencia2(k, n//2) * potencia2(k, n//2)
12      else:
13          return k * potencia2(k, n//2) * potencia2(k, n//2)
14
15  print(potencia2(2,5))
```

Edit this code

line that just executed
next line to execute

<< First | < Prev | Next > | Last >>

Step 33 of 70

Customize visualization

Print output (drag lower right corner to resize)

Frames Objects

Global frame
potencia1
potencia2

function
potencia1(k, n)

function
potencia2(k, n)

potencia2
k 2
n 5

potencia2
k 2
n 2

potencia2
k 2
n 1

potencia2
k 2
n 0

$$k^n = \begin{cases} 1, & \text{se } n = 0 \\ k^{n/2} \times k^{n/2}, & \text{se } n \text{ for positivo e par} \\ k \times k^{\lfloor n/2 \rfloor} \times k^{\lfloor n/2 \rfloor}, & \text{se } n \text{ for positivo e ímpar} \end{cases}$$

https://pythontutor.com
https://tinyurl.com/y6emx5yr

Python 3.6
(known limitations)

```
1  def potencia1(k, n):
2      if n == 0:
3          return 1
4      else:
5          return k * potencia1(k, n - 1)
6
7  def potencia2(k, n):
8      if n == 0:
9          return 1
10     elif n % 2 == 0:
11         return potencia2(k, n//2) * potencia2(k, n//2)
12     else:
13         return k * potencia2(k, n//2) * potencia2(k, n//2)
14
15 print(potencia2(2,5))
```

Edit this code

line that just executed
next line to execute

<< First | < Prev | Next > | Last >>

Step 34 of 70

Customize visualization

Print output (drag lower right corner to resize)

Frames Objects

Global frame
potencia1
potencia2

function
potencia1(k, n)

function
potencia2(k, n)

potencia2
k 2
n 5

potencia2
k 2
n 2

potencia2
k 2
n 1

potencia2
k 2
n 0

$$k^n = \begin{cases} 1, & \text{se } n = 0 \\ k^{n/2} \times k^{n/2}, & \text{se } n \text{ for positivo e par} \\ k \times k^{\lfloor n/2 \rfloor} \times k^{\lfloor n/2 \rfloor}, & \text{se } n \text{ for positivo e ímpar} \end{cases}$$

https://pythontutor.com
https://tinyurl.com/y6emx5yr

Python 3.6
(known limitations)

```
1  def potencia1(k, n):
2      if n == 0:
3          return 1
4      else:
5          return k * potencia1(k, n - 1)
6
7  def potencia2(k, n):
8      if n == 0:
9          return 1
10     elif n % 2 == 0:
11         return potencia2(k, n//2) * potencia2(k, n//2)
12     else:
13         return k * potencia2(k, n//2) * potencia2(k, n//2)
14
15 print(potencia2(2,5))
```

Edit this code

line that just executed
next line to execute

<< First | < Prev | Next > | Last >>

Step 35 of 70

Customize visualization

Print output (drag lower right corner to resize)

Frames Objects

Global frame
potencia1
potencia2

function
potencia1(k, n)

function
potencia2(k, n)

potencia2
k 2
n 5

potencia2
k 2
n 2

potencia2
k 2
n 1

potencia2
k 2
n 0

$$k^n = \begin{cases} 1, & \text{se } n = 0 \\ k^{n/2} \times k^{n/2}, & \text{se } n \text{ for positivo e par} \\ k \times k^{\lfloor n/2 \rfloor} \times k^{\lfloor n/2 \rfloor}, & \text{se } n \text{ for positivo e ímpar} \end{cases}$$

https://pythontutor.com
https://tinyurl.com/y6emx5yr

Python 3.6
(known limitations)

```
1  def potencia1(k, n):
2      if n == 0:
3          return 1
4      else:
5          return k * potencia1(k, n - 1)
6
7  def potencia2(k, n):
8      if n == 0:
9          return 1
10     elif n % 2 == 0:
11         return potencia2(k, n//2) * potencia2(k, n//2)
12     else:
13         return k * potencia2(k, n//2) * potencia2(k, n//2)
14
15 print(potencia2(2,5))
```

Edit this code

line that just executed
next line to execute

<< First | < Prev | Next > | Last >>

Step 36 of 70

Customize visualization

Print output (drag lower right corner to resize)

Frames Objects

Global frame
potencia1
potencia2

function
potencia1(k, n)

function
potencia2(k, n)

potencia2
k 2
n 5

potencia2
k 2
n 2

potencia2
k 2
n 1

potencia2
k 2
n 0
Return value 1

$$k^n = \begin{cases} 1, & \text{se } n = 0 \\ k^{n/2} \times k^{n/2}, & \text{se } n \text{ for positivo e par} \\ k \times k^{\lfloor n/2 \rfloor} \times k^{\lfloor n/2 \rfloor}, & \text{se } n \text{ for positivo e ímpar} \end{cases}$$

https://pythontutor.com
https://tinyurl.com/y6emx5yr

Python 3.6
(known limitations)

```
1  def potencia1(k, n):
2      if n == 0:
3          return 1
4      else:
5          return k * potencia1(k, n - 1)
6
7  def potencia2(k, n):
8      if n == 0:
9          return 1
10     elif n % 2 == 0:
11         return potencia2(k, n//2) * potencia2(k, n//2)
12     else:
13         return k * potencia2(k, n//2) * potencia2(k, n//2)
14
15 print(potencia2(2,5))
```

Edit this code

line that just executed
next line to execute

<< First | < Prev | Next > | Last >>

Step 37 of 70

Customize visualization

Print output (drag lower right corner to resize)

Frames Objects

Global frame
potencia1
potencia2

function
potencia1(k, n)

function
potencia2(k, n)

potencia2
k 2
n 5

potencia2
k 2
n 2

potencia2
k 2
n 1
Return value 2

$$k^n = \begin{cases} 1, & \text{se } n = 0 \\ k^{n/2} \times k^{n/2}, & \text{se } n \text{ for positivo e par} \\ k \times k^{\lfloor n/2 \rfloor} \times k^{\lfloor n/2 \rfloor}, & \text{se } n \text{ for positivo e ímpar} \end{cases}$$

https://pythontutor.com
https://tinyurl.com/y6emx5yr

Python 3.6
(known limitations)

```
1  def potencia1(k, n):
2      if n == 0:
3          return 1
4      else:
5          return k * potencia1(k, n - 1)
6
7  def potencia2(k, n):
8      if n == 0:
9          return 1
10     elif n % 2 == 0:
11         return potencia2(k, n//2) * potencia2(k, n//2)
12     else:
13         return k * potencia2(k, n//2) * potencia2(k, n//2)
14
15 print(potencia2(2,5))
```

Edit this code

line that just executed
next line to execute

<< First | < Prev | Next > | Last >>

Step 38 of 70

Customize visualization

Print output (drag lower right corner to resize)

Frames Objects

Global frame
potencia1
potencia2

function
potencia1(k, n)

function
potencia2(k, n)

potencia2
k 2
n 5

potencia2
k 2
n 2
Return value 4

$$k^n = \begin{cases} 1, & \text{se } n = 0 \\ k^{n/2} \times k^{n/2}, & \text{se } n \text{ for positivo e par} \\ k \times k^{\lfloor n/2 \rfloor} \times k^{\lfloor n/2 \rfloor}, & \text{se } n \text{ for positivo e ímpar} \end{cases}$$

https://pythontutor.com
https://tinyurl.com/y6emx5yr

Python 3.6
(known limitations)

```
1  def potencia1(k, n):
2      if n == 0:
3          return 1
4      else:
5          return k * potencia1(k, n - 1)
6
7  def potencia2(k, n):
8      if n == 0:
9          return 1
10     elif n % 2 == 0:
11         return potencia2(k, n//2) * potencia2(k, n//2)
12     else:
13         return k * potencia2(k, n//2) * potencia2(k, n//2)
14
15 print(potencia2(2,5))
```

Edit this code

line that just executed
next line to execute

<< First | < Prev | Next > | Last >>

Step 39 of 70

Customize visualization

Print output (drag lower right corner to resize)

Frames Objects

Global frame
potencia1
potencia2

function
potencia1(k, n)

function
potencia2(k, n)

potencia2
k 2
n 5

potencia2
k 2
n 2

$$k^n = \begin{cases} 1, & \text{se } n = 0 \\ k^{n/2} \times k^{n/2}, & \text{se } n \text{ for positivo e par} \\ k \times k^{\lfloor n/2 \rfloor} \times k^{\lfloor n/2 \rfloor}, & \text{se } n \text{ for positivo e ímpar} \end{cases}$$

https://pythontutor.com
https://tinyurl.com/y6emx5yr

Python 3.6
(known limitations)

```
1  def potencia1(k, n):
2      if n == 0:
3          return 1
4      else:
5          return k * potencia1(k, n - 1)
6
7  def potencia2(k, n):
8      if n == 0:
9          return 1
10     elif n % 2 == 0:
11         return potencia2(k, n//2) * potencia2(k, n//2)
12     else:
13         return k * potencia2(k, n//2) * potencia2(k, n//2)
14
15 print(potencia2(2,5))
```

Edit this code

line that just executed
next line to execute

<< First | < Prev | Next > | Last >>

Step 40 of 70

Customize visualization

Print output (drag lower right corner to resize)

Frames Objects

Global frame
potencia1
potencia2

function
potencia1(k, n)

function
potencia2(k, n)

potencia2
k 2
n 5

potencia2
k 2
n 2

$$k^n = \begin{cases} 1, & \text{se } n = 0 \\ k^{n/2} \times k^{n/2}, & \text{se } n \text{ for positivo e par} \\ k \times k^{\lfloor n/2 \rfloor} \times k^{\lfloor n/2 \rfloor}, & \text{se } n \text{ for positivo e ímpar} \end{cases}$$

https://pythontutor.com
https://tinyurl.com/y6emx5yr

Python 3.6
(known limitations)

```
1  def potencia1(k, n):
2      if n == 0:
3          return 1
4      else:
5          return k * potencia1(k, n - 1)
6
7  def potencia2(k, n):
8      if n == 0:
9          return 1
10     elif n % 2 == 0:
11         return potencia2(k, n//2) * potencia2(k, n//2)
12     else:
13         return k * potencia2(k, n//2) * potencia2(k, n//2)
14
15 print(potencia2(2,5))
```

Edit this code

line that just executed
next line to execute

<< First | < Prev | Next > | Last >>

Step 41 of 70

Customize visualization

Print output (drag lower right corner to resize)

Frames Objects

Global frame
potencia1
potencia2

function
potencia1(k, n)

function
potencia2(k, n)

potencia2
k 2
n 5

potencia2
k 2
n 2

$$k^n = \begin{cases} 1, & \text{se } n = 0 \\ k^{n/2} \times k^{n/2}, & \text{se } n \text{ for positivo e par} \\ k \times k^{\lfloor n/2 \rfloor} \times k^{\lfloor n/2 \rfloor}, & \text{se } n \text{ for positivo e ímpar} \end{cases}$$

https://pythontutor.com
https://tinyurl.com/y6emx5yr

Python 3.6
(known limitations)

```
1  def potencia1(k, n):
2      if n == 0:
3          return 1
4      else:
5          return k * potencia1(k, n - 1)
6
7  def potencia2(k, n):
8      if n == 0:
9          return 1
10     elif n % 2 == 0:
11         return potencia2(k, n//2) * potencia2(k, n//2)
12     else:
13         return k * potencia2(k, n//2) * potencia2(k, n//2)
14
15 print(potencia2(2,5))
```

Edit this code

line that just executed
next line to execute

<< First | < Prev | Next > | Last >>

Step 42 of 70

Customize visualization

Print output (drag lower right corner to resize)

Frames Objects

Global frame
potencia1
potencia2

function
potencia1(k, n)

function
potencia2(k, n)

potencia2
k 2
n 5

potencia2
k 2
n 2

$$k^n = \begin{cases} 1, & \text{se } n = 0 \\ k^{n/2} \times k^{n/2}, & \text{se } n \text{ for positivo e par} \\ k \times k^{\lfloor n/2 \rfloor} \times k^{\lfloor n/2 \rfloor}, & \text{se } n \text{ for positivo e ímpar} \end{cases}$$

https://pythontutor.com
https://tinyurl.com/y6emx5yr

Python 3.6
(known limitations)

```
1  def potencia1(k, n):
2      if n == 0:
3          return 1
4      else:
5          return k * potencia1(k, n - 1)
6
7  def potencia2(k, n):
8      if n == 0:
9          return 1
10     elif n % 2 == 0:
11         return potencia2(k, n//2) * potencia2(k, n//2)
12     else:
13         return k * potencia2(k, n//2) * potencia2(k, n//2)
14
15 print(potencia2(2,5))
```

Edit this code

line that just executed
next line to execute

<< First | < Prev | Next > | Last >>

Step 43 of 70

Customize visualization

Print output (drag lower right corner to resize)

Frames Objects

Global frame
potencia1
potencia2

function
potencia1(k, n)

function
potencia2(k, n)

potencia2
k 2
n 5

potencia2
k 2
n 2

potencia2
k 2
n 1

$$k^n = \begin{cases} 1, & \text{se } n = 0 \\ k^{n/2} \times k^{n/2}, & \text{se } n \text{ for positivo e par} \\ k \times k^{\lfloor n/2 \rfloor} \times k^{\lfloor n/2 \rfloor}, & \text{se } n \text{ for positivo e ímpar} \end{cases}$$

https://pythontutor.com
https://tinyurl.com/y6emx5yr

Python 3.6
(known limitations)

```
1  def potencia1(k, n):
2      if n == 0:
3          return 1
4      else:
5          return k * potencia1(k, n - 1)
6
7  def potencia2(k, n):
8      if n == 0:
9          return 1
10     elif n % 2 == 0:
11         return potencia2(k, n//2) * potencia2(k, n//2)
12     else:
13         return k * potencia2(k, n//2) * potencia2(k, n//2)
14
15 print(potencia2(2,5))
```

Edit this code

line that just executed
next line to execute

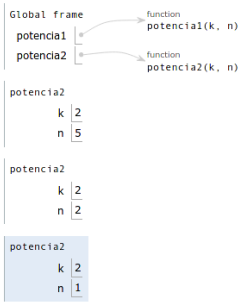
<< First | < Prev | Next > | Last >>

Step 44 of 70

Customize visualization

Print output (drag lower right corner to resize)

Frames Objects



$$k^n = \begin{cases} 1, & \text{se } n = 0 \\ k^{n/2} \times k^{n/2}, & \text{se } n \text{ for positivo e par} \\ k \times k^{\lfloor n/2 \rfloor} \times k^{\lfloor n/2 \rfloor}, & \text{se } n \text{ for positivo e ímpar} \end{cases}$$

https://pythontutor.com
https://tinyurl.com/y6emx5yr

Python 3.6
(known limitations)

```
1  def potencia1(k, n):
2      if n == 0:
3          return 1
4      else:
5          return k * potencia1(k, n - 1)
6
7  def potencia2(k, n):
8      if n == 0:
9          return 1
10     elif n % 2 == 0:
11         return potencia2(k, n//2) * potencia2(k, n//2)
12     else:
13         return k * potencia2(k, n//2) * potencia2(k, n//2)
14
15 print(potencia2(2,5))
```

Edit this code

line that just executed
next line to execute

<< First | < Prev | Next > | Last >>

Step 45 of 70

Customize visualization

Print output (drag lower right corner to resize)

Frames Objects

Global frame
potencia1
potencia2

function
potencia1(k, n)

function
potencia2(k, n)

potencia2
k 2
n 5

potencia2
k 2
n 2

potencia2
k 2
n 1

$$k^n = \begin{cases} 1, & \text{se } n = 0 \\ k^{n/2} \times k^{n/2}, & \text{se } n \text{ for positivo e par} \\ k \times k^{\lfloor n/2 \rfloor} \times k^{\lfloor n/2 \rfloor}, & \text{se } n \text{ for positivo e ímpar} \end{cases}$$

https://pythontutor.com
https://tinyurl.com/y6emx5yr

Python 3.6
(known limitations)

```
1  def potencia1(k, n):
2      if n == 0:
3          return 1
4      else:
5          return k * potencia1(k, n - 1)
6
7  def potencia2(k, n):
8      if n == 0:
9          return 1
10     elif n % 2 == 0:
11         return potencia2(k, n//2) * potencia2(k, n//2)
12     else:
13         return k * potencia2(k, n//2) * potencia2(k, n//2)
14
15 print(potencia2(2,5))
```

Edit this code

line that just executed
next line to execute

<< First | < Prev | Next > | Last >>

Step 46 of 70

Customize visualization

Print output (drag lower right corner to resize)

Frames Objects

Global frame
potencia1
potencia2

function
potencia1(k, n)

function
potencia2(k, n)

potencia2
k 2
n 5

potencia2
k 2
n 2

potencia2
k 2
n 1

$$k^n = \begin{cases} 1, & \text{se } n = 0 \\ k^{n/2} \times k^{n/2}, & \text{se } n \text{ for positivo e par} \\ k \times k^{\lfloor n/2 \rfloor} \times k^{\lfloor n/2 \rfloor}, & \text{se } n \text{ for positivo e ímpar} \end{cases}$$

https://pythontutor.com
https://tinyurl.com/y6emx5yr

Python 3.6
(known limitations)

```
 1  def potencia1(k, n):
 2      if n == 0:
 3          return 1
 4      else:
 5          return k * potencia1(k, n - 1)
 6
 7  def potencia2(k, n):
 8      if n == 0:
 9          return 1
10      elif n % 2 == 0:
11          return potencia2(k, n//2) * potencia2(k, n//2)
12      else:
13          return k * potencia2(k, n//2) * potencia2(k, n//2)
14
15  print(potencia2(2,5))
```

Edit this code

line that just executed
next line to execute

<< First | < Prev | Next > | Last >>

Step 47 of 70

Customize visualization

Print output (drag lower right corner to resize)

Frames Objects

Global frame
potencia1
potencia2

function
potencia1(k, n)

function
potencia2(k, n)

potencia2
k 2
n 5

potencia2
k 2
n 2

potencia2
k 2
n 1

potencia2
k 2
n 0

$$k^n = \begin{cases} 1, & \text{se } n = 0 \\ k^{n/2} \times k^{n/2}, & \text{se } n \text{ for positivo e par} \\ k \times k^{\lfloor n/2 \rfloor} \times k^{\lfloor n/2 \rfloor}, & \text{se } n \text{ for positivo e ímpar} \end{cases}$$

https://pythontutor.com
https://tinyurl.com/y6emx5yr

Python 3.6
(known limitations)

```
1  def potencia1(k, n):
2      if n == 0:
3          return 1
4      else:
5          return k * potencia1(k, n - 1)
6
7  def potencia2(k, n):
8      if n == 0:
9          return 1
10     elif n % 2 == 0:
11         return potencia2(k, n//2) * potencia2(k, n//2)
12     else:
13         return k * potencia2(k, n//2) * potencia2(k, n//2)
14
15 print(potencia2(2,5))
```

Edit this code

line that just executed
next line to execute

<< First | < Prev | Next > | Last >>

Step 48 of 70

Customize visualization

Print output (drag lower right corner to resize)

Frames Objects

Global frame
potencia1
potencia2

function potencia1(k, n)
function potencia2(k, n)

potencia2
k 2
n 5

potencia2
k 2
n 2

potencia2
k 2
n 1

potencia2
k 2
n 0

$$k^n = \begin{cases} 1, & \text{se } n = 0 \\ k^{n/2} \times k^{n/2}, & \text{se } n \text{ for positivo e par} \\ k \times k^{\lfloor n/2 \rfloor} \times k^{\lfloor n/2 \rfloor}, & \text{se } n \text{ for positivo e ímpar} \end{cases}$$

https://pythontutor.com
https://tinyurl.com/y6emx5yr

Python 3.6
(known limitations)

```
1  def potencia1(k, n):
2      if n == 0:
3          return 1
4      else:
5          return k * potencia1(k, n - 1)
6
7  def potencia2(k, n):
8      if n == 0:
9          return 1
10     elif n % 2 == 0:
11         return potencia2(k, n//2) * potencia2(k, n//2)
12     else:
13         return k * potencia2(k, n//2) * potencia2(k, n//2)
14
15 print(potencia2(2,5))
```

Edit this code

line that just executed
next line to execute

<< First | < Prev | Next > | Last >>

Step 49 of 70

Customize visualization

Print output (drag lower right corner to resize)

Frames Objects

Global frame
potencia1
potencia2

function potencia1(k, n)
function potencia2(k, n)

potencia2
k 2
n 5

potencia2
k 2
n 2

potencia2
k 2
n 1

potencia2
k 2
n 0

$$k^n = \begin{cases} 1, & \text{se } n = 0 \\ k^{n/2} \times k^{n/2}, & \text{se } n \text{ for positivo e par} \\ k \times k^{\lfloor n/2 \rfloor} \times k^{\lfloor n/2 \rfloor}, & \text{se } n \text{ for positivo e ímpar} \end{cases}$$

https://pythontutor.com
https://tinyurl.com/y6emx5yr

Python 3.6
(known limitations)

```
1  def potencia1(k, n):
2      if n == 0:
3          return 1
4      else:
5          return k * potencia1(k, n - 1)
6
7  def potencia2(k, n):
8      if n == 0:
9          return 1
10     elif n % 2 == 0:
11         return potencia2(k, n//2) * potencia2(k, n//2)
12     else:
13         return k * potencia2(k, n//2) * potencia2(k, n//2)
14
15 print(potencia2(2,5))
```

Edit this code

line that just executed
next line to execute

<< First | < Prev | Next > | Last >>

Step 50 of 70

Customize visualization

Print output (drag lower right corner to resize)

Frames Objects

Global frame
potencia1
potencia2

function
potencia1(k, n)

function
potencia2(k, n)

potencia2
k 2
n 5

potencia2
k 2
n 2

potencia2
k 2
n 1

potencia2
k 2
n 0
Return value 1

$$k^n = \begin{cases} 1, & \text{se } n = 0 \\ k^{n/2} \times k^{n/2}, & \text{se } n \text{ for positivo e par} \\ k \times k^{\lfloor n/2 \rfloor} \times k^{\lfloor n/2 \rfloor}, & \text{se } n \text{ for positivo e ímpar} \end{cases}$$

https://pythontutor.com
https://tinyurl.com/y6emx5yr

Python 3.6
(known limitations)

```
1  def potencia1(k, n):
2      if n == 0:
3          return 1
4      else:
5          return k * potencia1(k, n - 1)
6
7  def potencia2(k, n):
8      if n == 0:
9          return 1
10     elif n % 2 == 0:
11         return potencia2(k, n//2) * potencia2(k, n//2)
12     else:
13         return k * potencia2(k, n//2) * potencia2(k, n//2)
14
15 print(potencia2(2,5))
```

Edit this code

line that just executed
next line to execute

<< First | < Prev | Next > | Last >>

Step 51 of 70

Customize visualization

Print output (drag lower right corner to resize)

Frames Objects

Global frame
potencia1
potencia2

function
potencia1(k, n)

function
potencia2(k, n)

potencia2
k 2
n 5

potencia2
k 2
n 2

potencia2
k 2
n 1

potencia2
k 2
n 0

$$k^n = \begin{cases} 1, & \text{se } n = 0 \\ k^{n/2} \times k^{n/2}, & \text{se } n \text{ for positivo e par} \\ k \times k^{\lfloor n/2 \rfloor} \times k^{\lfloor n/2 \rfloor}, & \text{se } n \text{ for positivo e ímpar} \end{cases}$$

https://pythontutor.com
https://tinyurl.com/y6emx5yr

Python 3.6
(known limitations)

```
1  def potencia1(k, n):
2      if n == 0:
3          return 1
4      else:
5          return k * potencia1(k, n - 1)
6
7  def potencia2(k, n):
8      if n == 0:
9          return 1
10     elif n % 2 == 0:
11         return potencia2(k, n//2) * potencia2(k, n//2)
12     else:
13         return k * potencia2(k, n//2) * potencia2(k, n//2)
14
15 print(potencia2(2,5))
```

Edit this code

line that just executed
next line to execute

<< First | < Prev | Next > | Last >>

Step 52 of 70

Customize visualization

Print output (drag lower right corner to resize)

Frames Objects

Global frame
potencia1 → function potencia1(k, n)
potencia2 → function potencia2(k, n)

potencia2
k 2
n 5

potencia2
k 2
n 2

potencia2
k 2
n 1

potencia2
k 2
n 0

$$k^n = \begin{cases} 1, & \text{se } n = 0 \\ k^{n/2} \times k^{n/2}, & \text{se } n \text{ for positivo e par} \\ k \times k^{\lfloor n/2 \rfloor} \times k^{\lfloor n/2 \rfloor}, & \text{se } n \text{ for positivo e ímpar} \end{cases}$$

https://pythontutor.com
https://tinyurl.com/y6emx5yr

Python 3.6
(known limitations)

```
1  def potencia1(k, n):
2      if n == 0:
3          return 1
4      else:
5          return k * potencia1(k, n - 1)
6
7  def potencia2(k, n):
8      if n == 0:
9          return 1
10     elif n % 2 == 0:
11         return potencia2(k, n//2) * potencia2(k, n//2)
12     else:
13         return k * potencia2(k, n//2) * potencia2(k, n//2)
14
15 print(potencia2(2,5))
```

Edit this code

line that just executed
next line to execute

<< First | < Prev | Next > | Last >>

Step 53 of 70

Customize visualization

Print output (drag lower right corner to resize)

Frames Objects

Global frame
potencia1
potencia2

function
potencia1(k, n)

function
potencia2(k, n)

potencia2
k 2
n 5

potencia2
k 2
n 2

potencia2
k 2
n 1

potencia2
k 2
n 0

$$k^n = \begin{cases} 1, & \text{se } n = 0 \\ k^{n/2} \times k^{n/2}, & \text{se } n \text{ for positivo e par} \\ k \times k^{\lfloor n/2 \rfloor} \times k^{\lfloor n/2 \rfloor}, & \text{se } n \text{ for positivo e ímpar} \end{cases}$$

https://pythontutor.com
https://tinyurl.com/y6emx5yr

Python 3.6
(known limitations)

```
1  def potencia1(k, n):
2      if n == 0:
3          return 1
4      else:
5          return k * potencia1(k, n - 1)
6
7  def potencia2(k, n):
8      if n == 0:
9          return 1
10     elif n % 2 == 0:
11         return potencia2(k, n//2) * potencia2(k, n//2)
12     else:
13         return k * potencia2(k, n//2) * potencia2(k, n//2)
14
15 print(potencia2(2,5))
```

Edit this code

line that just executed
next line to execute

<< First | < Prev | Next > | Last >>

Step 54 of 70

Customize visualization

Print output (drag lower right corner to resize)

Frames Objects

Global frame
potencia1
potencia2

function
potencia1(k, n)

function
potencia2(k, n)

potencia2
k 2
n 5

potencia2
k 2
n 2

potencia2
k 2
n 1

potencia2
k 2
n 0
Return value 1

$$k^n = \begin{cases} 1, & \text{se } n = 0 \\ k^{n/2} \times k^{n/2}, & \text{se } n \text{ for positivo e par} \\ k \times k^{\lfloor n/2 \rfloor} \times k^{\lfloor n/2 \rfloor}, & \text{se } n \text{ for positivo e ímpar} \end{cases}$$

https://pythontutor.com
https://tinyurl.com/y6emx5yr

Python 3.6
(known limitations)

```
1  def potencia1(k, n):
2      if n == 0:
3          return 1
4      else:
5          return k * potencia1(k, n - 1)
6
7  def potencia2(k, n):
8      if n == 0:
9          return 1
10     elif n % 2 == 0:
11         return potencia2(k, n//2) * potencia2(k, n//2)
12     else:
13         return k * potencia2(k, n//2) * potencia2(k, n//2)
14
15 print(potencia2(2,5))
```

Edit this code

line that just executed
next line to execute

<< First | < Prev | Next > | Last >>

Step 55 of 70

Customize visualization

Print output (drag lower right corner to resize)

Frames Objects

Global frame
potencia1
potencia2

function
potencia1(k, n)

function
potencia2(k, n)

potencia2
k 2
n 5

potencia2
k 2
n 2

potencia2
k 2
n 1
Return value 2

$$k^n = \begin{cases} 1, & \text{se } n = 0 \\ k^{n/2} \times k^{n/2}, & \text{se } n \text{ for positivo e par} \\ k \times k^{\lfloor n/2 \rfloor} \times k^{\lfloor n/2 \rfloor}, & \text{se } n \text{ for positivo e ímpar} \end{cases}$$

https://pythontutor.com
https://tinyurl.com/y6emx5yr

Python 3.6
(known limitations)

```
 1  def potencia1(k, n):
 2      if n == 0:
 3          return 1
 4      else:
 5          return k * potencia1(k, n - 1)
 6
 7  def potencia2(k, n):
 8      if n == 0:
 9          return 1
10      elif n % 2 == 0:
11          return potencia2(k, n//2) * potencia2(k, n//2)
12      else:
13          return k * potencia2(k, n//2) * potencia2(k, n//2)
14
15  print(potencia2(2,5))
```

Edit this code

line that just executed
next line to execute

<< First | < Prev | Next > | Last >>

Step 56 of 70

Customize visualization

Print output (drag lower right corner to resize)

Frames Objects

Global frame
potencia1
potencia2

function
potencia1(k, n)

function
potencia2(k, n)

potencia2
k 2
n 5

potencia2
k 2
n 2

potencia2
k 2
n 1

$$k^n = \begin{cases} 1, & \text{se } n = 0 \\ k^{n/2} \times k^{n/2}, & \text{se } n \text{ for positivo e par} \\ k \times k^{\lfloor n/2 \rfloor} \times k^{\lfloor n/2 \rfloor}, & \text{se } n \text{ for positivo e ímpar} \end{cases}$$

https://pythontutor.com
https://tinyurl.com/y6emx5yr

Python 3.6
(known limitations)

```
1  def potencia1(k, n):
2      if n == 0:
3          return 1
4      else:
5          return k * potencia1(k, n - 1)
6
7  def potencia2(k, n):
8      if n == 0:
9          return 1
10     elif n % 2 == 0:
11         return potencia2(k, n//2) * potencia2(k, n//2)
12     else:
13         return k * potencia2(k, n//2) * potencia2(k, n//2)
14
15 print(potencia2(2,5))
```

Edit this code

line that just executed
next line to execute

<< First | < Prev | Next > | Last >>

Step 57 of 70

Customize visualization

Print output (drag lower right corner to resize)

Frames Objects

Global frame
potencia1
potencia2

function
potencia1(k, n)

function
potencia2(k, n)

potencia2
k 2
n 5

potencia2
k 2
n 2

potencia2
k 2
n 1

$$k^n = \begin{cases} 1, & \text{se } n = 0 \\ k^{n/2} \times k^{n/2}, & \text{se } n \text{ for positivo e par} \\ k \times k^{\lfloor n/2 \rfloor} \times k^{\lfloor n/2 \rfloor}, & \text{se } n \text{ for positivo e ímpar} \end{cases}$$

https://pythontutor.com
https://tinyurl.com/y6emx5yr

Python 3.6
(known limitations)

```
1  def potencia1(k, n):
2      if n == 0:
3          return 1
4      else:
5          return k * potencia1(k, n - 1)
6
7  def potencia2(k, n):
8      if n == 0:
9          return 1
10     elif n % 2 == 0:
11         return potencia2(k, n//2) * potencia2(k, n//2)
12     else:
13         return k * potencia2(k, n//2) * potencia2(k, n//2)
14
15 print(potencia2(2,5))
```

Edit this code

line that just executed
next line to execute

<< First | < Prev | Next > | Last >>

Step 58 of 70

Customize visualization

Print output (drag lower right corner to resize)

Frames Objects

Global frame
potencia1
potencia2

function
potencia1(k, n)

function
potencia2(k, n)

potencia2
k 2
n 5

potencia2
k 2
n 2

potencia2
k 2
n 1

$$k^n = \begin{cases} 1, & \text{se } n = 0 \\ k^{n/2} \times k^{n/2}, & \text{se } n \text{ for positivo e par} \\ k \times k^{\lfloor n/2 \rfloor} \times k^{\lfloor n/2 \rfloor}, & \text{se } n \text{ for positivo e ímpar} \end{cases}$$

https://pythontutor.com
https://tinyurl.com/y6emx5yr

Python 3.6
(known limitations)

```
1  def potencia1(k, n):
2      if n == 0:
3          return 1
4      else:
5          return k * potencia1(k, n - 1)
6
7  def potencia2(k, n):
8      if n == 0:
9          return 1
10     elif n % 2 == 0:
11         return potencia2(k, n//2) * potencia2(k, n//2)
12     else:
13         return k * potencia2(k, n//2) * potencia2(k, n//2)
14
15 print(potencia2(2,5))
```

Edit this code

line that just executed
next line to execute

<< First | < Prev | Next > | Last >>

Step 59 of 70

Customize visualization

Print output (drag lower right corner to resize)

Frames Objects

Global frame
potencia1
potencia2

function
potencia1(k, n)

function
potencia2(k, n)

potencia2
k 2
n 5

potencia2
k 2
n 2

potencia2
k 2
n 1

$$k^n = \begin{cases} 1, & \text{se } n = 0 \\ k^{n/2} \times k^{n/2}, & \text{se } n \text{ for positivo e par} \\ k \times k^{\lfloor n/2 \rfloor} \times k^{\lfloor n/2 \rfloor}, & \text{se } n \text{ for positivo e ímpar} \end{cases}$$

https://pythontutor.com
https://tinyurl.com/y6emx5yr

Python 3.6
(known limitations)

```
 1  def potencia1(k, n):
 2      if n == 0:
 3          return 1
 4      else:
 5          return k * potencia1(k, n - 1)
 6
 7  def potencia2(k, n):
 8      if n == 0:
 9          return 1
10      elif n % 2 == 0:
11          return potencia2(k, n//2) * potencia2(k, n//2)
12      else:
13          return k * potencia2(k, n//2) * potencia2(k, n//2)
14
15  print(potencia2(2,5))
```

Edit this code

line that just executed
next line to execute

<< First | < Prev | Next > | Last >>

Step 60 of 70

Customize visualization

Print output (drag lower right corner to resize)

Frames Objects

Global frame
potencia1
potencia2

function
potencia1(k, n)

function
potencia2(k, n)

potencia2
k 2
n 5

potencia2
k 2
n 2

potencia2
k 2
n 1

potencia2
k 2
n 0

$$k^n = \begin{cases} 1, & \text{se } n = 0 \\ k^{n/2} \times k^{n/2}, & \text{se } n \text{ for positivo e par} \\ k \times k^{\lfloor n/2 \rfloor} \times k^{\lfloor n/2 \rfloor}, & \text{se } n \text{ for positivo e ímpar} \end{cases}$$

https://pythontutor.com
https://tinyurl.com/y6emx5yr

Python 3.6
(known limitations)

```
1  def potencia1(k, n):
2      if n == 0:
3          return 1
4      else:
5          return k * potencia1(k, n - 1)
6
7  def potencia2(k, n):
8      if n == 0:
9          return 1
10     elif n % 2 == 0:
11         return potencia2(k, n//2) * potencia2(k, n//2)
12     else:
13         return k * potencia2(k, n//2) * potencia2(k, n//2)
14
15 print(potencia2(2,5))
```

Edit this code

line that just executed
next line to execute

<< First | < Prev | Next > | Last >>

Step 61 of 70

Customize visualization

Print output (drag lower right corner to resize)

Frames Objects

Global frame
potencia1
potencia2

function
potencia1(k, n)

function
potencia2(k, n)

potencia2
k 2
n 5

potencia2
k 2
n 2

potencia2
k 2
n 1

potencia2
k 2
n 0

$$k^n = \begin{cases} 1, & \text{se } n = 0 \\ k^{n/2} \times k^{n/2}, & \text{se } n \text{ for positivo e par} \\ k \times k^{\lfloor n/2 \rfloor} \times k^{\lfloor n/2 \rfloor}, & \text{se } n \text{ for positivo e ímpar} \end{cases}$$

https://pythontutor.com
https://tinyurl.com/y6emx5yr

Python 3.6
(known limitations)

```
1  def potencia1(k, n):
2      if n == 0:
3          return 1
4      else:
5          return k * potencia1(k, n - 1)
6
7  def potencia2(k, n):
8      if n == 0:
9          return 1
10     elif n % 2 == 0:
11         return potencia2(k, n//2) * potencia2(k, n//2)
12     else:
13         return k * potencia2(k, n//2) * potencia2(k, n//2)
14
15 print(potencia2(2,5))
```

Edit this code

line that just executed
next line to execute

<< First | < Prev | Next > | Last >>

Step 62 of 70

Customize visualization

Print output (drag lower right corner to resize)

Frames Objects

Global frame
potencia1
potencia2

function
potencia1(k, n)

function
potencia2(k, n)

potencia2
k 2
n 5

potencia2
k 2
n 2

potencia2
k 2
n 1

potencia2
k 2
n 0

$$k^n = \begin{cases} 1, & \text{se } n = 0 \\ k^{n/2} \times k^{n/2}, & \text{se } n \text{ for positivo e par} \\ k \times k^{\lfloor n/2 \rfloor} \times k^{\lfloor n/2 \rfloor}, & \text{se } n \text{ for positivo e ímpar} \end{cases}$$

https://pythontutor.com
https://tinyurl.com/y6emx5yr

Python 3.6
(known limitations)

```
1  def potencia1(k, n):
2      if n == 0:
3          return 1
4      else:
5          return k * potencia1(k, n - 1)
6
7  def potencia2(k, n):
8      if n == 0:
9          return 1
10     elif n % 2 == 0:
11         return potencia2(k, n//2) * potencia2(k, n//2)
12     else:
13         return k * potencia2(k, n//2) * potencia2(k, n//2)
14
15 print(potencia2(2,5))
```

Edit this code

line that just executed
next line to execute

<< First | < Prev | Next > | Last >>

Step 63 of 70

Customize visualization

Print output (drag lower right corner to resize)

Frames Objects

Global frame
potencia1
potencia2

function
potencia1(k, n)

function
potencia2(k, n)

potencia2
k 2
n 5

potencia2
k 2
n 2

potencia2
k 2
n 1

potencia2
k 2
n 0
Return value 1

$$k^n = \begin{cases} 1, & \text{se } n = 0 \\ k^{n/2} \times k^{n/2}, & \text{se } n \text{ for positivo e par} \\ k \times k^{\lfloor n/2 \rfloor} \times k^{\lfloor n/2 \rfloor}, & \text{se } n \text{ for positivo e ímpar} \end{cases}$$

https://pythontutor.com
https://tinyurl.com/y6emx5yr

Python 3.6
(known limitations)

```
1  def potencia1(k, n):
2      if n == 0:
3          return 1
4      else:
5          return k * potencia1(k, n - 1)
6
7  def potencia2(k, n):
8      if n == 0:
9          return 1
10     elif n % 2 == 0:
11         return potencia2(k, n//2) * potencia2(k, n//2)
12     else:
13         return k * potencia2(k, n//2) * potencia2(k, n//2)
14
15 print(potencia2(2,5))
```

Edit this code

line that just executed
next line to execute

<< First | < Prev | Next > | Last >>

Step 64 of 70

Customize visualization

Print output (drag lower right corner to resize)

Frames Objects

Global frame
potencia1
potencia2

function
potencia1(k, n)

function
potencia2(k, n)

potencia2
k 2
n 5

potencia2
k 2
n 2

potencia2
k 2
n 1

potencia2
k 2
n 0

$$k^n = \begin{cases} 1, & \text{se } n = 0 \\ k^{n/2} \times k^{n/2}, & \text{se } n \text{ for positivo e par} \\ k \times k^{\lfloor n/2 \rfloor} \times k^{\lfloor n/2 \rfloor}, & \text{se } n \text{ for positivo e ímpar} \end{cases}$$

https://pythontutor.com
https://tinyurl.com/y6emx5yr

Python 3.6
(known limitations)

```
1  def potencia1(k, n):
2      if n == 0:
3          return 1
4      else:
5          return k * potencia1(k, n - 1)
6
7  def potencia2(k, n):
8      if n == 0:
9          return 1
10     elif n % 2 == 0:
11         return potencia2(k, n//2) * potencia2(k, n//2)
12     else:
13         return k * potencia2(k, n//2) * potencia2(k, n//2)
14
15 print(potencia2(2,5))
```

Edit this code

line that just executed
next line to execute

<< First | < Prev | Next > | Last >>

Step 65 of 70

Customize visualization

Print output (drag lower right corner to resize)

Frames Objects

Global frame
potencia1
potencia2

function
potencia1(k, n)

function
potencia2(k, n)

potencia2
k 2
n 5

potencia2
k 2
n 2

potencia2
k 2
n 1

potencia2
k 2
n 0

$$k^n = \begin{cases} 1, & \text{se } n = 0 \\ k^{n/2} \times k^{n/2}, & \text{se } n \text{ for positivo e par} \\ k \times k^{\lfloor n/2 \rfloor} \times k^{\lfloor n/2 \rfloor}, & \text{se } n \text{ for positivo e ímpar} \end{cases}$$

https://pythontutor.com
https://tinyurl.com/y6emx5yr

Python 3.6
(known limitations)

```
1  def potencia1(k, n):
2      if n == 0:
3          return 1
4      else:
5          return k * potencia1(k, n - 1)
6
7  def potencia2(k, n):
8      if n == 0:
9          return 1
10     elif n % 2 == 0:
11         return potencia2(k, n//2) * potencia2(k, n//2)
12     else:
13         return k * potencia2(k, n//2) * potencia2(k, n//2)
14
15 print(potencia2(2,5))
```

Edit this code

line that just executed
next line to execute

<< First | < Prev | Next > | Last >>

Step 66 of 70

Customize visualization

Print output (drag lower right corner to resize)

Frames Objects

Global frame
potencia1
potencia2

function
potencia1(k, n)

function
potencia2(k, n)

potencia2
k 2
n 5

potencia2
k 2
n 2

potencia2
k 2
n 1

potencia2
k 2
n 0

$$k^n = \begin{cases} 1, & \text{se } n = 0 \\ k^{n/2} \times k^{n/2}, & \text{se } n \text{ for positivo e par} \\ k \times k^{\lfloor n/2 \rfloor} \times k^{\lfloor n/2 \rfloor}, & \text{se } n \text{ for positivo e ímpar} \end{cases}$$

https://pythontutor.com
https://tinyurl.com/y6emx5yr

Python 3.6
(known limitations)

```
1  def potencia1(k, n):
2      if n == 0:
3          return 1
4      else:
5          return k * potencia1(k, n - 1)
6
7  def potencia2(k, n):
8      if n == 0:
9          return 1
10     elif n % 2 == 0:
11         return potencia2(k, n//2) * potencia2(k, n//2)
12     else:
13         return k * potencia2(k, n//2) * potencia2(k, n//2)
14
15 print(potencia2(2,5))
```

Edit this code

line that just executed
next line to execute

<< First | < Prev | Next > | Last >>

Step 67 of 70

Customize visualization

Print output (drag lower right corner to resize)

Frames Objects

Global frame
potencia1 → function potencia1(k, n)
potencia2 → function potencia2(k, n)

potencia2
k 2
n 5

potencia2
k 2
n 2

potencia2
k 2
n 1

potencia2
k 2
n 0
Return value 1

$$k^n = \begin{cases} 1, & \text{se } n = 0 \\ k^{n/2} \times k^{n/2}, & \text{se } n \text{ for positivo e par} \\ k \times k^{\lfloor n/2 \rfloor} \times k^{\lfloor n/2 \rfloor}, & \text{se } n \text{ for positivo e ímpar} \end{cases}$$

https://pythontutor.com
https://tinyurl.com/y6emx5yr

Python 3.6
(known limitations)

```
1  def potencia1(k, n):
2      if n == 0:
3          return 1
4      else:
5          return k * potencia1(k, n - 1)
6
7  def potencia2(k, n):
8      if n == 0:
9          return 1
10     elif n % 2 == 0:
11         return potencia2(k, n//2) * potencia2(k, n//2)
12     else:
13         return k * potencia2(k, n//2) * potencia2(k, n//2)
14
15 print(potencia2(2,5))
```

Edit this code

line that just executed
next line to execute

<< First | < Prev | Next > | Last >>

Step 68 of 70

Customize visualization

Print output (drag lower right corner to resize)

Frames Objects

Global frame
potencia1
potencia2

function
potencia1(k, n)

function
potencia2(k, n)

potencia2
k 2
n 5

potencia2
k 2
n 2

potencia2
k 2
n 1
Return value 2

$$k^n = \begin{cases} 1, & \text{se } n = 0 \\ k^{n/2} \times k^{n/2}, & \text{se } n \text{ for positivo e par} \\ k \times k^{\lfloor n/2 \rfloor} \times k^{\lfloor n/2 \rfloor}, & \text{se } n \text{ for positivo e ímpar} \end{cases}$$

https://pythontutor.com
https://tinyurl.com/y6emx5yr

Python 3.6
(known limitations)

```
1  def potencia1(k, n):
2      if n == 0:
3          return 1
4      else:
5          return k * potencia1(k, n - 1)
6
7  def potencia2(k, n):
8      if n == 0:
9          return 1
10     elif n % 2 == 0:
11         return potencia2(k, n//2) * potencia2(k, n//2)
12     else:
13         return k * potencia2(k, n//2) * potencia2(k, n//2)
14
15 print(potencia2(2,5))
```

Edit this code

line that just executed
next line to execute

<< First | < Prev | Next > | Last >>

Step 69 of 70

Customize visualization

Print output (drag lower right corner to resize)

Frames Objects

Global frame
potencia1
potencia2

function
potencia1(k, n)

function
potencia2(k, n)

potencia2
k 2
n 5

potencia2
k 2
n 2
Return value 4

$$k^n = \begin{cases} 1, & \text{se } n = 0 \\ k^{n/2} \times k^{n/2}, & \text{se } n \text{ for positivo e par} \\ k \times k^{\lfloor n/2 \rfloor} \times k^{\lfloor n/2 \rfloor}, & \text{se } n \text{ for positivo e ímpar} \end{cases}$$

https://pythontutor.com
https://tinyurl.com/y6emx5yr

Python 3.6
(known limitations)

```
1  def potencia1(k, n):
2      if n == 0:
3          return 1
4      else:
5          return k * potencia1(k, n - 1)
6
7  def potencia2(k, n):
8      if n == 0:
9          return 1
10     elif n % 2 == 0:
11         return potencia2(k, n//2) * potencia2(k, n//2)
12     else:
13         return k * potencia2(k, n//2) * potencia2(k, n//2)
14
15 print(potencia2(2,5))
```

Edit this code

line that just executed
next line to execute

<< First | < Prev | Next > | Last >>

Step 70 of 70

Customize visualization

Print output (drag lower right corner to resize)

Frames Objects

Global frame
potencia1 → function potencia1(k, n)
potencia2 → function potencia2(k, n)

potencia2
k 2
n 5
Return value 32

$$k^n = \begin{cases} 1, & \text{se } n = 0 \\ k^{n/2} \times k^{n/2}, & \text{se } n \text{ for positivo e par} \\ k \times k^{\lfloor n/2 \rfloor} \times k^{\lfloor n/2 \rfloor}, & \text{se } n \text{ for positivo e ímpar} \end{cases}$$

https://pythontutor.com
https://tinyurl.com/y6emx5yr

Python 3.6
(known limitations)

```
1  def potencia1(k, n):
2      if n == 0:
3          return 1
4      else:
5          return k * potencia1(k, n - 1)
6
7  def potencia2(k, n):
8      if n == 0:
9          return 1
10     elif n % 2 == 0:
11         return potencia2(k, n//2) * potencia2(k, n//2)
12     else:
13         return k * potencia2(k, n//2) * potencia2(k, n//2)
14
15 print(potencia2(2,5))
```

Edit this code

line that just executed
next line to execute

<< First | < Prev | Next > | Last >>

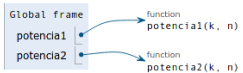
Done running (70 steps)

Customize visualization

Print output (drag lower right corner to resize)

```
32
```

Frames Objects



$$k^n = \begin{cases} 1, & \text{se } n = 0 \\ k^{n/2} \times k^{n/2}, & \text{se } n \text{ for positivo e par} \\ k \times k^{\lfloor n/2 \rfloor} \times k^{\lfloor n/2 \rfloor}, & \text{se } n \text{ for positivo e ímpar} \end{cases}$$

https://pythontutor.com
https://tinyurl.com/y6emx5yr

Python 3.6
(known limitations)

```
1  def potencia1(k, n):
2      if n == 0:
3          return 1
4      else:
5          return k * potencia1(k, n - 1)
6
7  def potencia2(k, n):
8      if n == 0:
9          return 1
10     elif n % 2 == 0:
11         return potencia2(k, n//2) * potencia2(k, n//2)
12     else:
13         return k * potencia2(k, n//2) * potencia2(k, n//2)
14
15 print(potencia2(2,5))
```

Edit this code

line that just executed
next line to execute

<< First | < Prev | Next > | Last >>

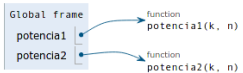
Done running (70 steps)

Customize visualization

Print output (drag lower right corner to resize)

```
32
```

Frames Objects

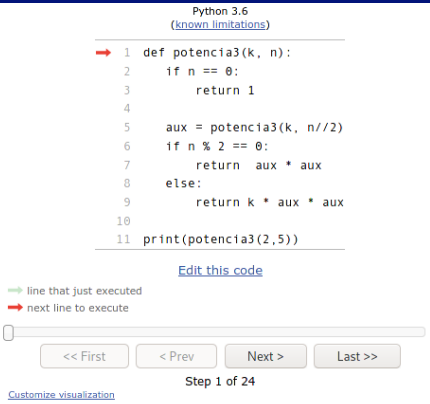


$$k^n = \begin{cases} 1, & \text{se } n = 0 \\ k^{n/2} \times k^{n/2}, & \text{se } n \text{ for positivo e par} \\ k \times k^{\lfloor n/2 \rfloor} \times k^{\lfloor n/2 \rfloor}, & \text{se } n \text{ for positivo e ímpar} \end{cases}$$

https://pythontutor.com
https://tinyurl.com/y6emx5yr

Implementação 3

$$k^n = \begin{cases} 1, & \text{se } n = 0 \\ k^{n/2} \times k^{n/2}, & \text{se } n \text{ for positivo e par} \\ k \times k^{\lfloor n/2 \rfloor} \times k^{\lfloor n/2 \rfloor}, & \text{se } n \text{ for positivo e ímpar} \end{cases}$$



$$k^n = \begin{cases} 1, & \text{se } n = 0 \\ k^{n/2} \times k^{n/2}, & \text{se } n \text{ for positivo e par} \\ k \times k^{\lfloor n/2 \rfloor} \times k^{\lfloor n/2 \rfloor}, & \text{se } n \text{ for positivo e ímpar} \end{cases}$$

https://pythontutor.com

https://tinyurl.com/mxhbmhd8

Python 3.6
(known limitations)

```
1  def potencia3(k, n):
2      if n == 0:
3          return 1
4
5      aux = potencia3(k, n//2)
6      if n % 2 == 0:
7          return  aux * aux
8      else:
9          return k * aux * aux
10
11 print(potencia3(2,5))
```

Edit this code

line that just executed
next line to execute

<< First | < Prev | Next > | Last >>

Step 2 of 24

Customize visualization

Print output (drag lower right corner to resize)

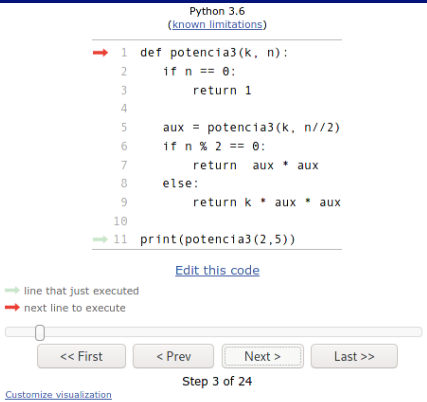
Frames Objects

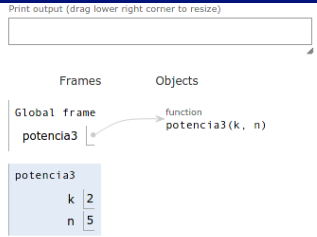
Global frame
potencia3

function
potencia3(k, n)

$$k^n = \begin{cases} 1, & \text{se } n = 0 \\ k^{n/2} \times k^{n/2}, & \text{se } n \text{ for positivo e par} \\ k \times k^{\lfloor n/2 \rfloor} \times k^{\lfloor n/2 \rfloor}, & \text{se } n \text{ for positivo e ímpar} \end{cases}$$

https://pythontutor.com
https://tinyurl.com/mxhbmhd8

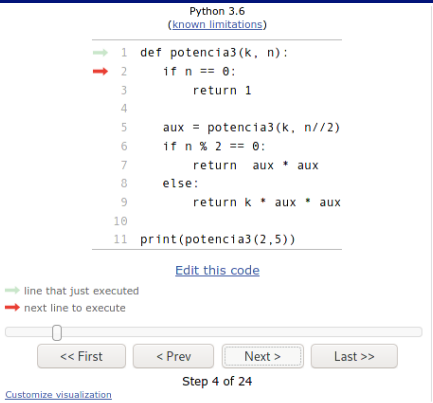




$$k^n = \begin{cases} 1, & \text{se } n = 0 \\ k^{n/2} \times k^{n/2}, & \text{se } n \text{ for positivo e par} \\ k \times k^{\lfloor n/2 \rfloor} \times k^{\lfloor n/2 \rfloor}, & \text{se } n \text{ for positivo e ímpar} \end{cases}$$

https://pythontutor.com

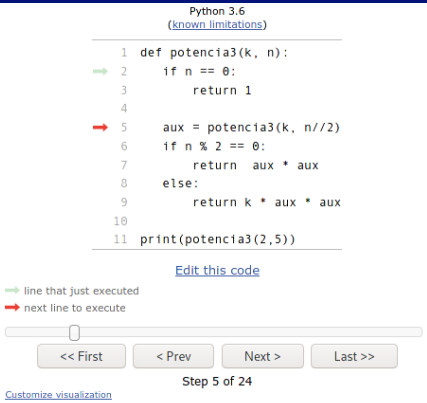
https://tinyurl.com/mxhbmhd8

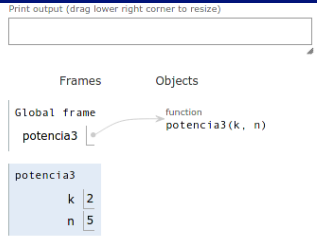


$$k^n = \begin{cases} 1, & \text{se } n = 0 \\ k^{n/2} \times k^{n/2}, & \text{se } n \text{ for positivo e par} \\ k \times k^{\lfloor n/2 \rfloor} \times k^{\lfloor n/2 \rfloor}, & \text{se } n \text{ for positivo e ímpar} \end{cases}$$

https://pythontutor.com

https://tinyurl.com/mxhbmhd8

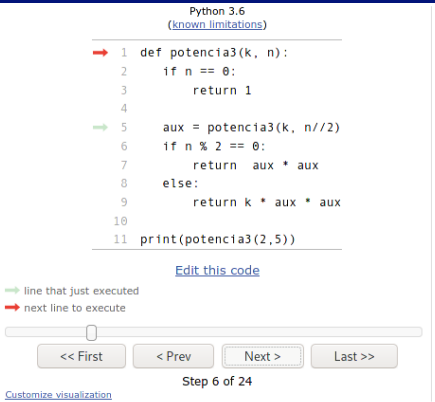


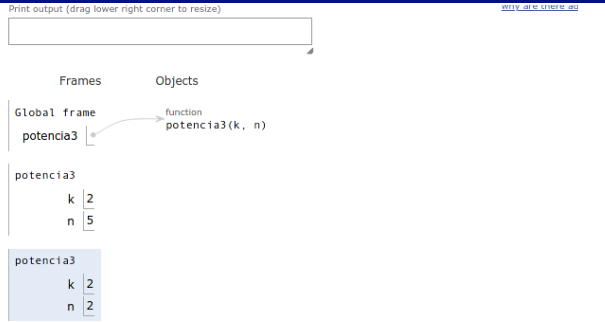


$$k^n = \begin{cases} 1, & \text{se } n = 0 \\ k^{n/2} \times k^{n/2}, & \text{se } n \text{ for positivo e par} \\ k \times k^{\lfloor n/2 \rfloor} \times k^{\lfloor n/2 \rfloor}, & \text{se } n \text{ for positivo e ímpar} \end{cases}$$

https://pythontutor.com

https://tinyurl.com/mxhbmhd8

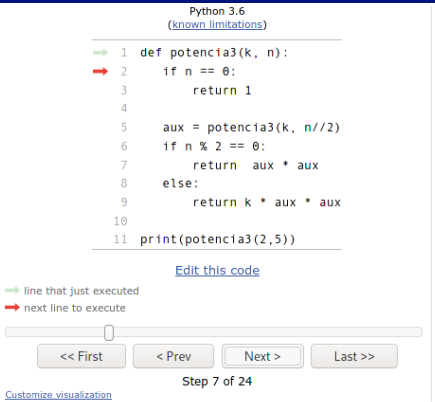


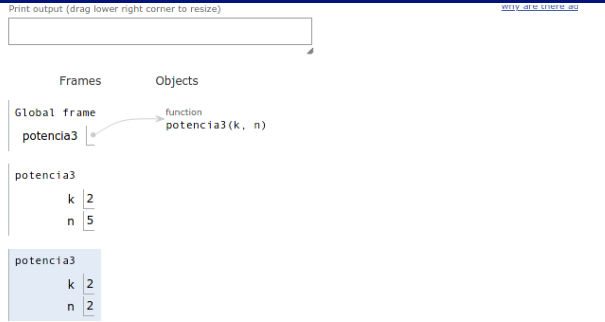


$$k^n = \begin{cases} 1, & \text{se } n = 0 \\ k^{n/2} \times k^{n/2}, & \text{se } n \text{ for positivo e par} \\ k \times k^{\lfloor n/2 \rfloor} \times k^{\lfloor n/2 \rfloor}, & \text{se } n \text{ for positivo e ímpar} \end{cases}$$

https://pythontutor.com

https://tinyurl.com/mxhbmhd8

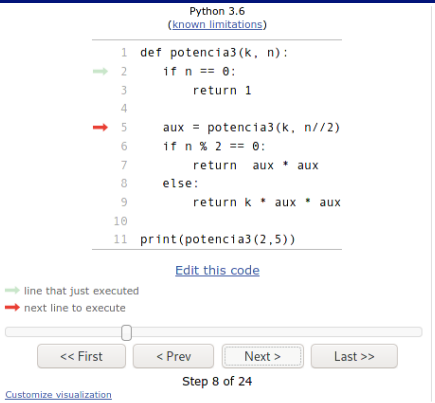


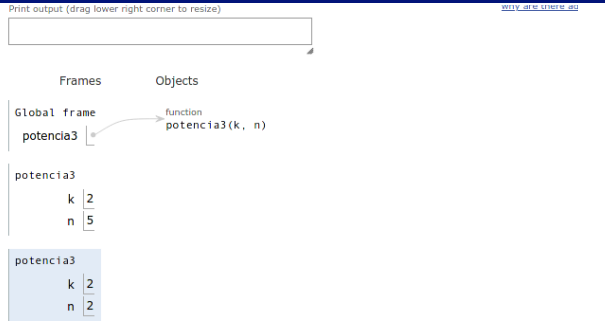


$$k^n = \begin{cases} 1, & \text{se } n = 0 \\ k^{n/2} \times k^{n/2}, & \text{se } n \text{ for positivo e par} \\ k \times k^{\lfloor n/2 \rfloor} \times k^{\lfloor n/2 \rfloor}, & \text{se } n \text{ for positivo e ímpar} \end{cases}$$

https://pythontutor.com

https://tinyurl.com/mxhbmhd8

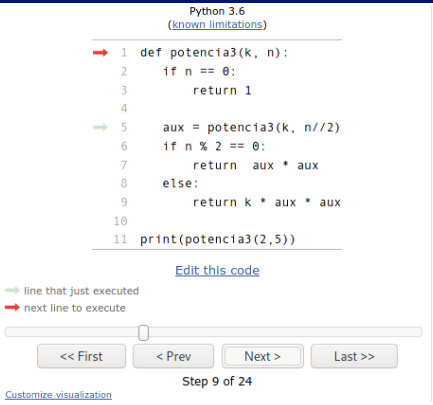


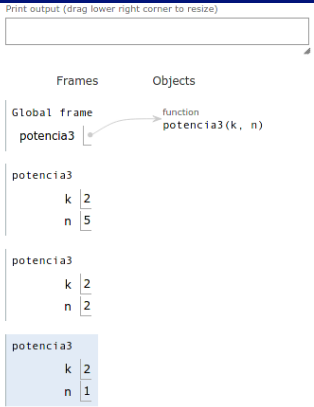


$$k^n = \begin{cases} 1, & \text{se } n = 0 \\ k^{n/2} \times k^{n/2}, & \text{se } n \text{ for positivo e par} \\ k \times k^{\lfloor n/2 \rfloor} \times k^{\lfloor n/2 \rfloor}, & \text{se } n \text{ for positivo e ímpar} \end{cases}$$

https://pythontutor.com

https://tinyurl.com/mxhbmhd8

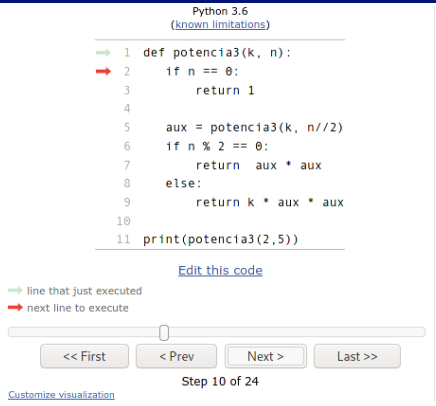


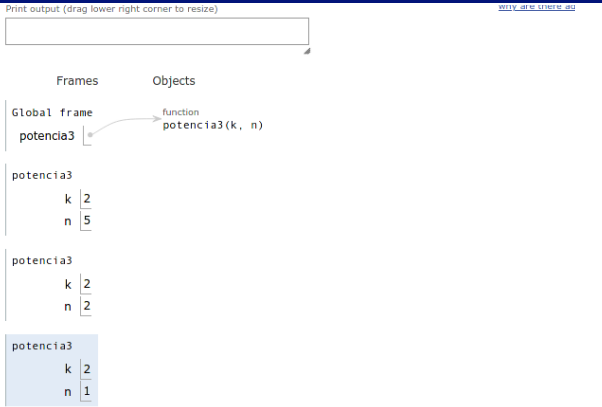


$$k^n = \begin{cases} 1, & \text{se } n = 0 \\ k^{n/2} \times k^{n/2}, & \text{se } n \text{ for positivo e par} \\ k \times k^{\lfloor n/2 \rfloor} \times k^{\lfloor n/2 \rfloor}, & \text{se } n \text{ for positivo e ímpar} \end{cases}$$

https://pythontutor.com

https://tinyurl.com/mxhbmhd8

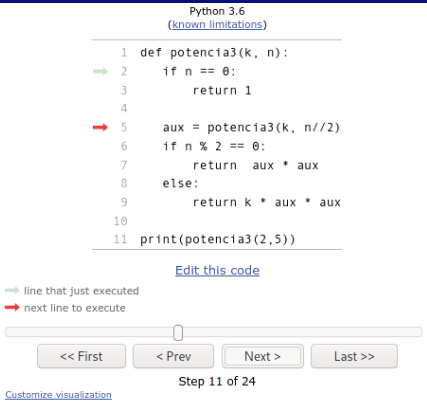


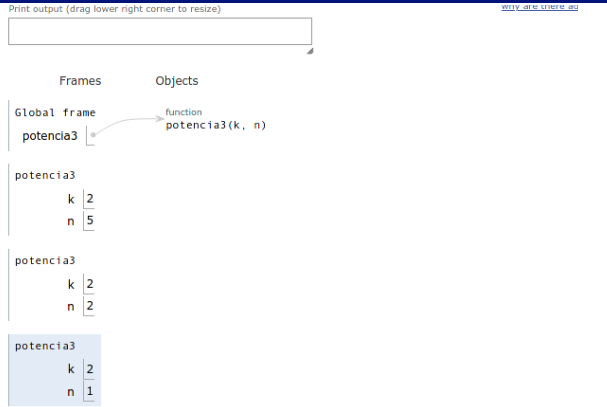


$$k^n = \begin{cases} 1, & \text{se } n = 0 \\ k^{n/2} \times k^{n/2}, & \text{se } n \text{ for positivo e par} \\ k \times k^{\lfloor n/2 \rfloor} \times k^{\lfloor n/2 \rfloor}, & \text{se } n \text{ for positivo e ímpar} \end{cases}$$

https://pythontutor.com

https://tinyurl.com/mxhbmhd8

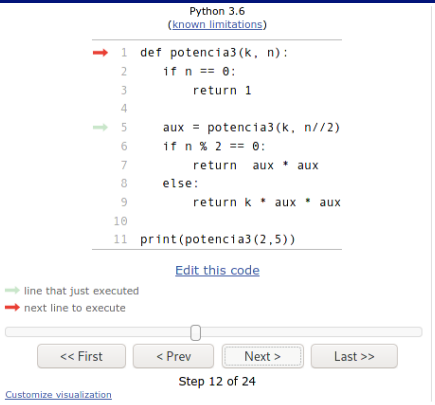


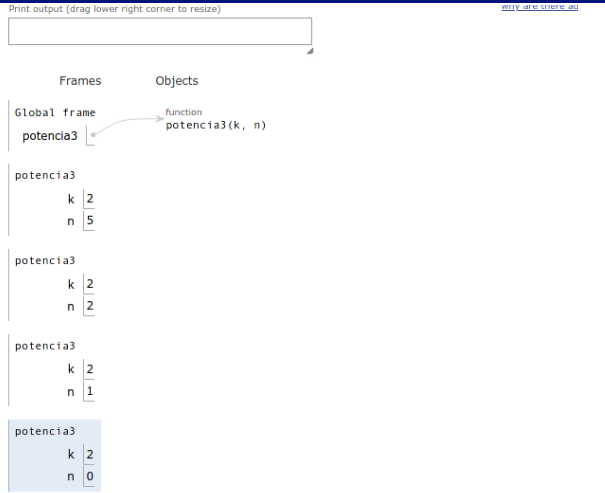


$$k^n = \begin{cases} 1, & \text{se } n = 0 \\ k^{n/2} \times k^{n/2}, & \text{se } n \text{ for positivo e par} \\ k \times k^{\lfloor n/2 \rfloor} \times k^{\lfloor n/2 \rfloor}, & \text{se } n \text{ for positivo e ímpar} \end{cases}$$

https://pythontutor.com

https://tinyurl.com/mxhbmhd8

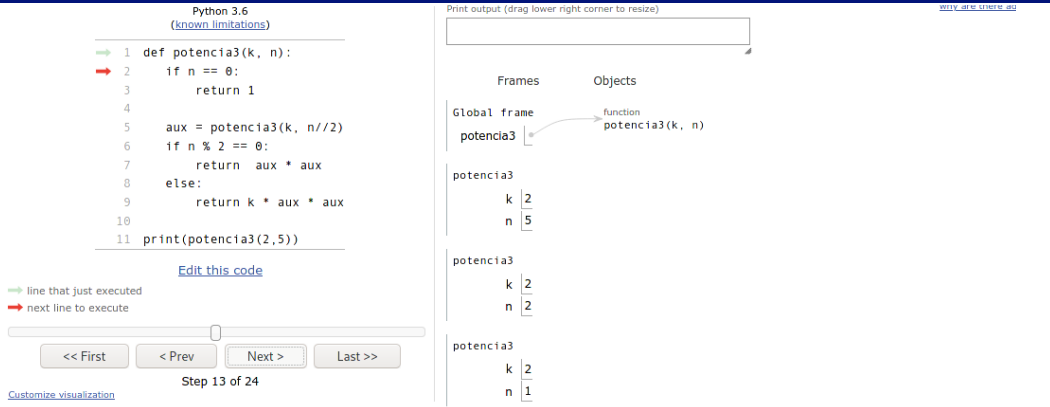




$$k^n = \begin{cases} 1, & \text{se } n = 0 \\ k^{n/2} \times k^{n/2}, & \text{se } n \text{ for positivo e par} \\ k \times k^{\lfloor n/2 \rfloor} \times k^{\lfloor n/2 \rfloor}, & \text{se } n \text{ for positivo e ímpar} \end{cases}$$

https://pythontutor.com

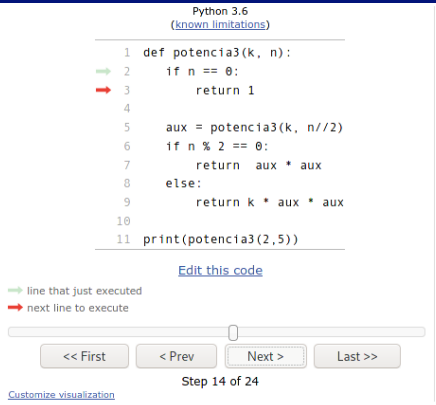
https://tinyurl.com/mxhbmhd8

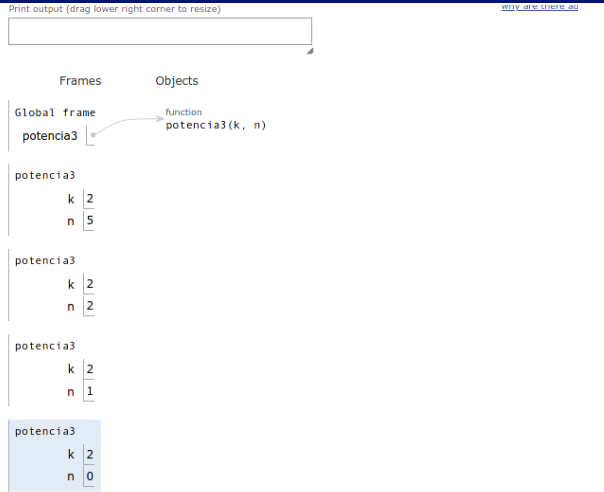


$$k^n = \begin{cases} 1, & \text{se } n = 0 \\ k^{n/2} \times k^{n/2}, & \text{se } n \text{ for positivo e par} \\ k \times k^{\lfloor n/2 \rfloor} \times k^{\lfloor n/2 \rfloor}, & \text{se } n \text{ for positivo e ímpar} \end{cases}$$

https://pythontutor.com

https://tinyurl.com/mxhbmhd8

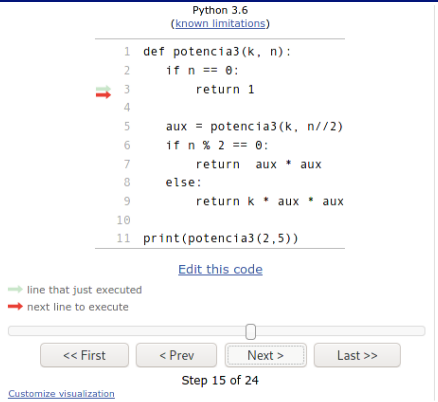




$$k^n = \begin{cases} 1, & \text{se } n = 0 \\ k^{n/2} \times k^{n/2}, & \text{se } n \text{ for positivo e par} \\ k \times k^{\lfloor n/2 \rfloor} \times k^{\lfloor n/2 \rfloor}, & \text{se } n \text{ for positivo e ímpar} \end{cases}$$

https://pythontutor.com

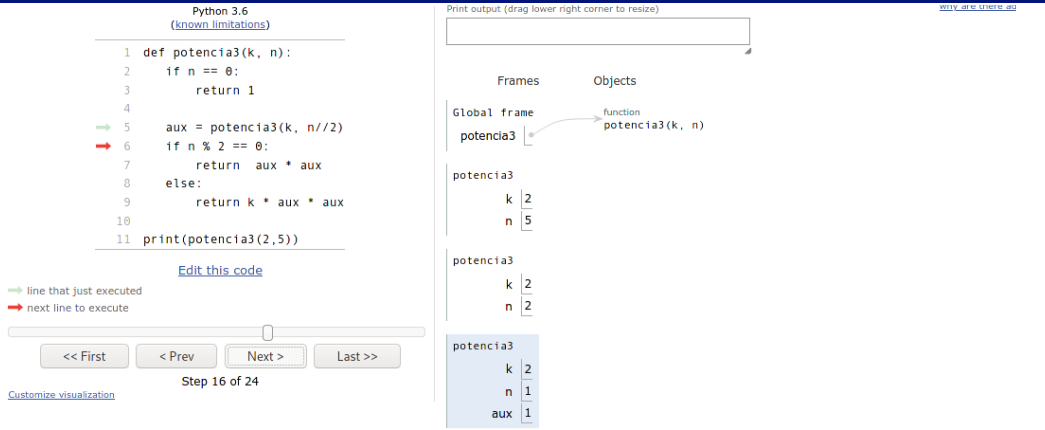
https://tinyurl.com/mxhbmhd8



$$k^n = \begin{cases} 1, & \text{se } n = 0 \\ k^{n/2} \times k^{n/2}, & \text{se } n \text{ for positivo e par} \\ k \times k^{\lfloor n/2 \rfloor} \times k^{\lfloor n/2 \rfloor}, & \text{se } n \text{ for positivo e ímpar} \end{cases}$$

https://pythontutor.com

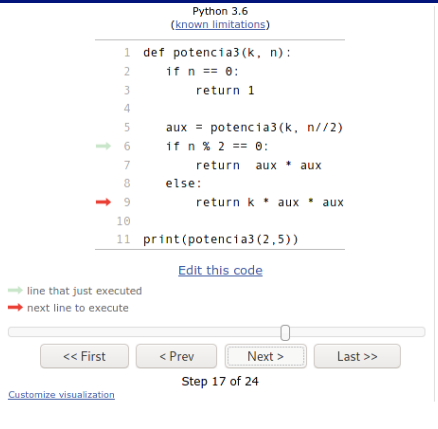
https://tinyurl.com/mxhbmhd8

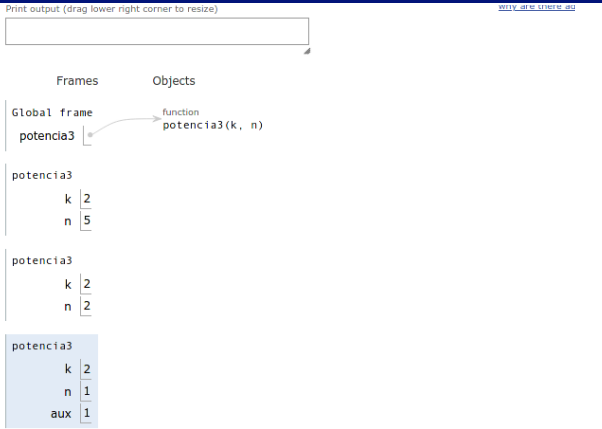


$$k^n = \begin{cases} 1, & \text{se } n = 0 \\ k^{n/2} \times k^{n/2}, & \text{se } n \text{ for positivo e par} \\ k \times k^{\lfloor n/2 \rfloor} \times k^{\lfloor n/2 \rfloor}, & \text{se } n \text{ for positivo e ímpar} \end{cases}$$

https://pythontutor.com

https://tinyurl.com/mxhbmhd8

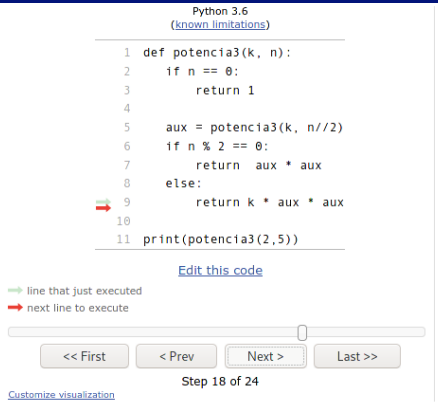


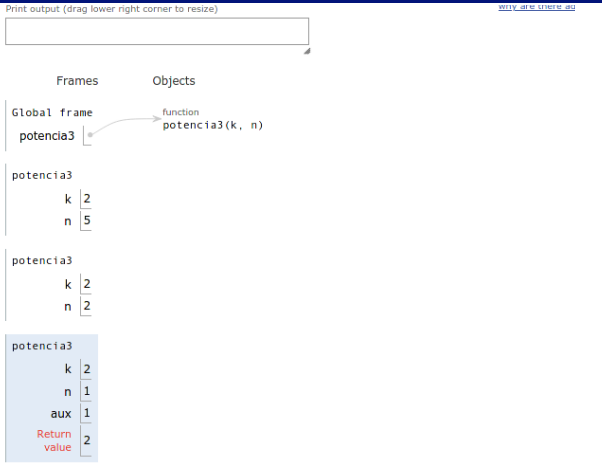


$$k^n = \begin{cases} 1, & \text{se } n = 0 \\ k^{n/2} \times k^{n/2}, & \text{se } n \text{ for positivo e par} \\ k \times k^{\lfloor n/2 \rfloor} \times k^{\lfloor n/2 \rfloor}, & \text{se } n \text{ for positivo e ímpar} \end{cases}$$

https://pythontutor.com

https://tinyurl.com/mxhbmhd8

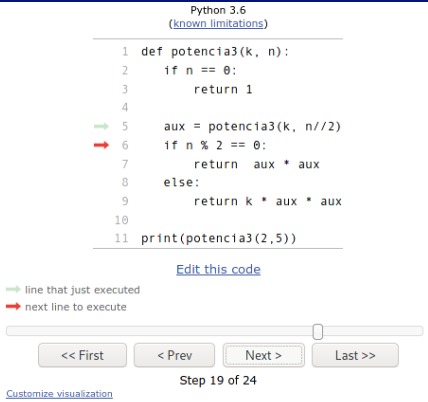


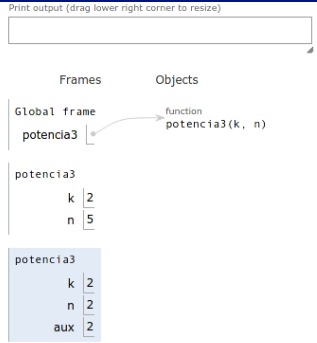


$$k^n = \begin{cases} 1, & \text{se } n = 0 \\ k^{n/2} \times k^{n/2}, & \text{se } n \text{ for positivo e par} \\ k \times k^{\lfloor n/2 \rfloor} \times k^{\lfloor n/2 \rfloor}, & \text{se } n \text{ for positivo e ímpar} \end{cases}$$

https://pythontutor.com

https://tinyurl.com/mxhbmhd8

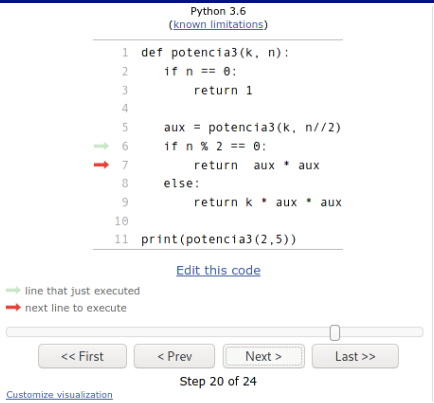


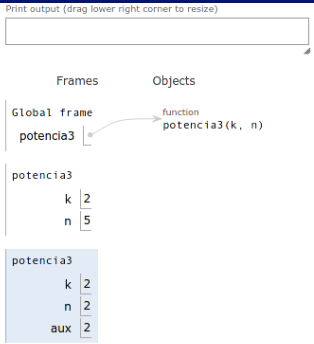


$$k^n = \begin{cases} 1, & \text{se } n = 0 \\ k^{n/2} \times k^{n/2}, & \text{se } n \text{ for positivo e par} \\ k \times k^{\lfloor n/2 \rfloor} \times k^{\lfloor n/2 \rfloor}, & \text{se } n \text{ for positivo e ímpar} \end{cases}$$

https://pythontutor.com

https://tinyurl.com/mxhbmhd8

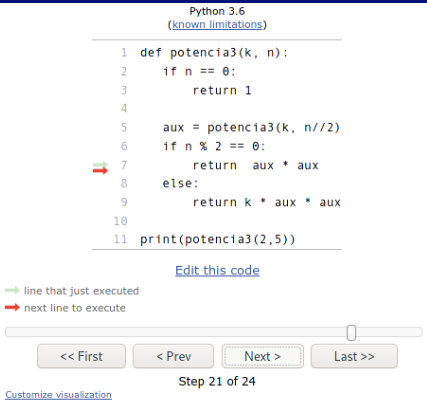


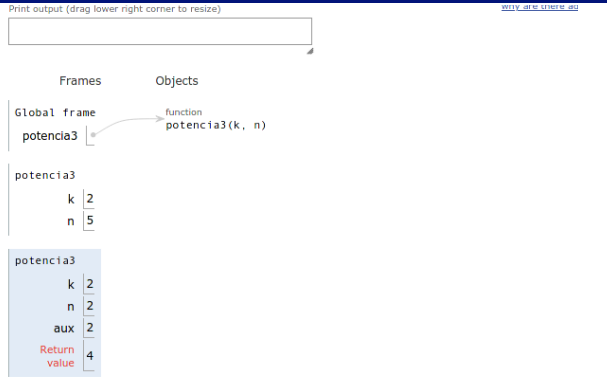


$$k^n = \begin{cases} 1, & \text{se } n = 0 \\ k^{n/2} \times k^{n/2}, & \text{se } n \text{ for positivo e par} \\ k \times k^{\lfloor n/2 \rfloor} \times k^{\lfloor n/2 \rfloor}, & \text{se } n \text{ for positivo e ímpar} \end{cases}$$

https://pythontutor.com

https://tinyurl.com/mxhbmhd8

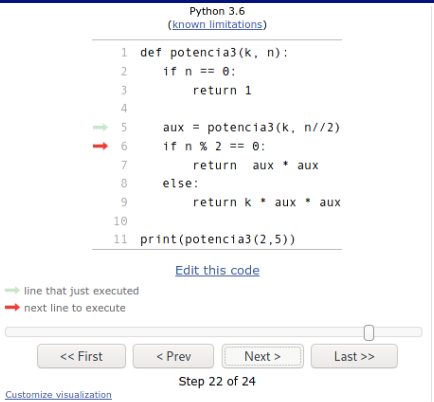




$$k^n = \begin{cases} 1, & \text{se } n = 0 \\ k^{n/2} \times k^{n/2}, & \text{se } n \text{ for positivo e par} \\ k \times k^{\lfloor n/2 \rfloor} \times k^{\lfloor n/2 \rfloor}, & \text{se } n \text{ for positivo e ímpar} \end{cases}$$

https://pythontutor.com

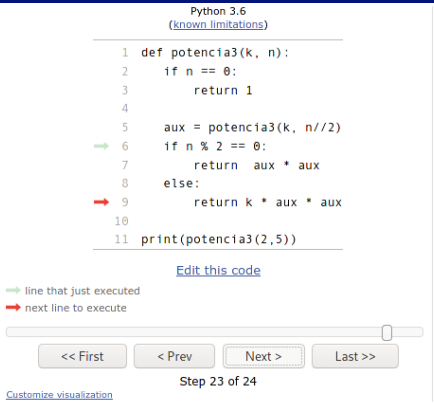
https://tinyurl.com/mxhbmhd8

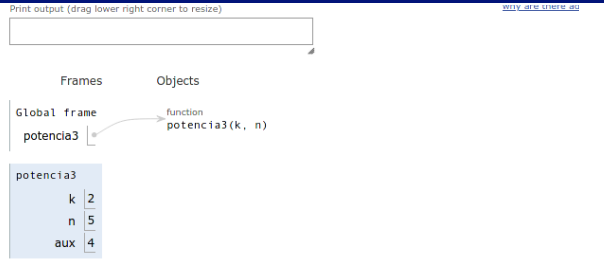


$$k^n = \begin{cases} 1, & \text{se } n = 0 \\ k^{n/2} \times k^{n/2}, & \text{se } n \text{ for positivo e par} \\ k \times k^{\lfloor n/2 \rfloor} \times k^{\lfloor n/2 \rfloor}, & \text{se } n \text{ for positivo e ímpar} \end{cases}$$

https://pythontutor.com

https://tinyurl.com/mxhbmhd8

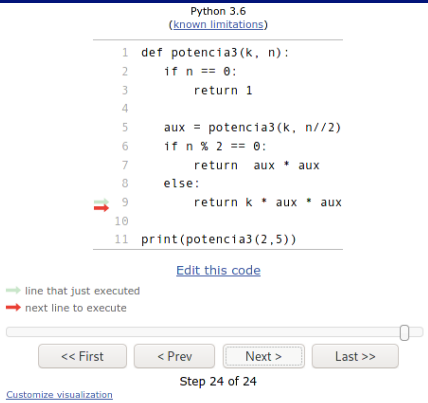


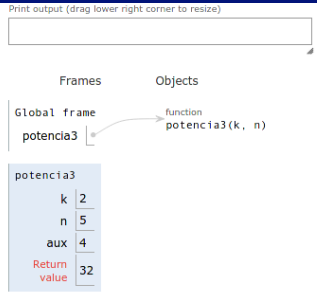


$$k^n = \begin{cases} 1, & \text{se } n = 0 \\ k^{n/2} \times k^{n/2}, & \text{se } n \text{ for positivo e par} \\ k \times k^{\lfloor n/2 \rfloor} \times k^{\lfloor n/2 \rfloor}, & \text{se } n \text{ for positivo e ímpar} \end{cases}$$

https://pythontutor.com

https://tinyurl.com/mxhbmhd8





$$k^n = \begin{cases} 1, & \text{se } n = 0 \\ k^{n/2} \times k^{n/2}, & \text{se } n \text{ for positivo e par} \\ k \times k^{\lfloor n/2 \rfloor} \times k^{\lfloor n/2 \rfloor}, & \text{se } n \text{ for positivo e ímpar} \end{cases}$$

https://pythontutor.com

https://tinyurl.com/mxhbmhd8

Python 3.6
(known limitations)

```
1  def potencia3(k, n):
2      if n == 0:
3          return 1
4
5      aux = potencia3(k, n//2)
6      if n % 2 == 0:
7          return  aux * aux
8      else:
9          return k * aux * aux
10
11 print(potencia3(2,5))
```

Edit this code

line that just executed
next line to execute

<< First | < Prev | Next > | Last >>

Done running (24 steps)

Customize visualization

Print output (drag lower right corner to resize)

```
32
```

Frames Objects

Global frame
potencia3

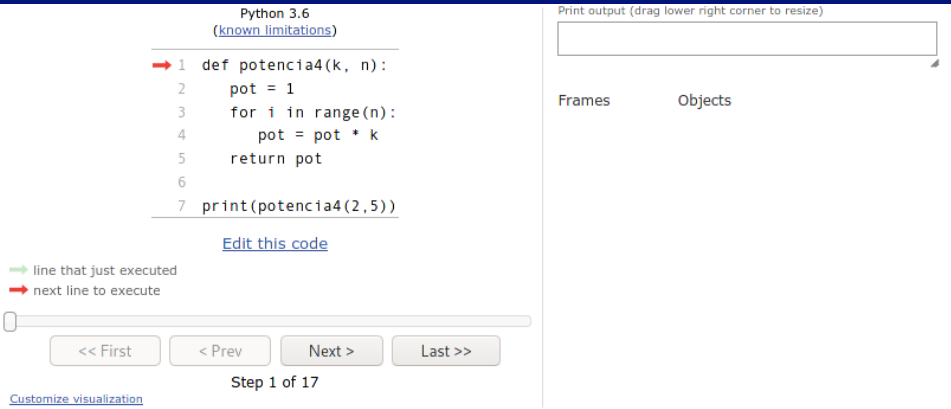
function
potencia3(k, n)

$$k^n = \begin{cases} 1, & \text{se } n = 0 \\ k^{n/2} \times k^{n/2}, & \text{se } n \text{ for positivo e par} \\ k \times k^{\lfloor n/2 \rfloor} \times k^{\lfloor n/2 \rfloor}, & \text{se } n \text{ for positivo e ímpar} \end{cases}$$

https://pythontutor.com
https://tinyurl.com/mxhbmhd8

Implementação 4

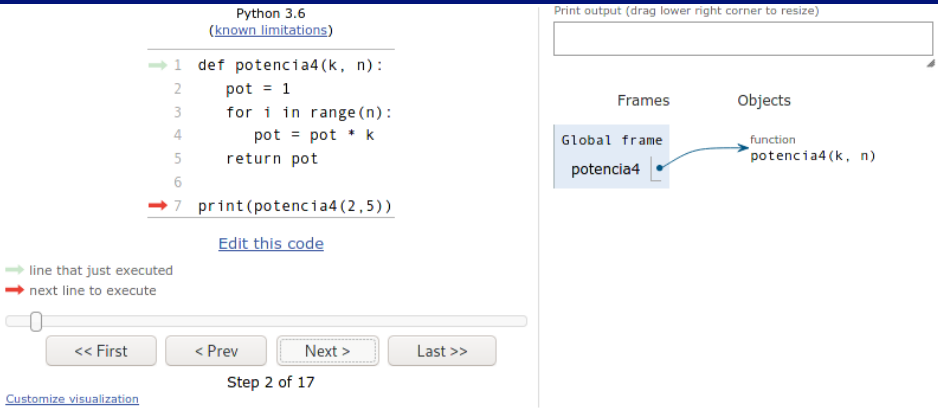
$$k^n = \prod_{i=1}^{n} k = \underbrace{k \times k \times \ldots \times k}_{n \text{ vezes}}$$



$$k^n = \prod_{i=1}^{n} k = \underbrace{k \times k \times \ldots \times k}_{n \text{ vezes}}$$

https://pythontutor.com

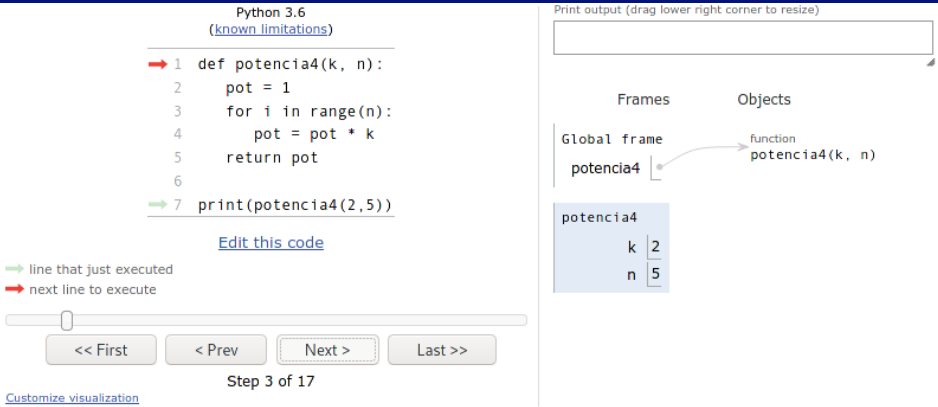
https://tinyurl.com/ms2zpcry



$$k^n = \prod_{i=1}^{n} k = \underbrace{k \times k \times \ldots \times k}_{n \text{ vezes}}$$

https://pythontutor.com

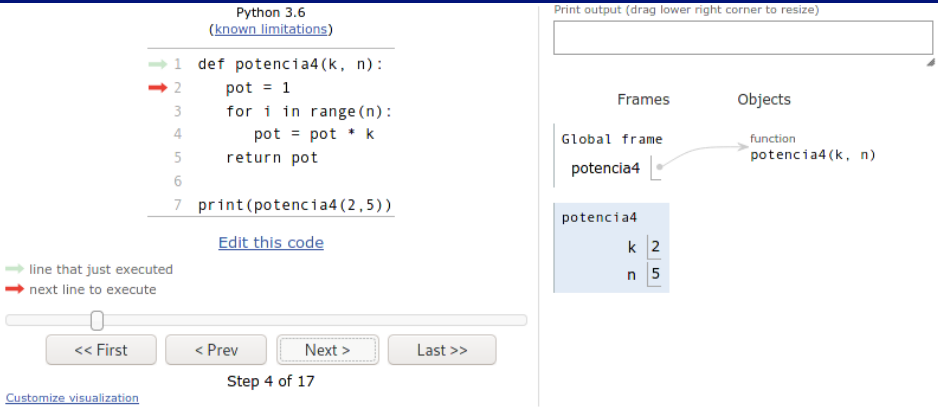
https://tinyurl.com/ms2zpcry



$$k^n = \prod_{i=1}^{n} k = \underbrace{k \times k \times \ldots \times k}_{n \text{ vezes}}$$

https://pythontutor.com

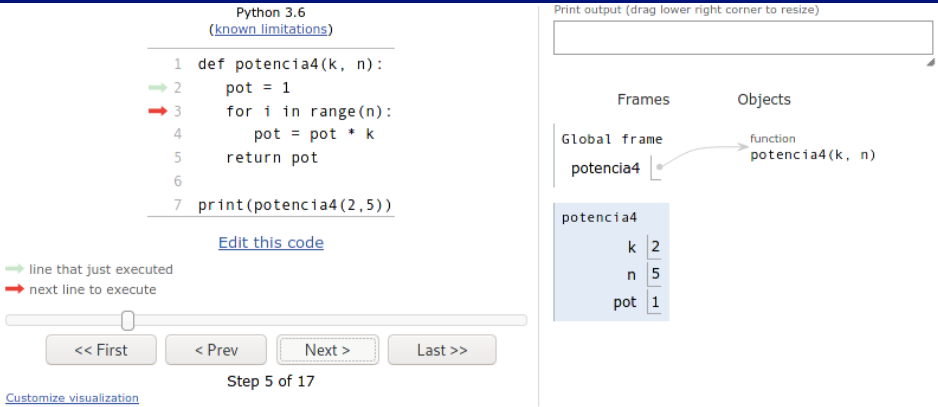
https://tinyurl.com/ms2zpcry



$$k^n = \prod_{i=1}^{n} k = \underbrace{k \times k \times \ldots \times k}_{n \text{ vezes}}$$

https://pythontutor.com

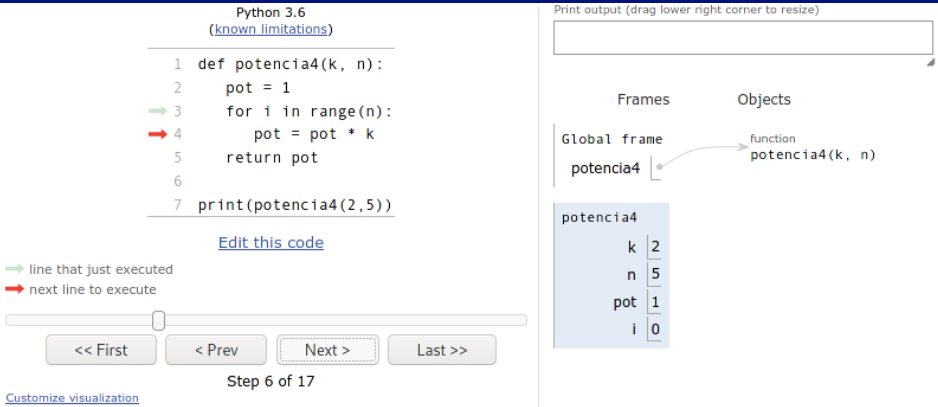
https://tinyurl.com/ms2zpcry



$$k^n = \prod_{i=1}^{n} k = \underbrace{k \times k \times \ldots \times k}_{n \text{ vezes}}$$

https://pythontutor.com

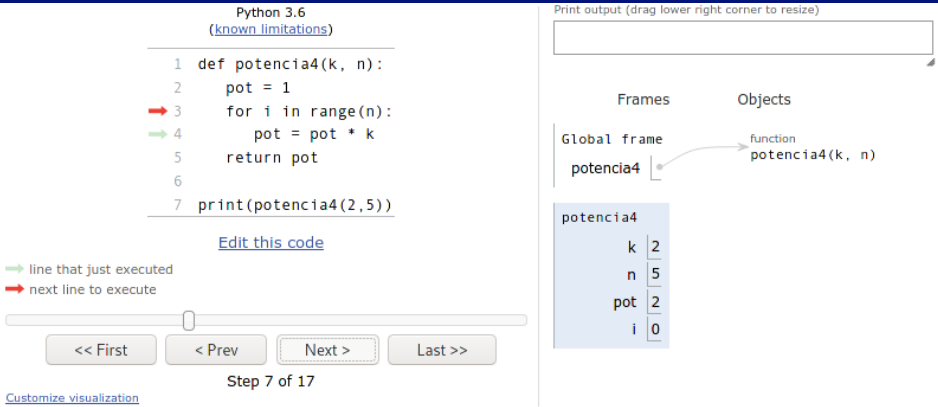
https://tinyurl.com/ms2zpcry



$$k^n = \prod_{i=1}^{n} k = \underbrace{k \times k \times \ldots \times k}_{n \text{ vezes}}$$

https://pythontutor.com

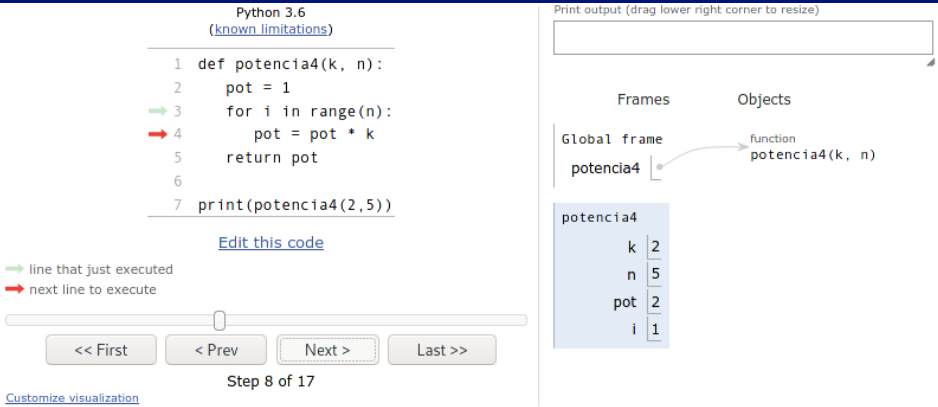
https://tinyurl.com/ms2zpcry



$$k^n = \prod_{i=1}^{n} k = \underbrace{k \times k \times \ldots \times k}_{n \text{ vezes}}$$

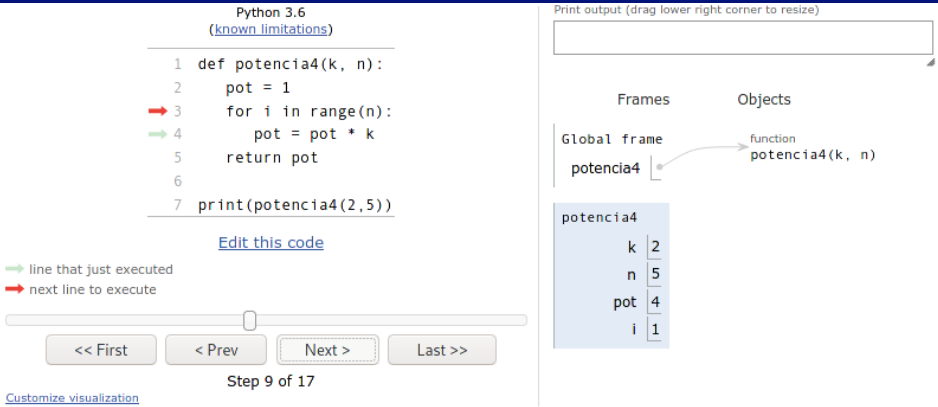
https://pythontutor.com

https://tinyurl.com/ms2zpcry



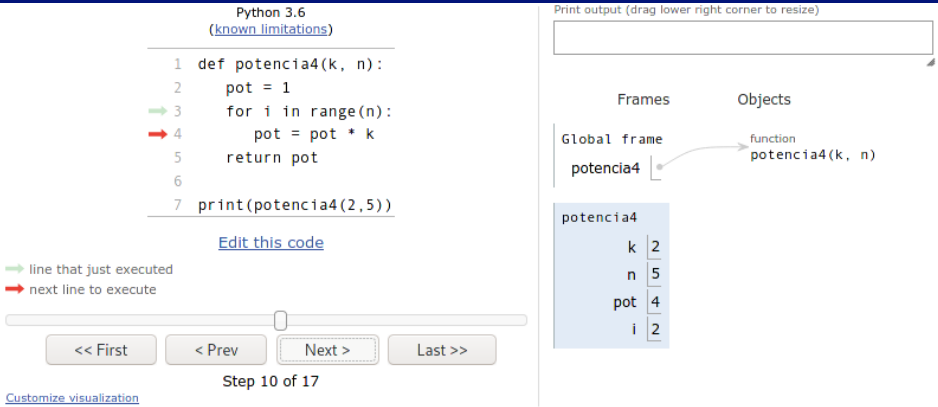
$$k^n = \prod_{i=1}^{n} k = \underbrace{k \times k \times \ldots \times k}_{n \text{ vezes}}$$

https://pythontutor.com
https://tinyurl.com/ms2zpcry



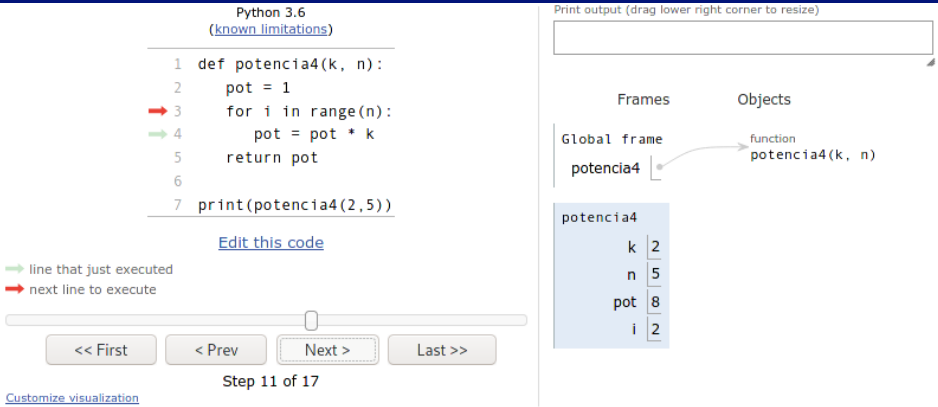
$$k^n = \prod_{i=1}^{n} k = \underbrace{k \times k \times \ldots \times k}_{n \text{ vezes}}$$

https://pythontutor.com
https://tinyurl.com/ms2zpcry



$$k^n = \prod_{i=1}^{n} k = \underbrace{k \times k \times \ldots \times k}_{n \text{ vezes}}$$

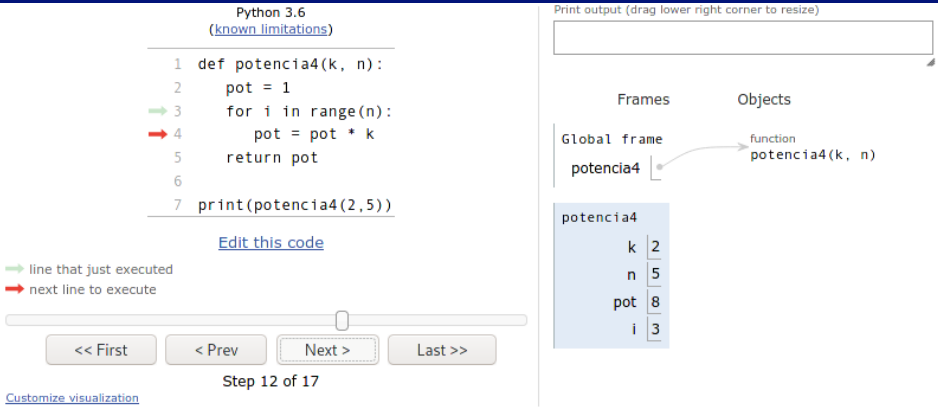
https://pythontutor.com
https://tinyurl.com/ms2zpcry



$$k^n = \prod_{i=1}^{n} k = \underbrace{k \times k \times \ldots \times k}_{n \text{ vezes}}$$

https://pythontutor.com

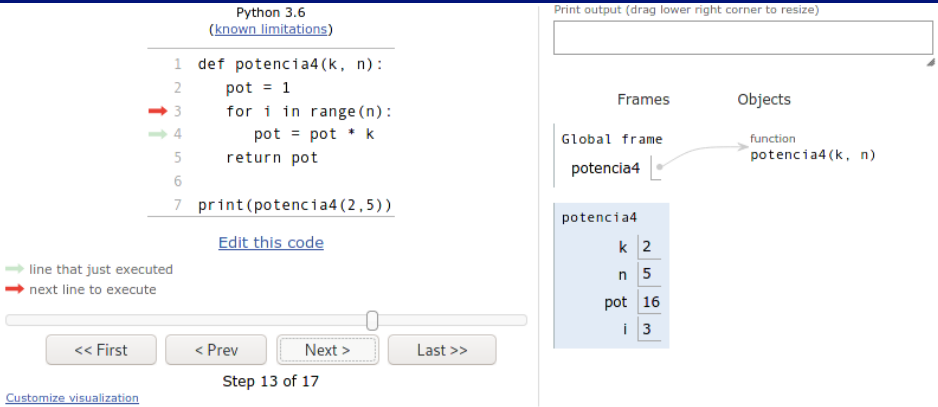
https://tinyurl.com/ms2zpcry



$$k^n = \prod_{i=1}^{n} k = \underbrace{k \times k \times \ldots \times k}_{n \text{ vezes}}$$

https://pythontutor.com

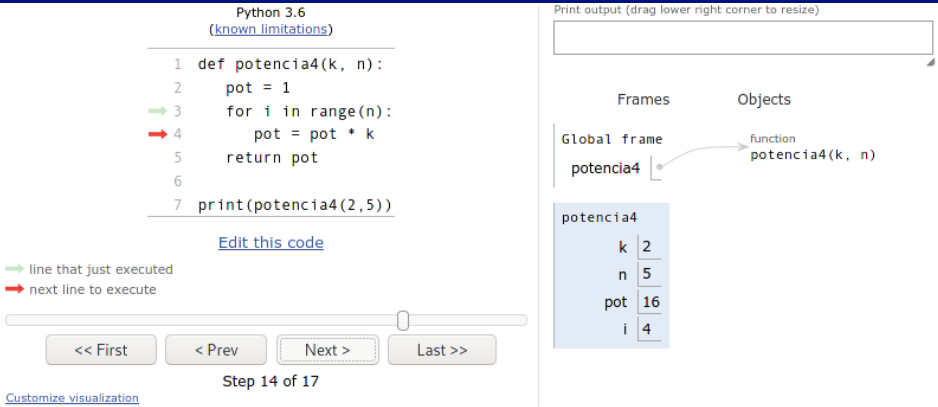
https://tinyurl.com/ms2zpcry



$$k^n = \prod_{i=1}^{n} k = \underbrace{k \times k \times \ldots \times k}_{n \text{ vezes}}$$

https://pythontutor.com

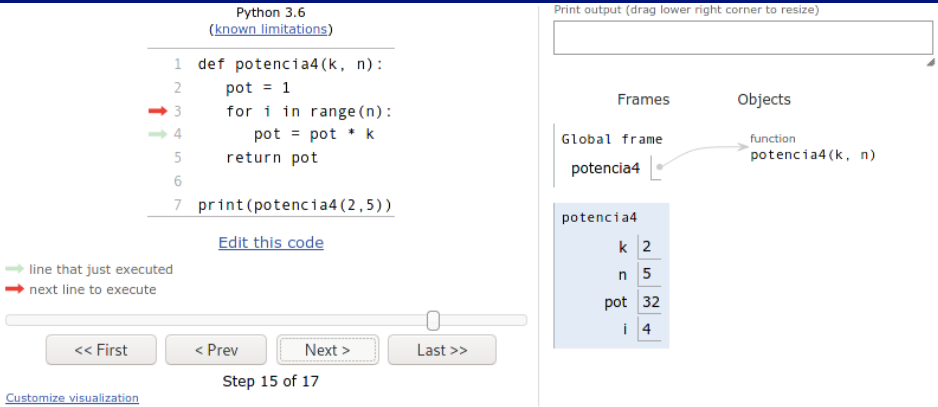
https://tinyurl.com/ms2zpcry



$$k^n = \prod_{i=1}^{n} k = \underbrace{k \times k \times \ldots \times k}_{n \text{ vezes}}$$

https://pythontutor.com

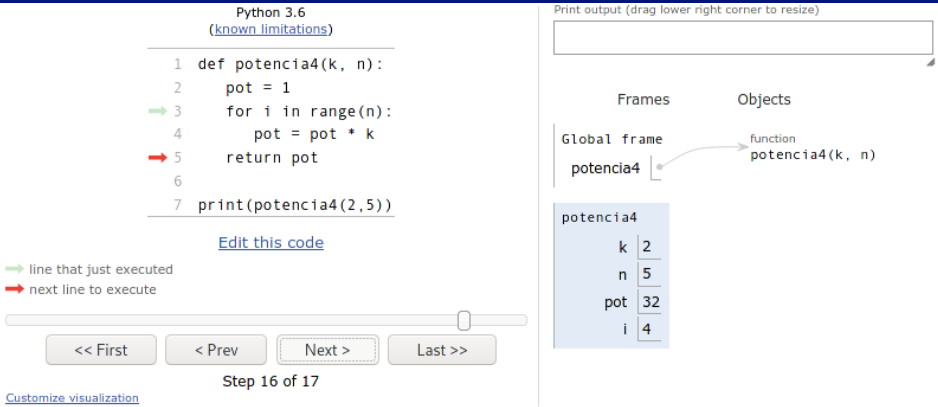
https://tinyurl.com/ms2zpcry



$$k^n = \prod_{i=1}^{n} k = \underbrace{k \times k \times \ldots \times k}_{n \text{ vezes}}$$

https://pythontutor.com

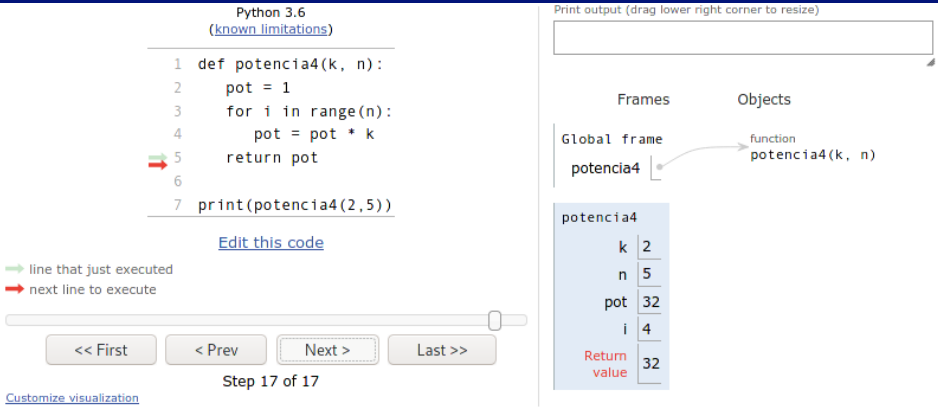
https://tinyurl.com/ms2zpcry



$$k^n = \prod_{i=1}^{n} k = \underbrace{k \times k \times \ldots \times k}_{n \text{ vezes}}$$

https://pythontutor.com

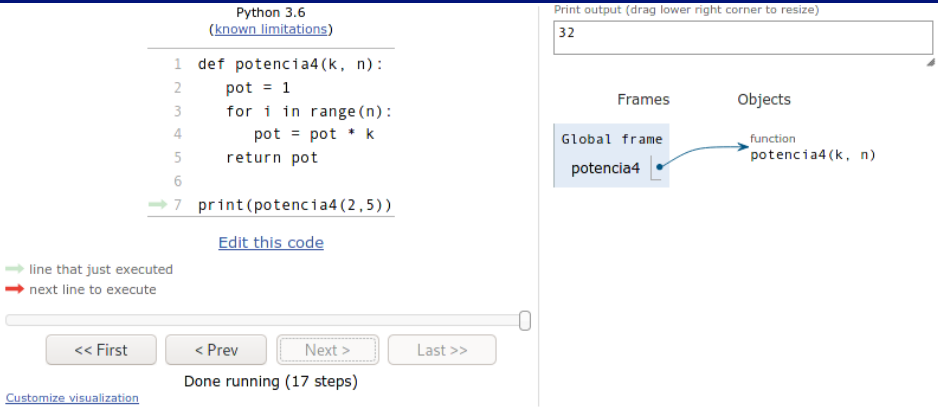
https://tinyurl.com/ms2zpcry



$$k^n = \prod_{i=1}^{n} k = \underbrace{k \times k \times \ldots \times k}_{n \text{ vezes}}$$

https://pythontutor.com

https://tinyurl.com/ms2zpcry



$$k^n = \prod_{i=1}^{n} k = \underbrace{k \times k \times \ldots \times k}_{n \text{ vezes}}$$

https://pythontutor.com
https://tinyurl.com/ms2zpcry

Comparando as implementações

```
c1=c2=c3=c4=0

def potencia1(k, n):
   global c1
   c1+=1
   if n == 0:
       return 1
   else:
      return k * potencia1(k, n - 1)

def potencia2(k, n):
   global c2
   c2+=1
   if n == 0:
       return 1
   elif n % 2 == 0:
       return potencia2(k, n//2) * potencia2(k, n//2)
   else:
       return k * potencia2(k, n//2) * potencia2(k, n//2)

def potencia3(k, n):
   global c3
   c3+=1
   if n == 0:
       return 1
   aux = potencia3(k, n//2)
   if n % 2 == 0:
       return  aux * aux
   else:
       return k * aux * aux

def potencia4(k, n):
   global c4
   pot = 1
   for i in range(n):
      c4+=1
      pot = pot * k
   return pot
```

```
k=2
for n in range(20):
   c1=c2=c3=c4=0
   potencia1(k,n)
   potencia2(k,n)
   potencia3(k,n)
   potencia4(k,n)
   print("{} \t {} \t {} \t {} \t {}".format(n,c1,c2,c3,c4))


print()
k=2
for n in range(100,1000,100):
   c1=c2=c3=c4=0
   potencia1(k,n)
   potencia2(k,n)
   potencia3(k,n)
   potencia4(k,n)
   print("{} \t {} \t {} \t {} \t {}".format(n,c1,c2,c3,c4))

print()
k=2
for n in range(1000,10000,1000):
   c1=c2=c3=c4=0
   potencia1(k,n)
   potencia2(k,n)
   potencia3(k,n)
   potencia4(k,n)
   print("{} \t {} \t {} \t {} \t {}".format(n,c1,c2,c3,c4))
```

```
$ python3 compara_exp.py
0       1       1       1       0
1       2       3       2       1
2       3       7       3       2
3       4       7       3       3
4       5       15      4       4
5       6       15      4       5
6       7       15      4       6
7       8       15      4       7
8       9       31      5       8
9       10      31      5       9
10      11      31      5       10
11      12      31      5       11
12      13      31      5       12
13      14      31      5       13
14      15      31      5       14
15      16      31      5       15
16      17      63      6       16
17      18      63      6       17
18      19      63      6       18
19      20      63      6       19

100     101     255     8       100
200     201     511     9       200
300     301     1023    10      300
400     401     1023    10      400
500     501     1023    10      500
600     601     2047    11      600
700     701     2047    11      700
800     801     2047    11      800
900     901     2047    11      900
```

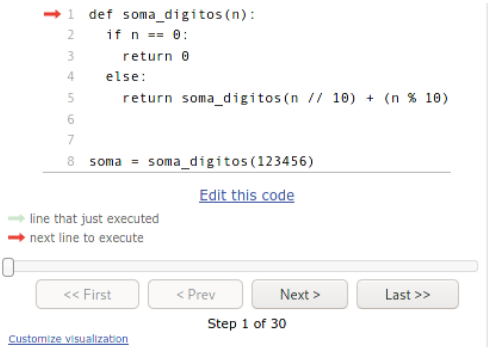
```
Traceback (most recent call last):
  File "compara_exp.py", line 64, in <module>
    potencia1(k,n)
  File "compara_exp.py", line 9, in potencia1
    return k * potencia1(k, n - 1)
  File "compara_exp.py", line 9, in potencia1
    return k * potencia1(k, n - 1)
  File "compara_exp.py", line 9, in potencia1
    return k * potencia1(k, n - 1)
  [Previous line repeated 995 more times]
  File "compara_exp.py", line 6, in potencia1
    if n == 0:
RecursionError: maximum recursion depth exceeded in comparison
$
```

Exemplo 10

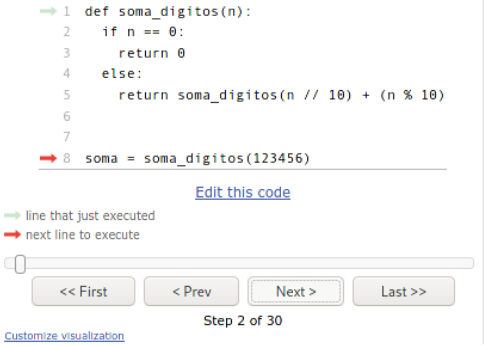
https://ic.unicamp.br/~mc102/aulas/aula12.pdf

- Soma dos dígitos de um número inteiro não negativo.

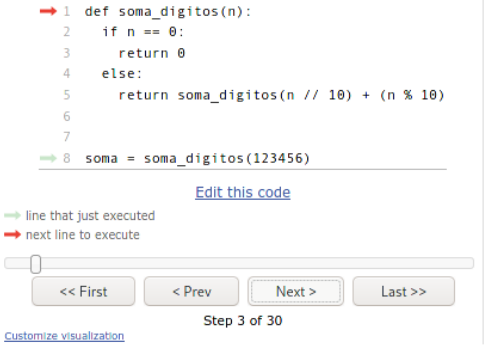
```
1 def soma_digitos(n):
2   if n == 0:
3     return 0
4   else:
5     return soma_digitos(n // 10) + (n % 10)
```



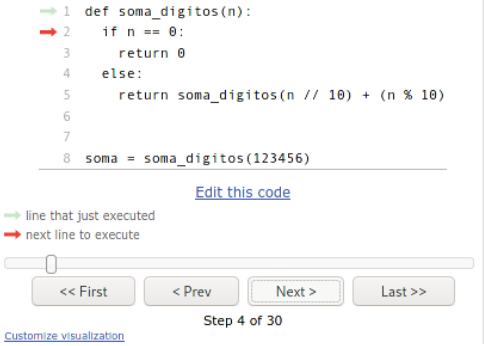
https://pythontutor.com
https://tinyurl.com/4asc5unh



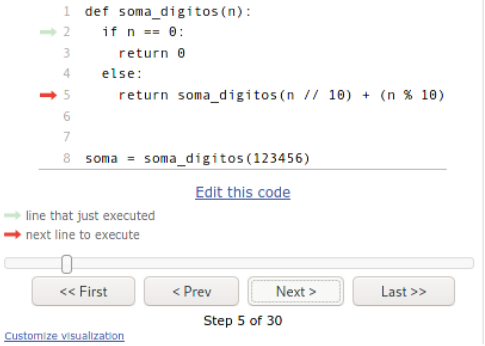
https://pythontutor.com
https://tinyurl.com/4asc5unh



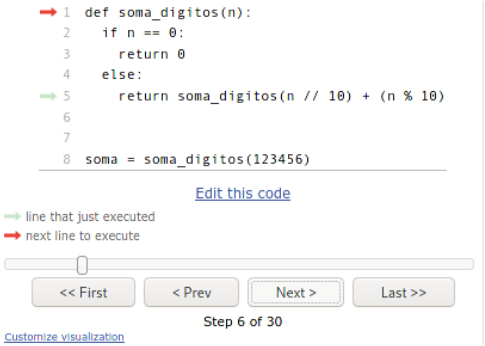
https://pythontutor.com
https://tinyurl.com/4asc5unh

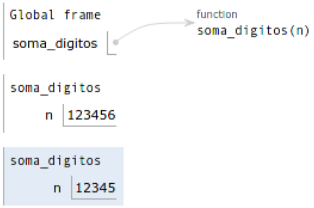


https://pythontutor.com
https://tinyurl.com/4asc5unh

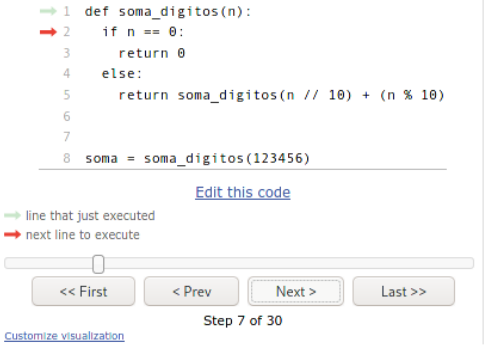


https://pythontutor.com
https://tinyurl.com/4asc5unh

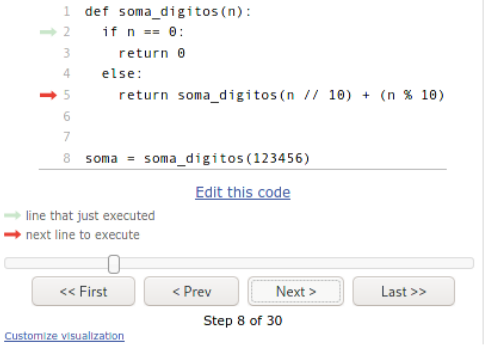




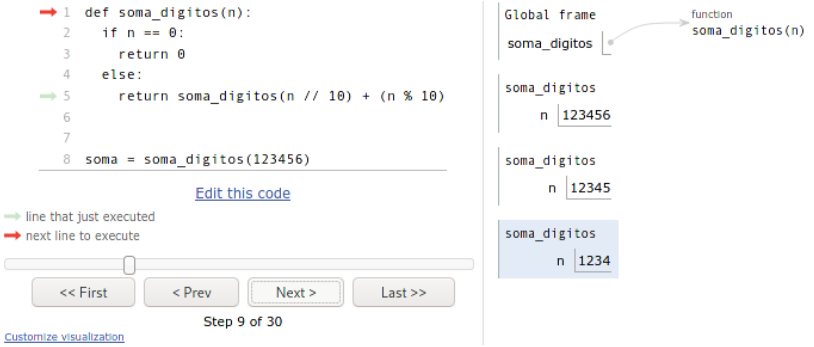
https://pythontutor.com
https://tinyurl.com/4asc5unh



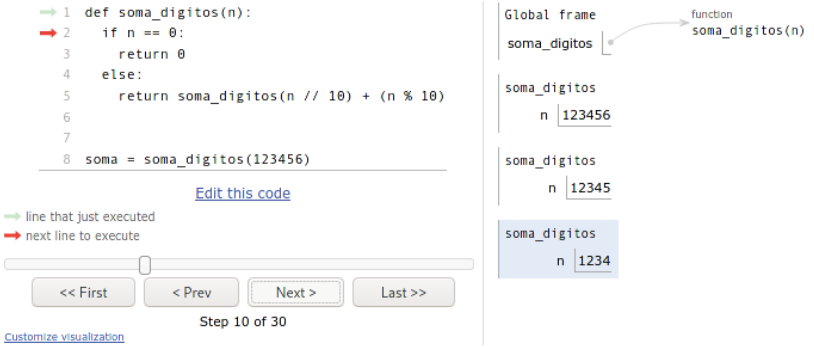
https://pythontutor.com
https://tinyurl.com/4asc5unh



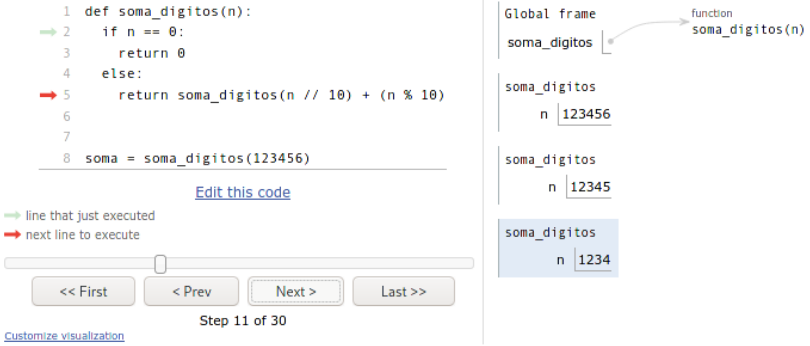
https://pythontutor.com
https://tinyurl.com/4asc5unh



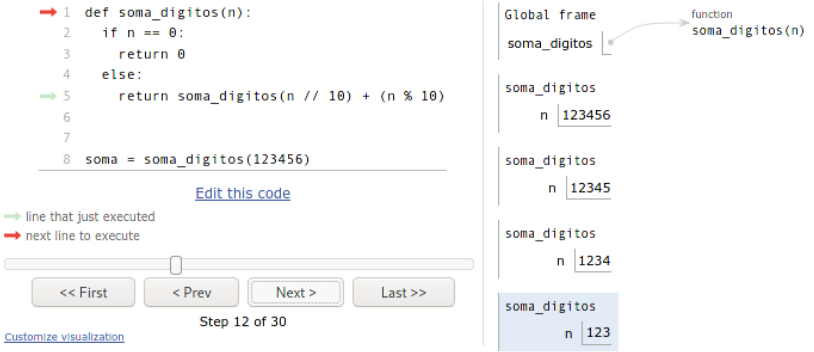
https://pythontutor.com
https://tinyurl.com/4asc5unh



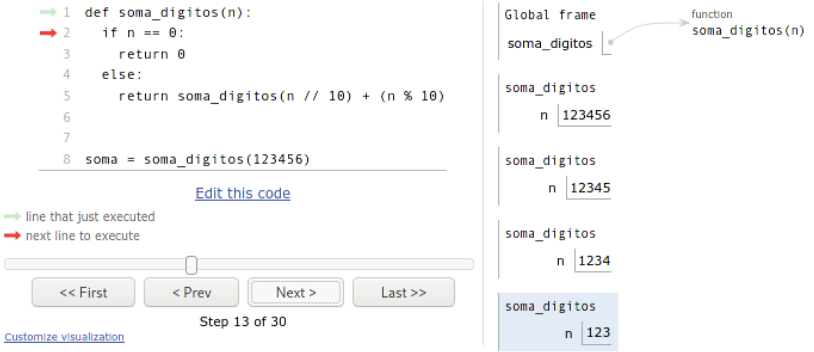
https://pythontutor.com
https://tinyurl.com/4asc5unh



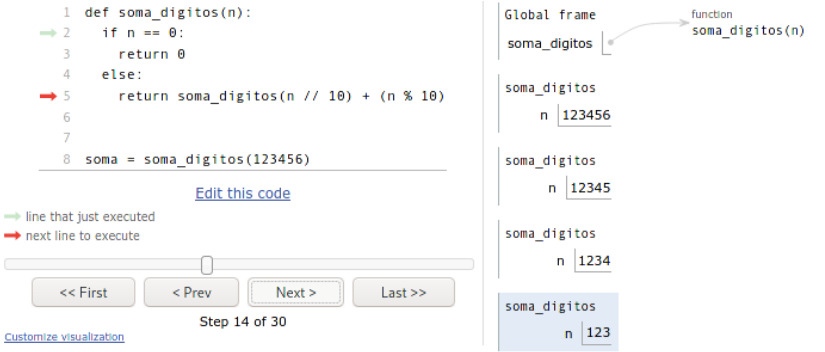
https://pythontutor.com
https://tinyurl.com/4asc5unh



https://pythontutor.com
https://tinyurl.com/4asc5unh

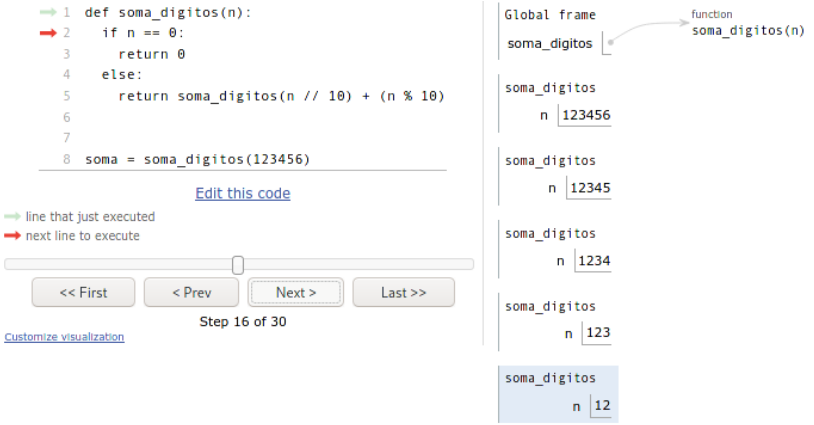


https://pythontutor.com
https://tinyurl.com/4asc5unh

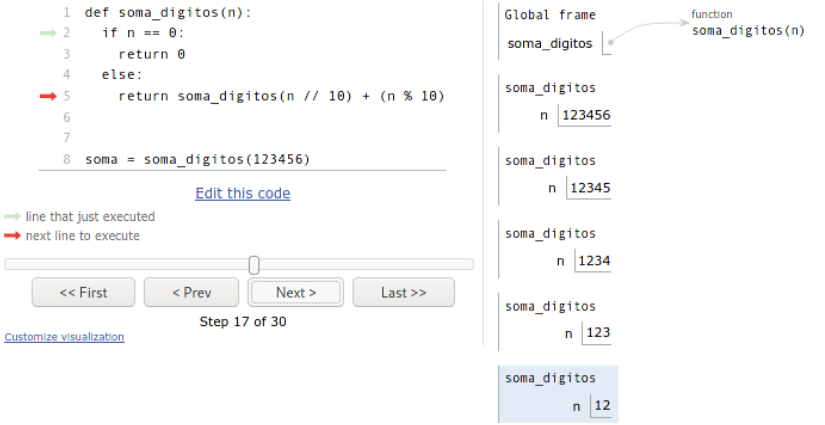


https://pythontutor.com
https://tinyurl.com/4asc5unh

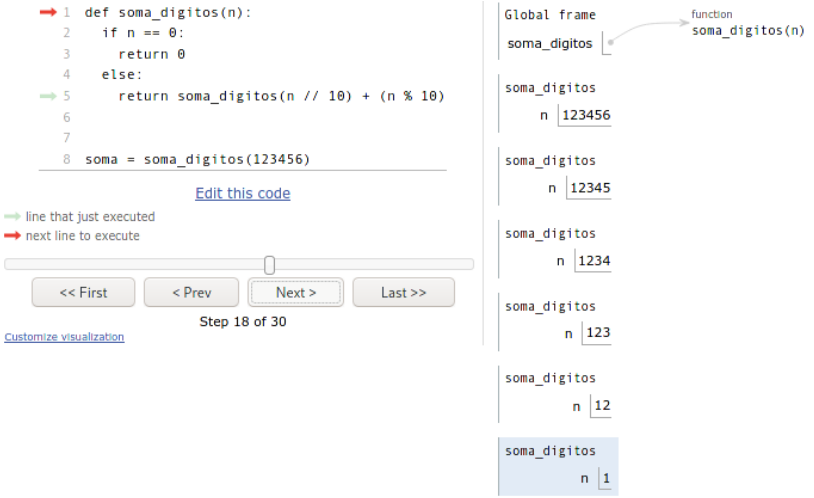
https://pythontutor.com
https://tinyurl.com/4asc5unh



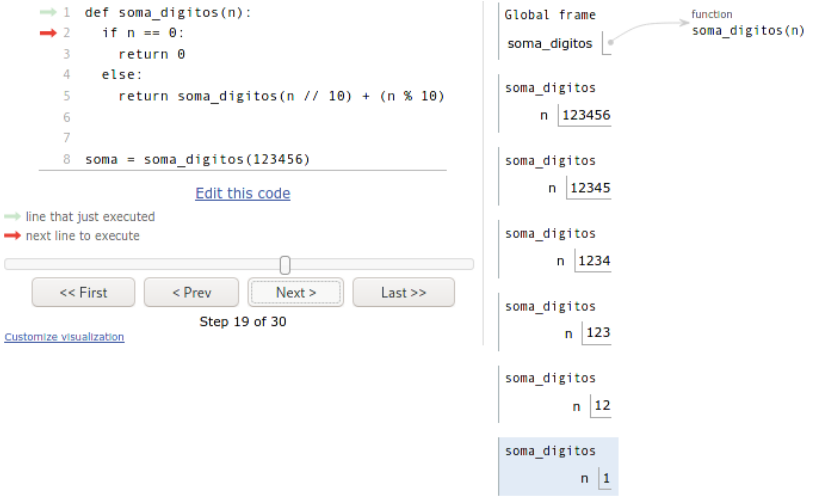
https://pythontutor.com
https://tinyurl.com/4asc5unh



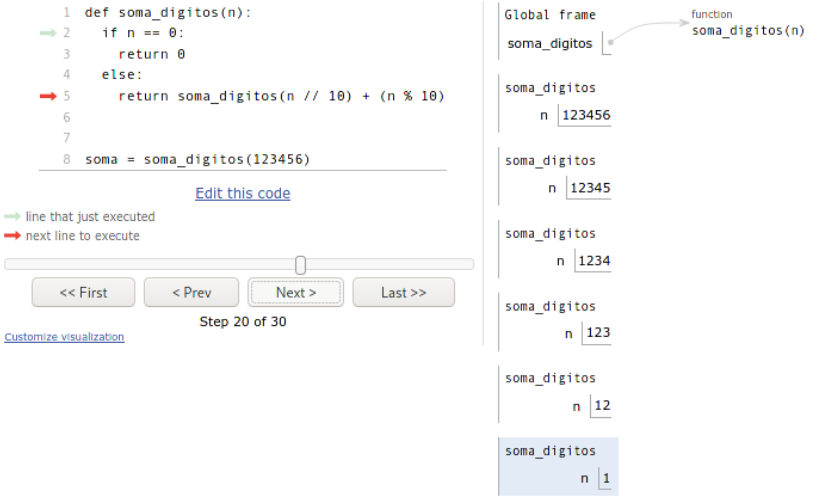
https://pythontutor.com
https://tinyurl.com/4asc5unh



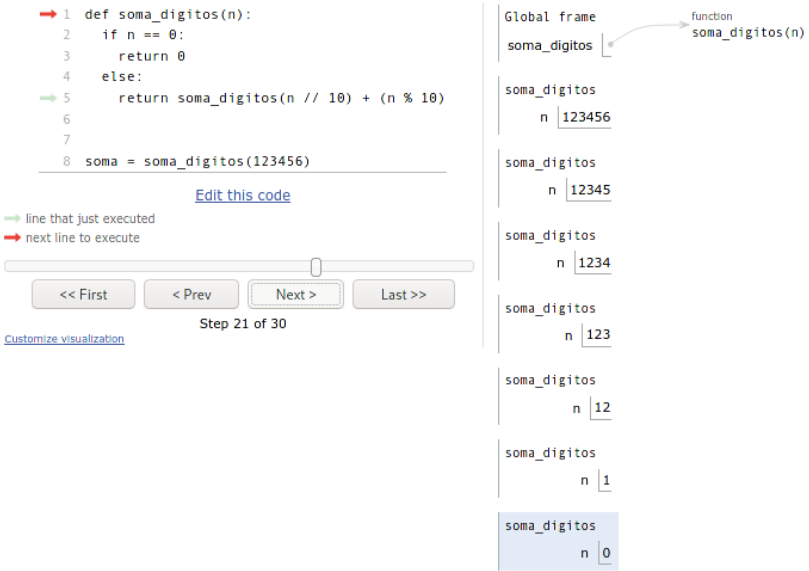
https://pythontutor.com
https://tinyurl.com/4asc5unh



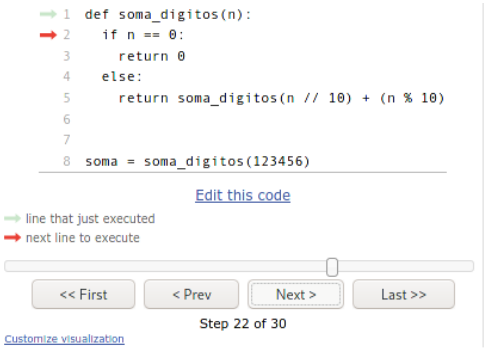
https://pythontutor.com
https://tinyurl.com/4asc5unh

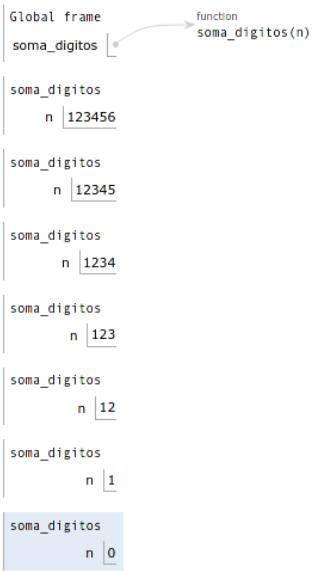


https://pythontutor.com
https://tinyurl.com/4asc5unh

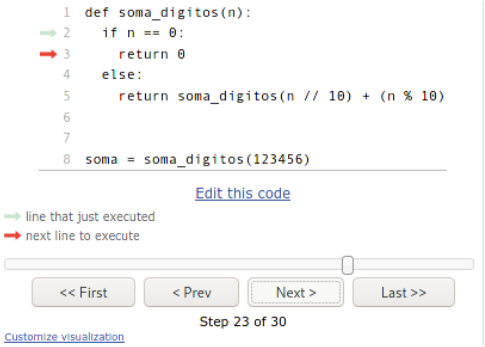


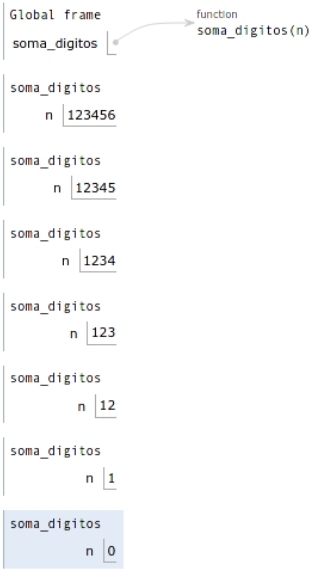
https://pythontutor.com
https://tinyurl.com/4asc5unh





https://pythontutor.com
https://tinyurl.com/4asc5unh





https://pythontutor.com
https://tinyurl.com/4asc5unh

```
1  def soma_digitos(n):
2    if n == 0:
3      return 0
4    else:
5      return soma_digitos(n // 10) + (n % 10)
6
7
8  soma = soma_digitos(123456)
```

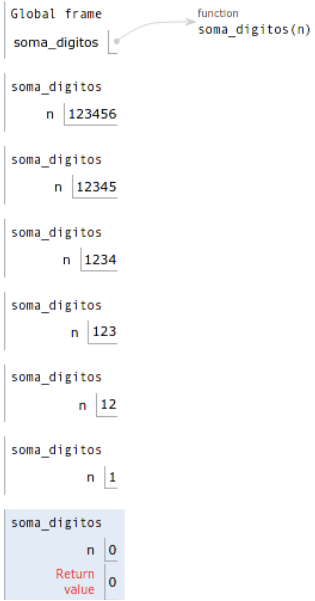
Edit this code

line that just executed
next line to execute

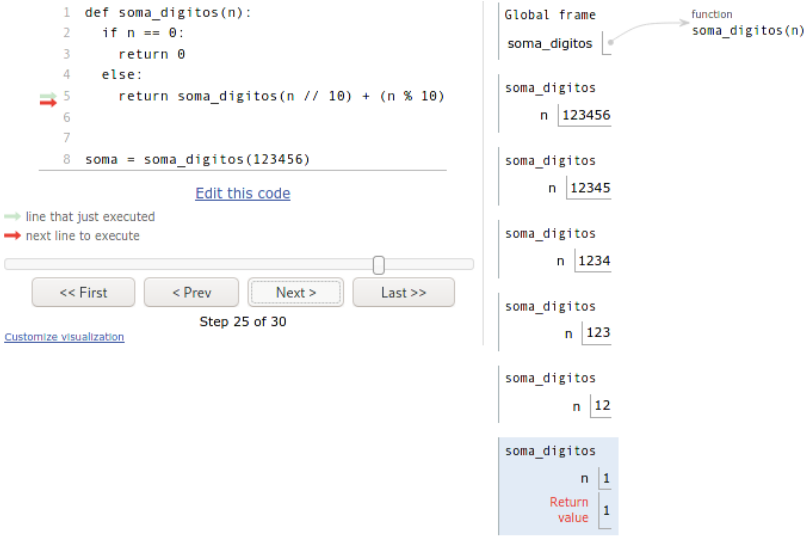
<< First | < Prev | Next > | Last >>

Step 24 of 30

Customize visualization



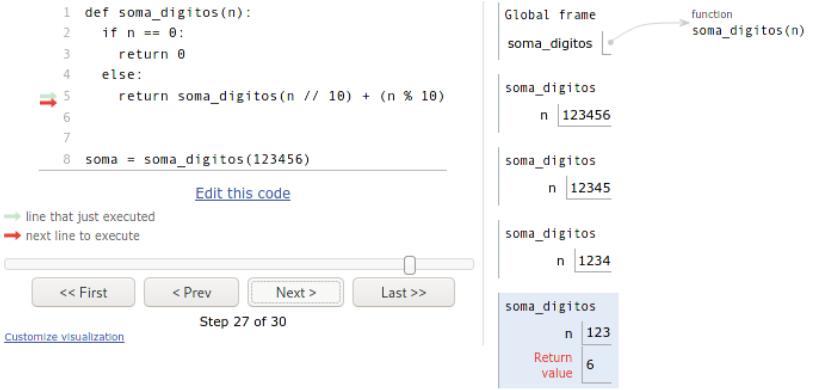
https://pythontutor.com
https://tinyurl.com/4asc5unh



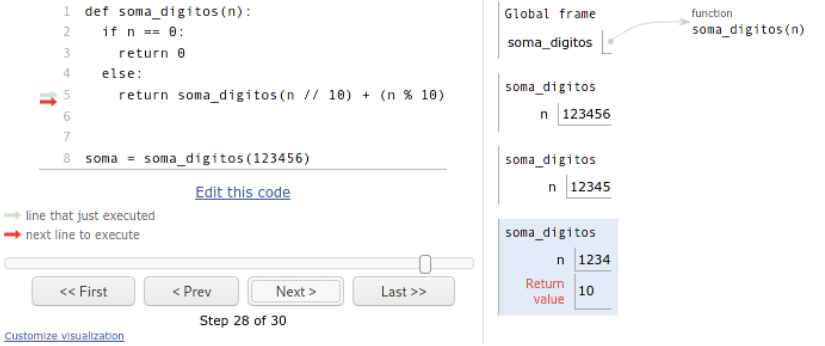
https://pythontutor.com
https://tinyurl.com/4asc5unh

https://pythontutor.com

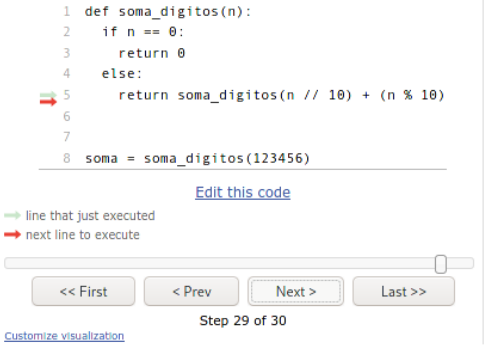
https://tinyurl.com/4asc5unh



https://pythontutor.com
https://tinyurl.com/4asc5unh

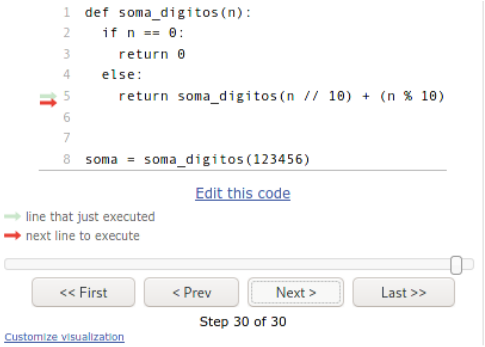


https://pythontutor.com
https://tinyurl.com/4asc5unh

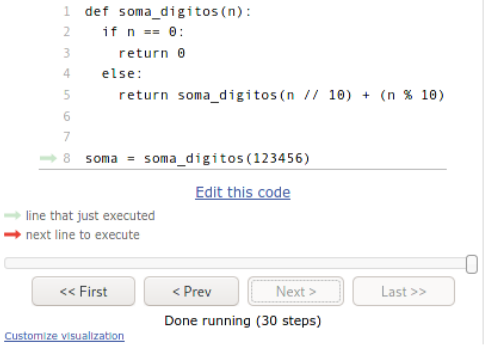


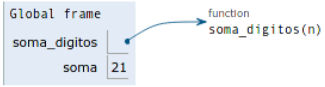
https://pythontutor.com

https://tinyurl.com/4asc5unh



https://pythontutor.com
https://tinyurl.com/4asc5unh





https://pythontutor.com
https://tinyurl.com/4asc5unh

https://ic.unicamp.br/~mc102/aulas/aula12.pdf

- Soma dos dígitos de um número inteiro não negativo.

```
1 def soma_digitos(n):
2   if n < 10:
3     return n
4   else:
5     return soma_digitos(n // 10) + (n % 10)
```

Exemplo 11

https://ic.unicamp.br/~mc102/aulas/aula12.pdf

- O algoritmo de Euclides para o cálculo do Máximo Divisor Comum entre dois números inteiros não negativos (e não ambos iguais a zero) x e y pode ser resumido na seguinte fórmula:

$$mdc(x, y) = \begin{cases} x, & \text{se } y = 0 \\ mdc(y, x \,\%\, y), & \text{se } y > 0 \end{cases}$$

Ideias para provar ....

$$mdc(x, y) = mdc(y, x\%y)$$

Existe D tal que $x = D \times X$ e $y = D \times Y$ onde D é o maior possivel.

$mdc(x, y) = mdc(y, x\%y)$

Existe D tal que $x = D \times X$ e $y = D \times Y$ onde D é o maior possivel.

$x = q \times y + r$ onde $r = x\%y$

$mdc(x, y) = mdc(y, x\%y)$

Existe D tal que $x = D \times X$ e $y = D \times Y$ onde D é o maior possivel.

$x = q \times y + r$ onde $r = x\%y$

$D \times X = q \times D \times Y + r$

$mdc(x, y) = mdc(y, x\%y)$

Existe D tal que $x = D \times X$ e $y = D \times Y$ onde D é o maior possivel.

$x = q \times y + r$ onde $r = x\%y$

$D \times X = q \times D \times Y + r \qquad \Rightarrow \qquad r = D \times (X - q \times Y)$

$mdc(x, y) = mdc(y, x\%y)$

Existe D tal que $x = D \times X$ e $y = D \times Y$ onde D é o maior possivel.

$x = q \times y + r$ onde $r = x\%y$

$D \times X = q \times D \times Y + r \qquad \Rightarrow \qquad r = D \times (X - q \times Y)$

$mdc(y, x\%y) = mdc(D \times Y,\ D \times (X - q \times Y))$

$mdc(x, y) = mdc(y, x\%y)$

Existe D tal que $x = D \times X$ e $y = D \times Y$ onde D é o maior possivel.

$x = q \times y + r$ onde $r = x\%y$

$D \times X = q \times D \times Y + r \qquad \Rightarrow \qquad r = D \times (X - q \times Y)$

$mdc(y, x\%y) = mdc(D \times Y,\ D \times (X - q \times Y))$

Se Y e $(X - q \times Y)$ tem um divisor comum, então

$mdc(x, y) = mdc(y, x\%y)$

Existe D tal que $x = D \times X$ e $y = D \times Y$ onde D é o maior possivel.

$x = q \times y + r$ onde $r = x\%y$

$D \times X = q \times D \times Y + r \qquad \Rightarrow \qquad r = D \times (X - q \times Y)$

$mdc(y, x\%y) = mdc(D \times Y,\ D \times (X - q \times Y))$

Se Y e $(X - q \times Y)$ tem um divisor comum, então

esse divisor tambem divide X e D não é o maximo (Contradição!!)

$mdc(x, y) = mdc(y, x\%y)$

Existe D tal que $x = D \times X$ e $y = D \times Y$ onde D é o maior possivel.

$x = q \times y + r$ onde $r = x\%y$

$D \times X = q \times D \times Y + r \qquad \Rightarrow \qquad r = D \times (X - q \times Y)$

$mdc(y, x\%y) = mdc(D \times Y,\ D \times (X - q \times Y))$

Se Y e $(X - q \times Y)$ tem um divisor comum, então

esse divisor tambem divide X e D não é o maximo (Contradição!!)

Concluimos que Y e $(X - q \times Y)$ não tem um divisor comum, então

$mdc(x, y) = mdc(y, x\%y)$

Existe D tal que $x = D \times X$ e $y = D \times Y$ onde D é o maior possivel.

$x = q \times y + r$ onde $r = x\%y$

$D \times X = q \times D \times Y + r \qquad \Rightarrow \qquad r = D \times (X - q \times Y)$

$mdc(y, x\%y) = mdc(D \times Y,\ D \times (X - q \times Y))$

Se Y e $(X - q \times Y)$ tem um divisor comum, então

esse divisor tambem divide X e D não é o maximo (Contradição!!)

Concluimos que Y e $(X - q \times Y)$ não tem um divisor comum, então

$mdc(y, x\%y) = D$

$mdc(x, y) = mdc(y, x\%y)$

Existe D tal que $x = D \times X$ e $y = D \times Y$ onde D é o maior possivel.

$x = q \times y + r$ onde $r = x\%y$

$D \times X = q \times D \times Y + r \qquad \Rightarrow \qquad r = D \times (X - q \times Y)$

$mdc(y, x\%y) = mdc(D \times Y,\ D \times (X - q \times Y))$

Se Y e $(X - q \times Y)$ tem um divisor comum, então

esse divisor tambem divide X e D não é o maximo (Contradição!!)

Concluimos que Y e $(X - q \times Y)$ não tem um divisor comum, então

$mdc(y, x\%y) = D = mdc(x, y)$

https://ic.unicamp.br/~mc102/aulas/aula12.pdf

```
1  def mdc(x, y):
2    while y > 0:
3      (x, y) = (y, x % y)
4
5    return x
```

https://ic.unicamp.br/~mc102/aulas/aula12.pdf

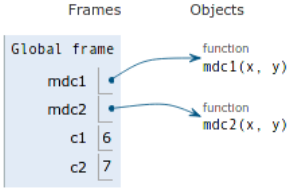
```
1  def mdc(x, y):
2    if y == 0:
3      return x
4    else:
5      return mdc(y, x % y)
```

Python 3.6
(known limitations)

```
1
2  def mdc1(x, y):
3      global c1
4      while y > 0:
5          c1+=1
6          (x, y) = (y, x % y)
7      return x
8
9
10 def mdc2(x, y):
11   global c2
12   c2+=1
13   if y == 0:
14     return x
15   else:
16     return mdc2(y, x % y)
17
18 c1=c2=0
19 print(mdc1(34,1892),mdc2(34,1892),c1,c2)
```

Print output (drag lower right corner to resize)

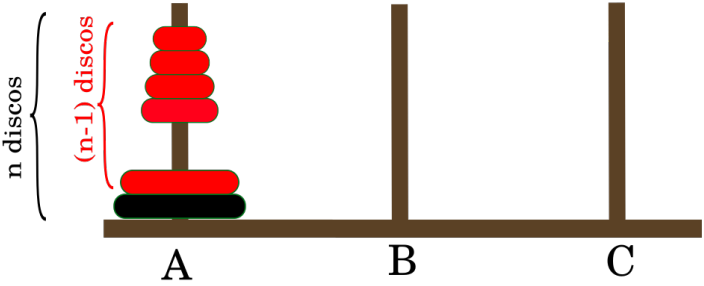
```
2 2 6 7
```



https://pythontutor.com
https://tinyurl.com/3yzz7yk7

Exemplo 12

Torre de Hanói



https://ic.unicamp.br/~mc102/aulas/aula12.pdf

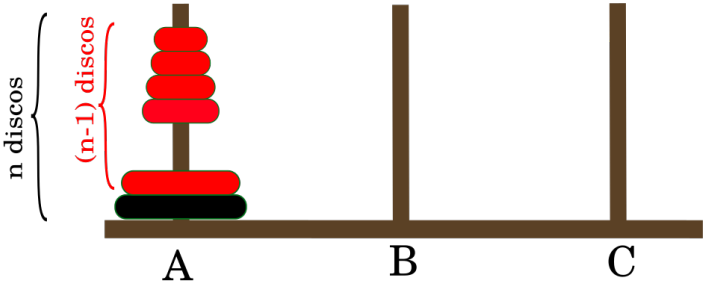
- Considere *n* discos de diâmetros diferentes colocados em um pino A.
- O problema da Torre de Hanói consiste em transferir os *n* discos do pino A (inicial) para o pino C (final), usando um pino B como auxiliar.
- Entretanto, deve-se respeitar algumas regras:
  - Apenas o disco do topo de um pino pode ser movido.
  - Nunca um disco de diâmetro maior pode ficar sobre um disco de diâmetro menor.

https://ic.unicamp.br/~mc102/aulas/aula12.pdf

- O problema foi descrito pela primeira vez no ocidente em 1883 pelo matemático francês Édouard Lucas, baseado numa lenda hindu, onde Brahma havia ordenado que os monges do templo de Kashi Vishwanath movessem uma pilha de 64 discos de ouro, segundo as regras previamentes descritas.
- Quando todos os discos tivessem sido movidos, o mundo acabaria.

Torre de Hanói: "Problema de tamanho **n**"

Mover **n** discos desde A até B usando C

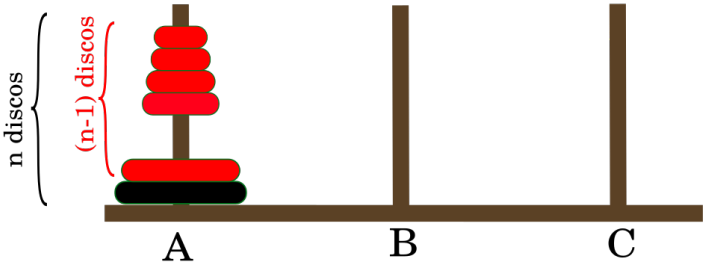


"Problema de tamanho **n**" = P(n, A, B, C)

Torre de Hanói:

Mover **n** discos desde A até B usando C

Mover **n-1** discos desde A até B usando C

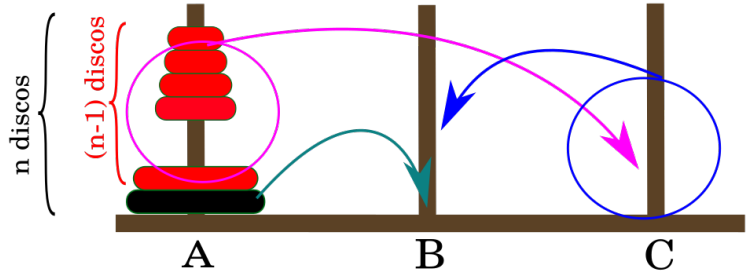


"Problema de tamanho **n**" = P(n , A, B, C)

"Problema de tamanho **n-1**" = P(n-1, A, B, C)

"Problema de tamanho **n**" = P(n, A, B, C)

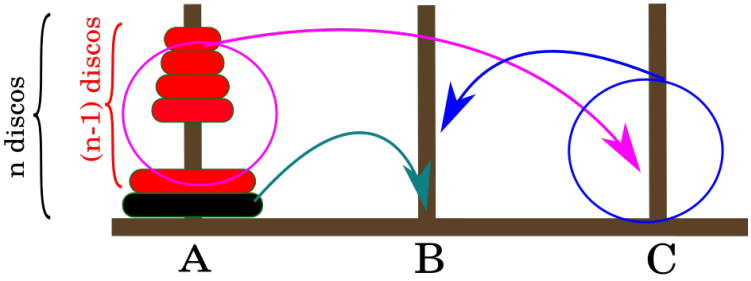
Mover **n** discos desde A até B usando C



$$P(n, A, B, C) = P(n-1,A,C,B) + P(1,A,B,C) + P(n-1,C,B,A)$$

$$T(n) = \quad T(n-1) \quad + 1 + \quad T(n-1)$$

$$P(n, A, B, C) = P(n-1,A,C,B) + P(1,A,B,C) + P(n-1,C,B,A)$$



$$T(n) = 2\,T(n-1) + 1 \quad \text{e} \quad T(1) = 1$$

https://ic.unicamp.br/~mc102/aulas/aula12.pdf

- Vamos usar indução para obter um algoritmo para este problema.

Teorema

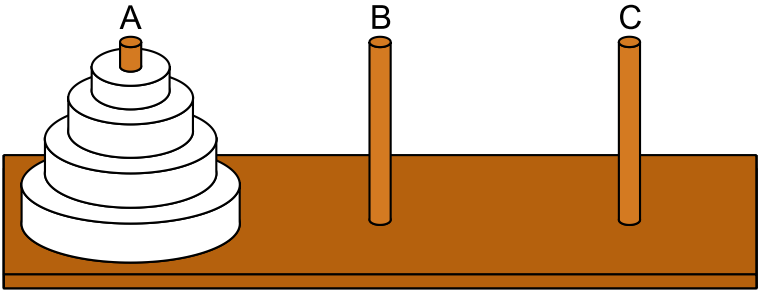
É possível resolver o problema da Torre de Hanói com n discos.

https://ic.unicamp.br/~mc102/aulas/aula12.pdf

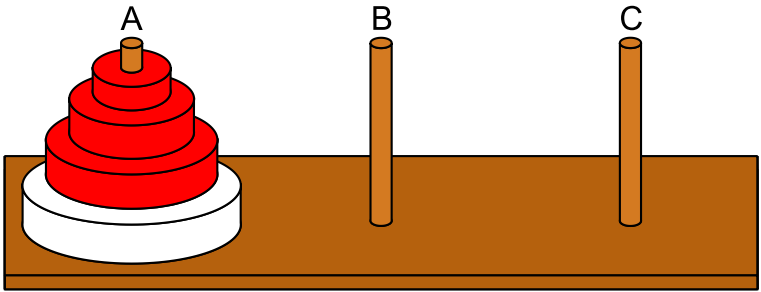
Prova

- Base da Indução: $n = 1$. Neste caso, temos apenas um disco. Basta mover este disco do pino A para o pino C.
- Hipótese de Indução: Sabemos como resolver o problema quando há $n - 1$ discos.
- Passo de Indução: Devemos resolver o problema para n discos assumindo que sabemos resolver o problema com $n - 1$ discos.
- Por hipótese de indução, sabemos mover os $n - 1$ primeiros discos do pino A para o pino B usando o pino C como auxiliar.
- Depois de movermos estes $n - 1$ discos, movemos o maior disco (que continua no pino A) para o pino C.
- Novamente, pela hipótese de indução, sabemos mover os $n - 1$ discos do pino B para o pino C usando o pino A como auxiliar.
- Com isso, temos uma solução para o caso em que há n discos.

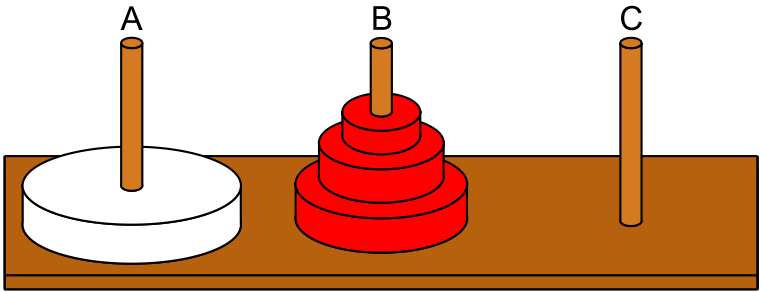
https://ic.unicamp.br/~mc102/aulas/aula12.pdf



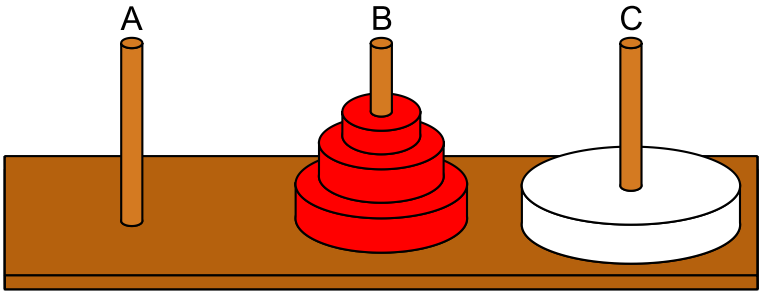
https://ic.unicamp.br/~mc102/aulas/aula12.pdf



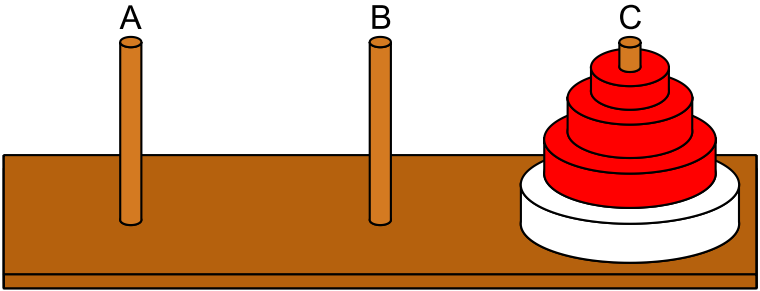
https://ic.unicamp.br/~mc102/aulas/aula12.pdf

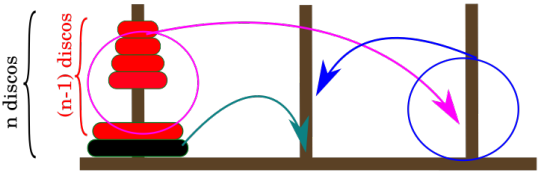


https://ic.unicamp.br/~mc102/aulas/aula12.pdf



https://ic.unicamp.br/~mc102/aulas/aula12.pdf





$$T(n) = 2\,T(n-1) + 1 \quad \text{e} \quad T(1) = 1$$

$$\begin{aligned}
T(2) &= 2\,T(1) + 1 = 3 \\
T(3) &= 2\,T(2) + 1 = 7 \\
T(4) &= 2\,T(3) + 1 = 15 \\
T(5) &= 2\,T(4) + 1 = 31 \\
T(6) &= 2\,T(5) + 1 = 63
\end{aligned}$$

https://ic.unicamp.br/~mc102/aulas/aula12.pdf

- Como solucionar o problema de forma recursiva:
  - Se $n = 1$ então mova o único disco de A para C.
  - Caso contrário ($n > 1$), desloque de forma recursiva os $n - 1$ primeiros discos de A para B, usando C como auxiliar.
  - Mova o último disco de A para C.
  - Mova, de forma recursiva, os $n - 1$ discos de B para C, usando A como auxiliar.

https://ic.unicamp.br/~mc102/aulas/aula12.pdf

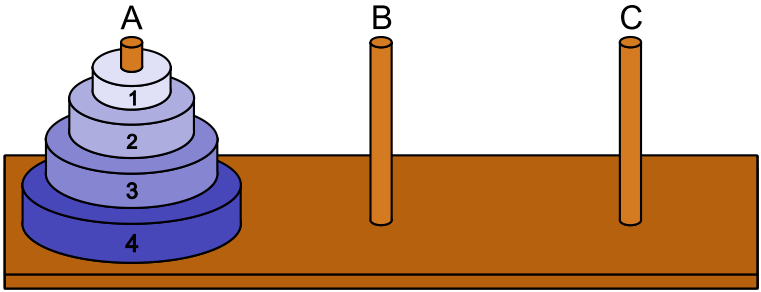
```
1 def hanoi(n, inicial, final, auxiliar):
2   s = "Mova o disco {} do pino {} para o pino {}"
3   if n == 1:
4     print(s.format(n, inicial, final))
5   else:
6     hanoi(n - 1, inicial, auxiliar, final)
7     print(s.format(n, inicial, final))
8     hanoi(n - 1, auxiliar, final, inicial)
```

https://ic.unicamp.br/~mc102/aulas/aula12.pdf

- Solução para `hanoi(4, "A", "C", "B")`:

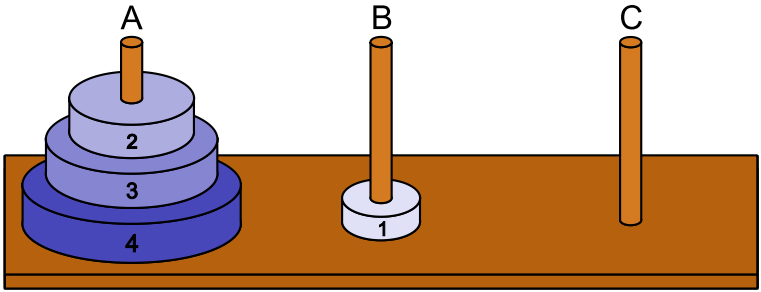
```
Mova o disco 1 do pino A para o pino B
Mova o disco 2 do pino A para o pino C
Mova o disco 1 do pino B para o pino C
Mova o disco 3 do pino A para o pino B
Mova o disco 1 do pino C para o pino A
Mova o disco 2 do pino C para o pino B
Mova o disco 1 do pino A para o pino B
Mova o disco 4 do pino A para o pino C
Mova o disco 1 do pino B para o pino C
Mova o disco 2 do pino B para o pino A
Mova o disco 1 do pino C para o pino A
Mova o disco 3 do pino B para o pino C
Mova o disco 1 do pino A para o pino B
Mova o disco 2 do pino A para o pino C
Mova o disco 1 do pino B para o pino C
```

https://ic.unicamp.br/~mc102/aulas/aula12.pdf



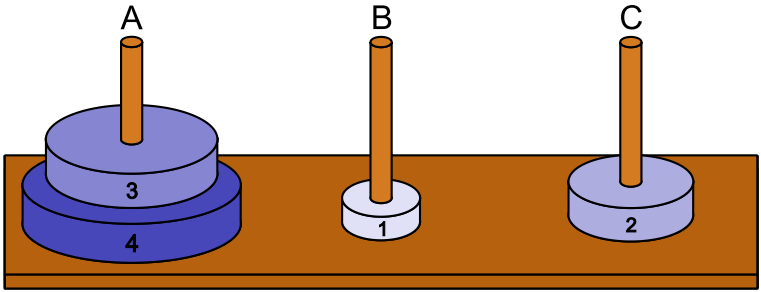
Configuração Inicial

https://ic.unicamp.br/~mc102/aulas/aula12.pdf



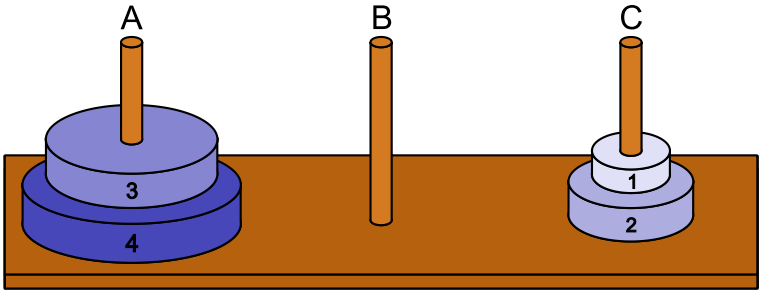

```
Mova o disco 1 do pino A para o pino B
```

https://ic.unicamp.br/~mc102/aulas/aula12.pdf



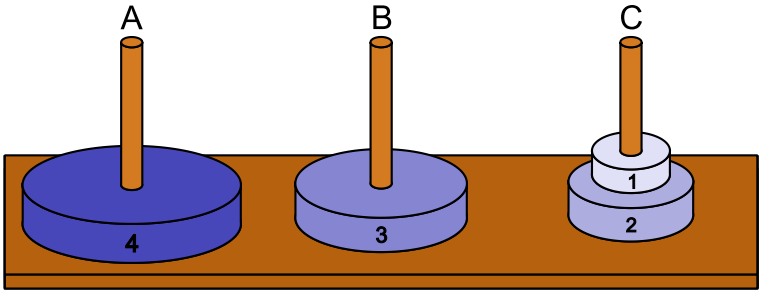

```
Mova o disco 2 do pino A para o pino C
```

https://ic.unicamp.br/~mc102/aulas/aula12.pdf



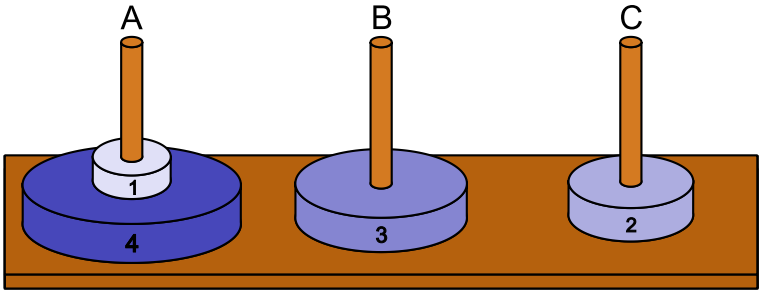

```
Mova o disco 1 do pino B para o pino C
```

https://ic.unicamp.br/~mc102/aulas/aula12.pdf



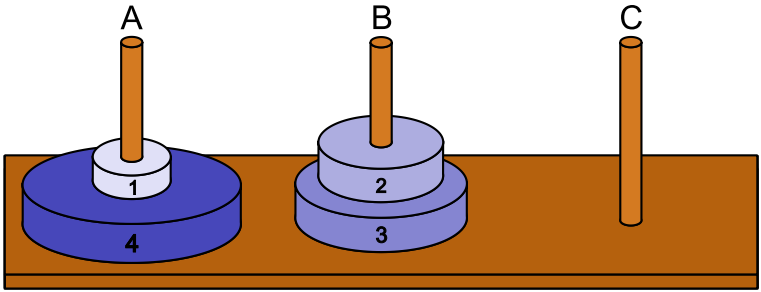

```
Mova o disco 3 do pino A para o pino B
```

https://ic.unicamp.br/~mc102/aulas/aula12.pdf



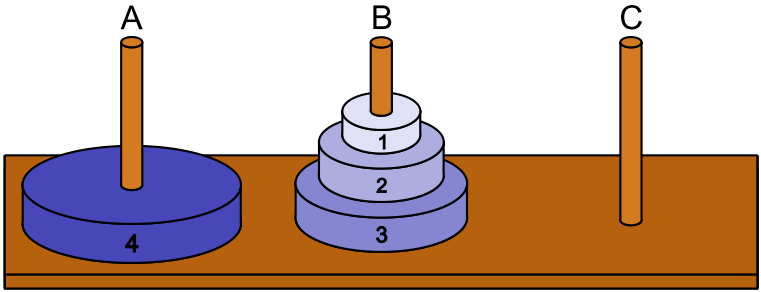

```
Mova o disco 1 do pino C para o pino A
```

https://ic.unicamp.br/~mc102/aulas/aula12.pdf



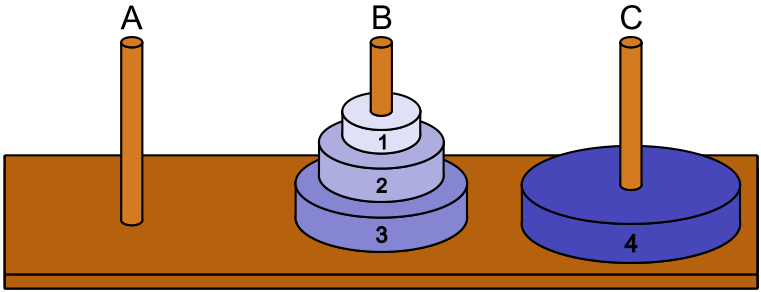

```
Mova o disco 2 do pino C para o pino B
```

https://ic.unicamp.br/~mc102/aulas/aula12.pdf



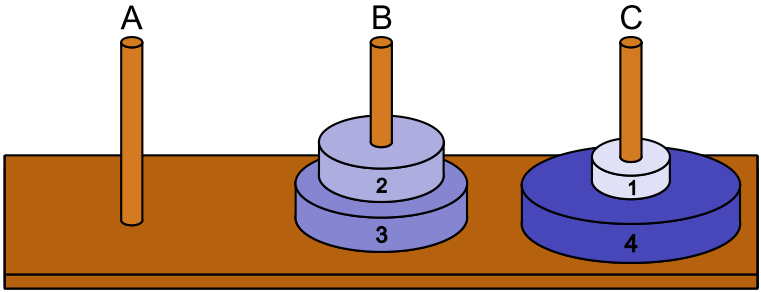

```
Mova o disco 1 do pino A para o pino B
```

https://ic.unicamp.br/~mc102/aulas/aula12.pdf



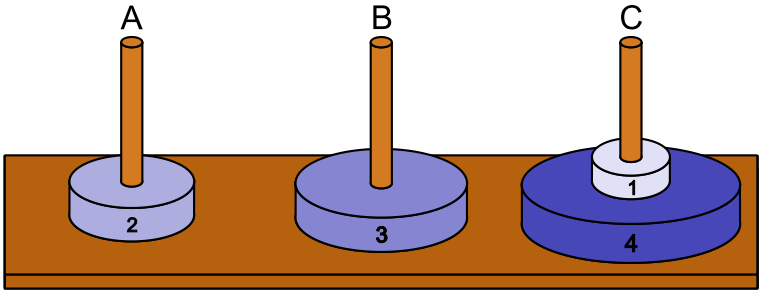

```
Mova o disco 4 do pino A para o pino C
```

https://ic.unicamp.br/~mc102/aulas/aula12.pdf



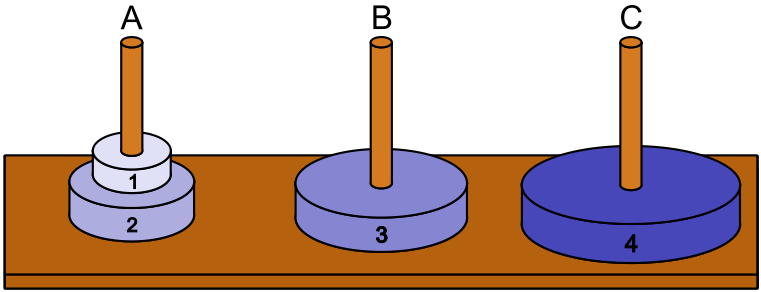

```
Mova o disco 1 do pino B para o pino C
```

https://ic.unicamp.br/~mc102/aulas/aula12.pdf



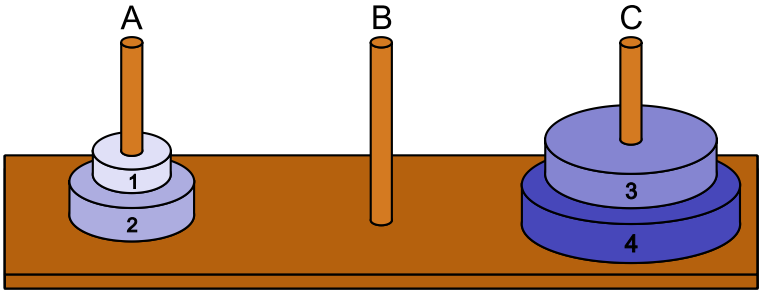

```
Mova o disco 2 do pino B para o pino A
```

https://ic.unicamp.br/~mc102/aulas/aula12.pdf



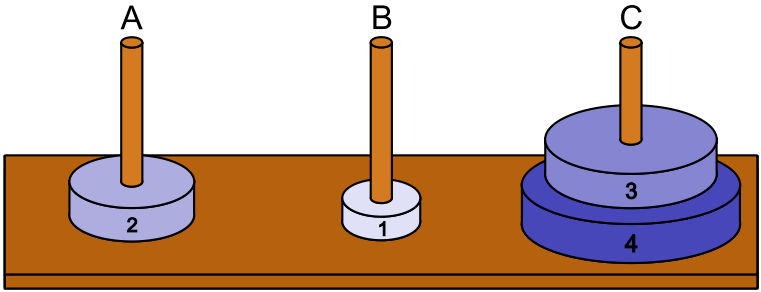

```
Mova o disco 1 do pino C para o pino A
```

https://ic.unicamp.br/~mc102/aulas/aula12.pdf



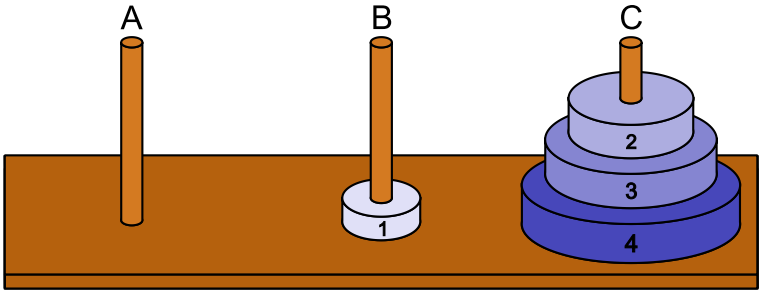

```
Mova o disco 3 do pino B para o pino C
```

https://ic.unicamp.br/~mc102/aulas/aula12.pdf



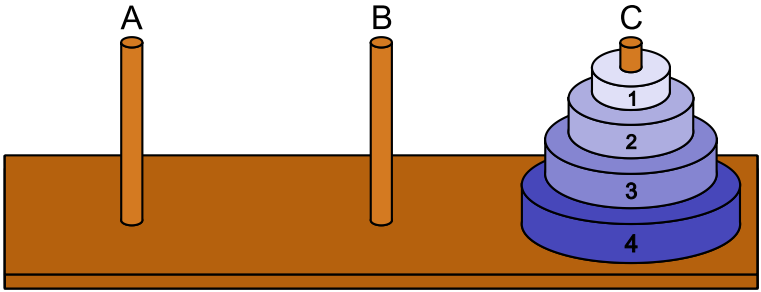

```
Mova o disco 1 do pino A para o pino B
```

https://ic.unicamp.br/~mc102/aulas/aula12.pdf




```
Mova o disco 2 do pino A para o pino C
```

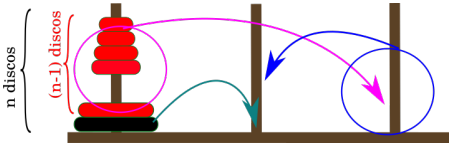
https://ic.unicamp.br/~mc102/aulas/aula12.pdf




```
Mova o disco 1 do pino B para o pino C
```

https://ic.unicamp.br/~mc102/aulas/aula12.pdf

- Seja $T(n)$ o número de movimentos necessários para mover uma pilha de n discos.
- Claramente temos que:
  - $T(1) = 1$
  - $T(n) = 2T(n-1) + 1$
- O que nos permite deduzir que:
  - $T(2) = 2T(1) + 1 = 3$
  - $T(3) = 2T(2) + 1 = 7$
  - $T(4) = 2T(3) + 1 = 15$
  - $T(5) = 2T(4) + 1 = 31$
  - $\ldots$
  - $T(n) = 2^n - 1$
- No caso de 64 discos são necessários 18.446.744.073.709.551.615 movimentos ou, aproximadamente, 585 bilhões de anos, se cada movimento puder ser feito em um segundo.



$$T(n) = 2\ T(n-1) + 1$$

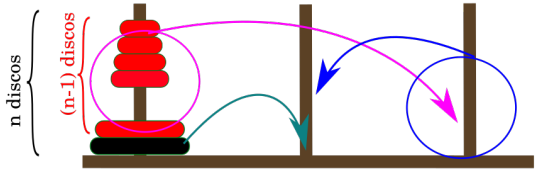
Podemos assumir que $T(n) = A\ B^n + C$, assim:

$$A\ B^n + C = 2(A\ B^{n-1} + C) + 1$$

Temos que: $C = 2C + 1$ então $C = -1$
e que: $A\ B^n = 2\ A\ B^{n-1}$ então $B = 2$

Como: $T(1) = 1$ aplicando em $T(n) = A\ B^n + C$
temos que: $T(1) = A\ 2^1 - 1 = 1$ então $A = 1$

Finalmente: $T(n) = 2^n - 1$



$$T(n) = 2\,T(n-1) + 1$$
$$T(1) = 1$$

$$T(2) = 2\,T(1) + 1 = 3$$
$$T(3) = 2\,T(2) + 1 = 7$$
$$T(4) = 2\,T(3) + 1 = 15$$
$$T(5) = 2\,T(4) + 1 = 31$$
$$T(6) = 2\,T(5) + 1 = 63$$

$$T(n) = 2^n - 1$$

$$T(100) = 2^{100} - 1 = 1.2676 \times 10^{30}$$

40 196 936 841 331 475 186 983, 23 anos, 1 pç x seg

40 196 936 841 331 475, 18 anos, 1 milhão pçs x seg

2 679 795, 78 vezes a idade do universo(15bi), 1 milhão pçs x seg

Perguntas ....

Referências

- Zanoni Dias, MC102, Algoritmos e Programação de Computadores, IC/UNICAMP, 2021. https://ic.unicamp.br/~mc102/

  Aula Introdutória [slides] [vídeo]
  Primeira Aula de Laboratório [slides] [vídeo]
  Python Básico: Tipos, Variáveis, Operadores, Entrada e Saída [slides] [vídeo]
  Comandos Condicionais [slides] [vídeo]
  Comandos de Repetição [slides] [vídeo]
  Listas e Tuplas [slides] [vídeo]
  Strings [slides] [vídeo]
  Dicionários [slides] [vídeo]
  Funções [slides] [vídeo]
  Objetos Multidimensionais [slides] [vídeo]
  Algoritmos de Ordenação [slides] [vídeo]
  Algoritmos de Busca [slides] [vídeo]
  Recursão [slides] [vídeo]
  Algoritmos de Ordenação Recursivos [slides] [vídeo]
  Arquivos [slides] [vídeo]
  Expressões Regulares [slides] [vídeo]
  Execução de Testes no Google Cloud Shell [slides] [vídeo]
  Numpy [slides] [vídeo]
  Pandas [slides] [vídeo]

- Panda - Cursos de Computação em Python (IME -USP) https://panda.ime.usp.br/

  Como Pensar Como um Cientista da Computação https://panda.ime.usp.br/pensepy/static/pensepy/
  Aulas de Introdução à Computação em Python https://panda.ime.usp.br/aulasPython/static/aulasPython/

- Fabio Kon, Introdução à Ciência da Computação com Python http://bit.ly/FabioKon/
- Socratica, Python Programming Tutorials http://bit.ly/SocraticaPython/
- Google - online editor for cloud-native applications (Python programming) https://shell.cloud.google.com/
- w3schools - Python Tutorial https://www.w3schools.com/python/
- Outros, citados nos Slides.