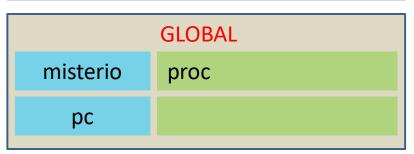
Paso 1: Estado inicial.

```
def misterio(n):
       if n == 0:
3
            yield [1]
       else:
5
            for x in misterio(n-1):
6
                  r = []
                  for y in suspenso(0, x):
8
                       r = [*r, y]
9
                  yield r
10 for x in misterio(5):
11
        print(x)
```

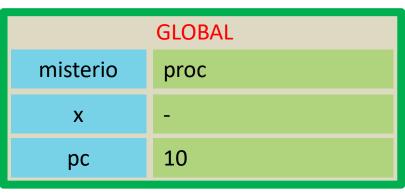


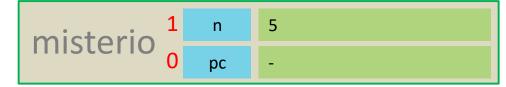
IMPRIME:

Nota: En cada llamada recursiva nos referiremos a los procedimientos recursivos con ' para ayudar a diferenciar

Paso 2: Línea 10 inicia el ciclo for, ejecuta *misterio(5)* se crea el marco de pila de misterio.

```
def misterio(n):
2
       if n == 0:
3
            yield [1]
4
       else:
5
            for x in misterio(n-1):
6
                 r = []
7
                 for y in suspenso(0, x):
8
                       r = [*r, y]
9
                 yield r
10 for x in misterio(5):
        print(x)
11
```

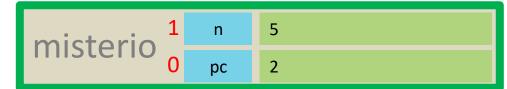




Paso 3: Se evalúa la condición de la línea 2 *if* n == 0

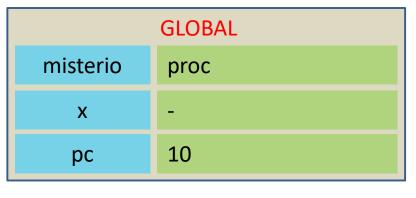
```
def misterio(n):
       if n == 0:
3
            yield [1]
       else:
5
            for x in misterio(n-1):
6
                  r = []
                  for y in suspenso(0, x):
8
                       r = [*r, y]
9
                  yield r
10 for x in misterio(5):
        print(x)
11
```

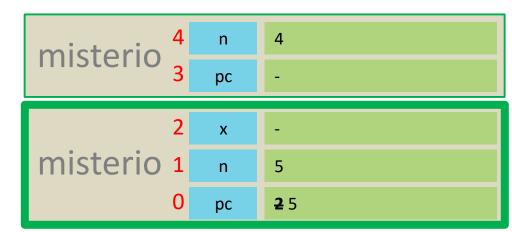
GLOBAL	
misterio	proc
Х	-
рс	10



Paso 4: Como n != 0 **No se cumple** la condición y pasa a la línea 5, inicia el ciclo for, ejecuta misterio(n-1), es decir, misterio(4), se crea el marco de pila de misterio'

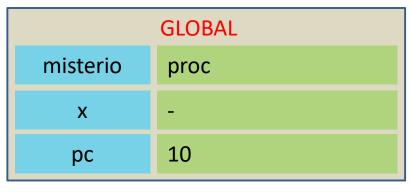
```
def misterio(n):
2
       if n == 0:
3
            yield [1]
       else:
5
            for x in misterio(n-1):
6
                  r = []
                  for y in suspenso(0, x):
8
                       r = [*r, y]
9
                  yield r
10 for x in misterio(5):
11
        print(x)
```

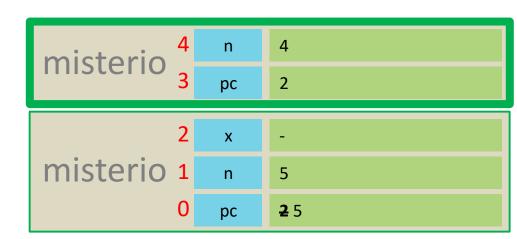




Paso 5: Se evalúa la condición de la línea 2 *if* n == 0

```
def misterio(n):
       if n == 0:
3
            yield [1]
4
       else:
5
            for x in misterio(n-1):
6
                  r = []
7
                  for y in suspenso(0, x):
8
                       r = [*r, y]
9
                  yield r
10 for x in misterio(5):
        print(x)
11
```

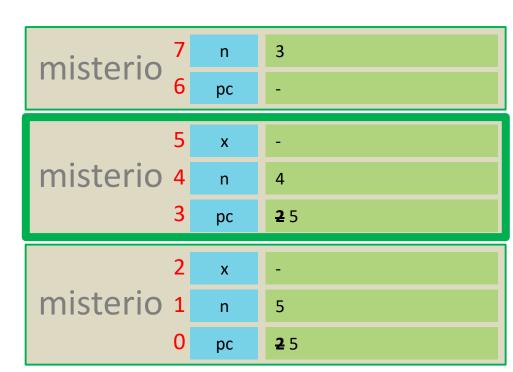




Paso 6: Como n != 0 **No se cumple** la condición y pasa a la línea 5, inicia el ciclo for, ejecuta misterio(n-1), es decir, misterio(3), se crea el marco de pila de misterio''

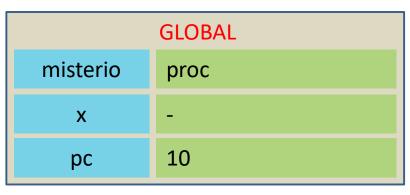
```
def misterio(n):
2
       if n == 0:
3
            yield [1]
       else:
5
            for x in misterio(n-1):
6
                  r = []
7
                  for y in suspenso(0, x):
8
                       r = [*r, y]
9
                  yield r
10 for x in misterio(5):
11
        print(x)
```

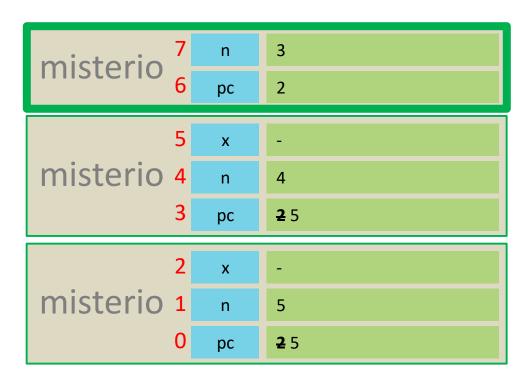




Paso 7: Se evalúa la condición de la línea 2 *if* n == 0

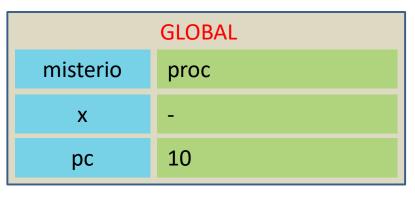
```
def misterio(n):
       if n == 0:
3
            yield [1]
4
       else:
5
            for x in misterio(n-1):
6
                  r = []
7
                  for y in suspenso(0, x):
8
                       r = [*r, y]
9
                  yield r
10 for x in misterio(5):
        print(x)
11
```

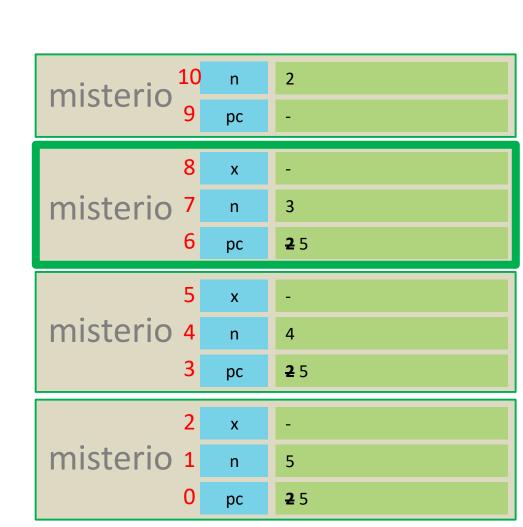




Paso 8: Como n != 0 **No se cumple** la condición y pasa a la línea 5, inicia el ciclo for, ejecuta misterio(n-1), es decir, misterio(2), se crea el marco de pila de misterio'''

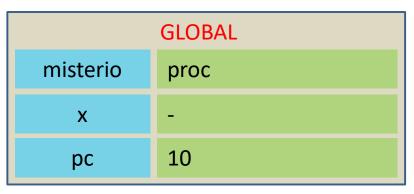
```
def misterio(n):
2
       if n == 0:
3
             yield [1]
       else:
5
             for x in misterio(n-1):
6
                  r = []
                  for y in suspenso(0, x):
8
                       r = [*r, y]
9
                  yield r
10 for x in misterio(5):
11
        print(x)
```

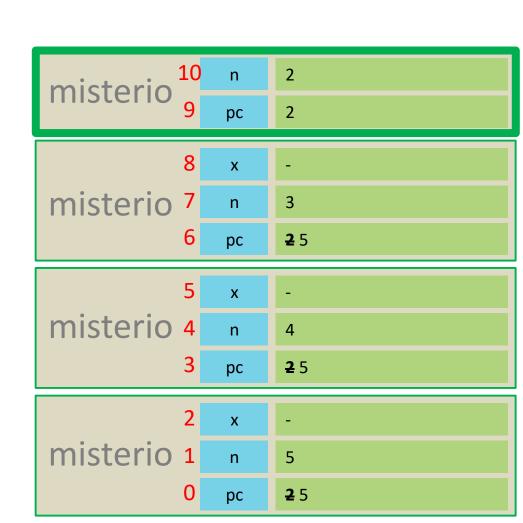




Paso 9: Se evalúa la condición de la línea 2 *if* n == 0

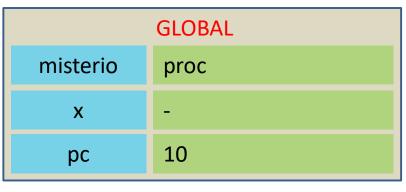
```
def misterio(n):
       if n == 0:
3
            yield [1]
4
       else:
5
            for x in misterio(n-1):
6
                  r = []
7
                  for y in suspenso(0, x):
8
                       r = [*r, y]
9
                  yield r
10 for x in misterio(5):
        print(x)
11
```

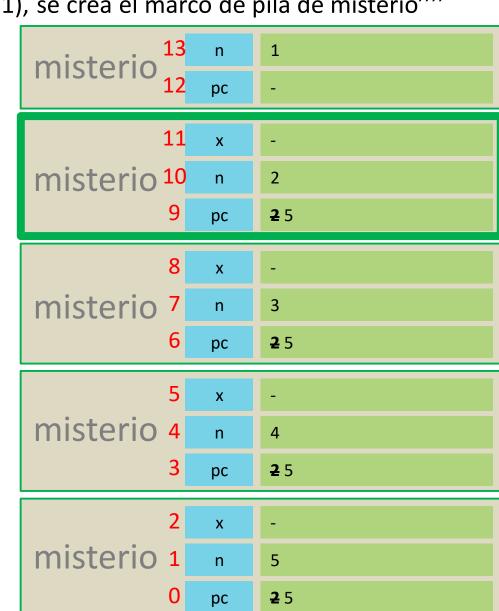




Paso 10: Como n != 0 **No se cumple** la condición y pasa a la línea 5, inicia el ciclo for, ejecuta misterio(n-1), es decir, misterio(1), se crea el marco de pila de misterio''''

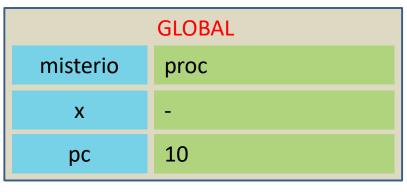
```
def misterio(n):
       if n == 0:
3
             yield [1]
       else:
5
             for x in misterio(n-1):
6
                  r = []
                  for y in suspenso(0, x):
8
                       r = [*r, y]
9
                  yield r
10 for x in misterio(5):
11
        print(x)
```

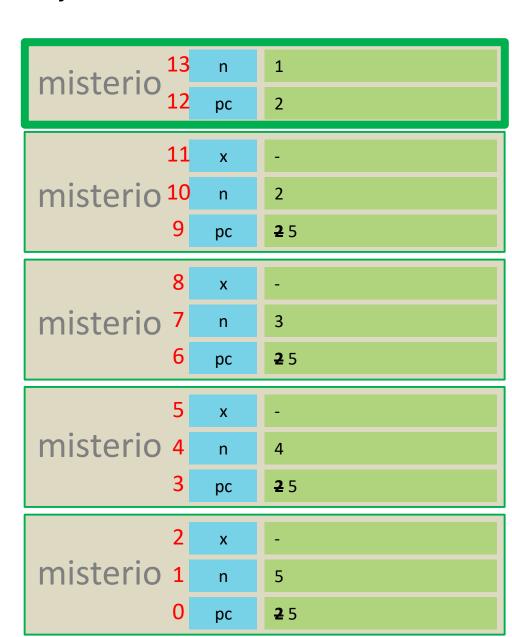




Paso 11: Se evalúa la condición de la línea 2 *if* n == 0

```
def misterio(n):
       if n == 0:
3
            yield [1]
       else:
5
            for x in misterio(n-1):
6
                  r = []
                  for y in suspenso(0, x):
8
                       r = [*r, y]
9
                  yield r
10 for x in misterio(5):
        print(x)
11
```

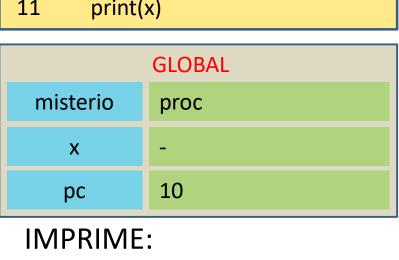




Examen 2 Problema 3 Misterio 16 0 n misterio Paso 12: Como n != 0 No se cumple la condición y рс pasa a la línea 5, inicia el ciclo for, ejecuta misterio(n-1), es decir, misterio(0), se crea el 14 Х marco de pila de misterio"" misterio 13 1 n def misterio(n): 12 **2** 5 рс if n == 0: 2 3 yield [1] 11 Х 4 else: misterio 10 2 n 5 for x in misterio(n-1): 6 **2** 5 рс r = []for y in suspenso(0, x): 8 Х 8 r = [*r, y]misterio 7 9 yield r n 3 10 for x in misterio(5): 6 **2** 5 рс print(x) 11 Х **GLOBAL** misterio 4 4 n misterio proc рс **2** 5 Χ Х 10 рс misterio 1 n 5 **IMPRIME: 2** 5 рс

Examen 2 Problema 3 Misterio 16 0 n misterio Paso 13: Se evalúa la condición de la línea 2 рс 2 if n == 014 Х misterio 13 1 n def misterio(n): 12 **2**5 рс if n == 0: 3 yield [1] 11 Х 4 else: misterio 10 2 n 5 for x in misterio(n-1): 6 **2** 5 рс r = [] 7 for y in suspenso(0, x): 8 8 Χ r = [*r, y]9 misterio ⁷ 3 yield r n 10 for x in misterio(5): **2** 5 рс print(x) 11 Х **GLOBAL** misterio 4 4 n misterio proc **2** 5 рс Χ Χ 10 рс misterio 1 n 5 **IMPRIME: 2** 5 рс

Examen 2 Problema 3 Misterio 16 n misterio Paso 14: Como n == 0 Sí se cumple la condición y рс pasa a la línea 3, ejecuta yield [1] 14 Х misterio 13 n def misterio(n): 12 рс if n == 0: 2 3 yield [1] 11 Х 4 else: misterio 10 n 5 for x in misterio(n-1): 6 рс r = [] 7 for y in suspenso(0, x): 8 8 Χ r = [*r, y]9 misterio ⁷ yield r n 10 for x in misterio(5): рс print(x) 11 Х **GLOBAL** misterio 4 n misterio proc рс



3 **2** 5 4 **2** 5 Х misterio 1 n 5

2 5

рс

0

2 3

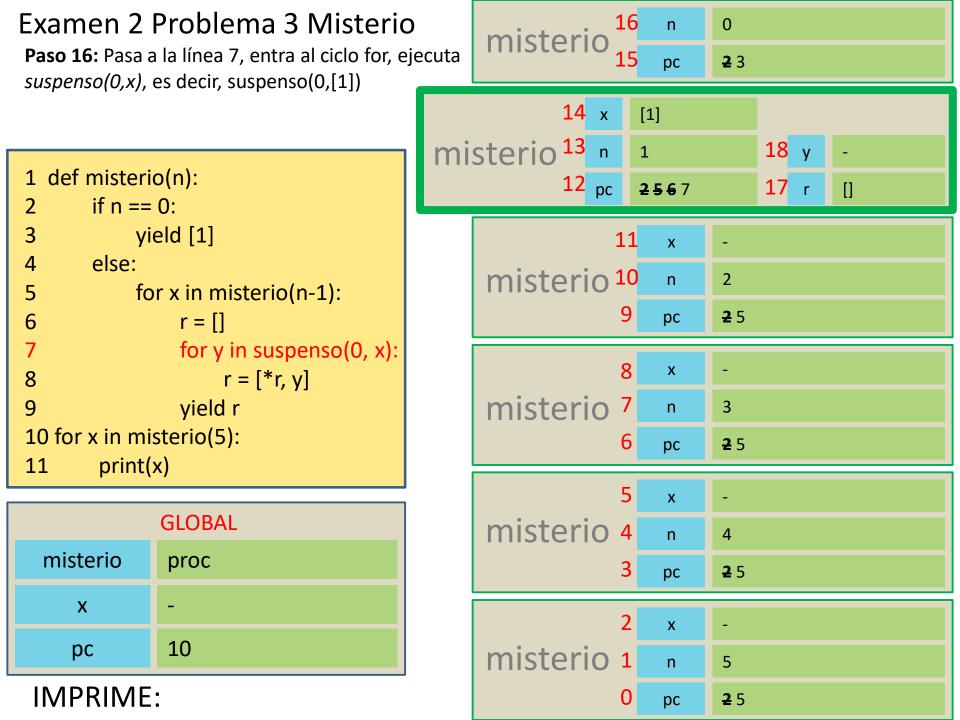
1

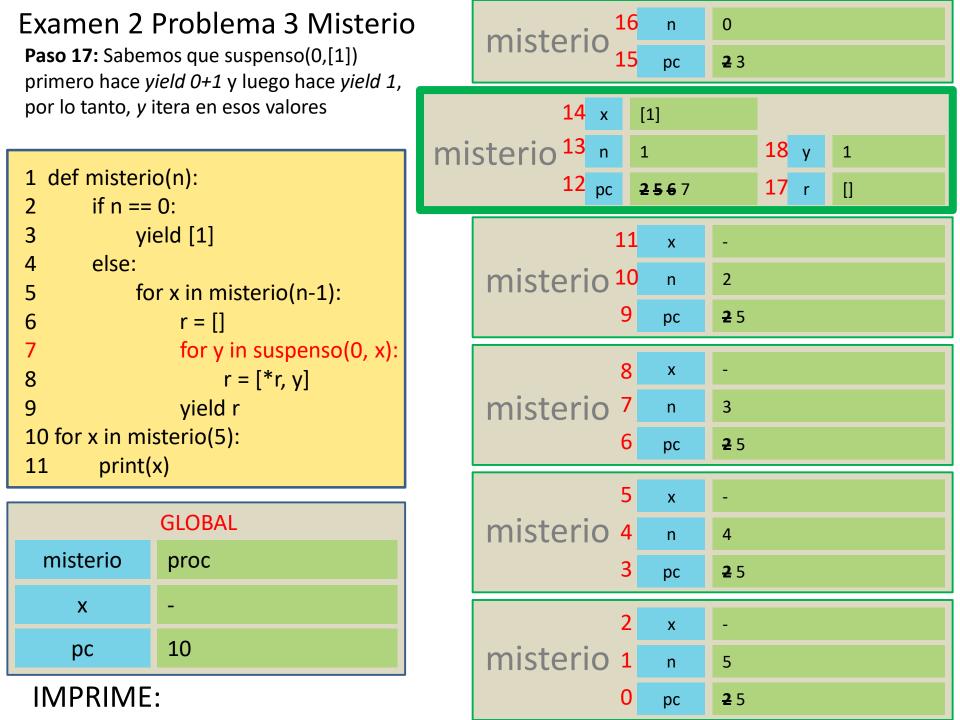
25

2

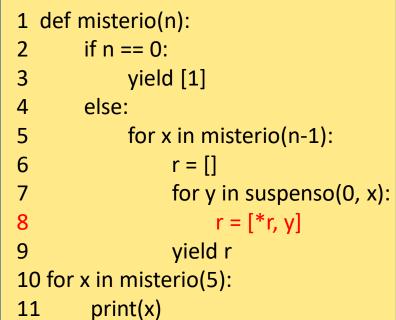
2 5

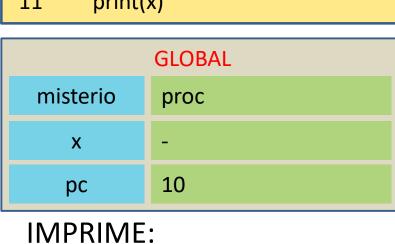
Examen 2 Problema 3 Misterio 0 n misterio Paso 15: Vuelve al marco de misterio" y pasa a la **2** 3 рс línea 6, asigna a r 14 x [1] misterio 13 n 1 def misterio(n): 12 pc **17** r **25**6 [] if n == 0: 2 3 yield [1] 11 Χ 4 else: misterio 10 n 5 for x in misterio(n-1): 6 **2** 5 рс r = []7 for y in suspenso(0, x): 8 8 Χ r = [*r, y]9 misterio ⁷ 3 yield r n 10 for x in misterio(5): **2** 5 рс print(x) 11 Х **GLOBAL** misterio 4 4 n misterio proc **2** 5 рс Χ Χ 10 рс misterio 1 n 5 **IMPRIME: 2** 5 рс

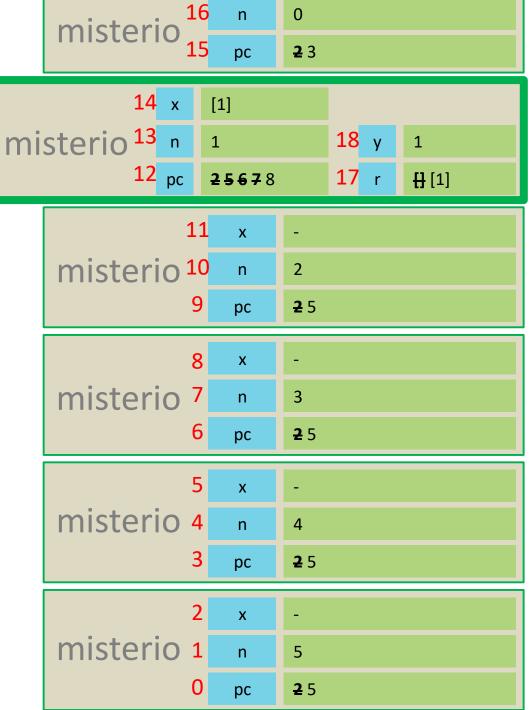




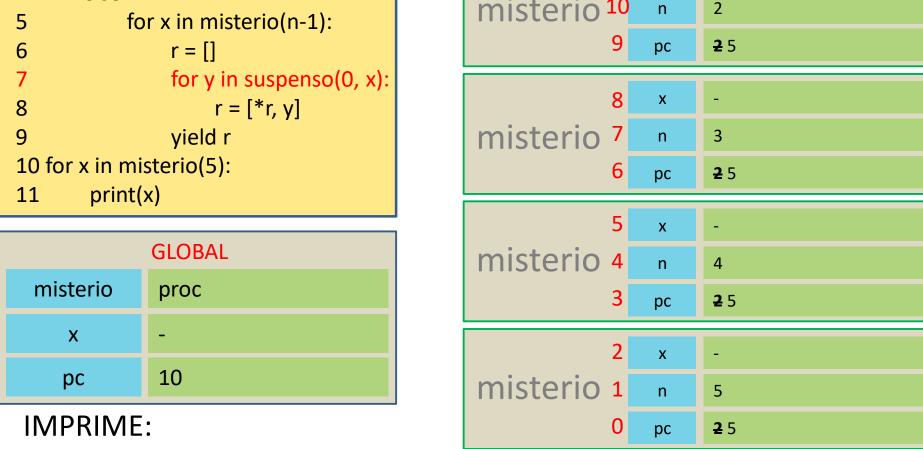
Examen 2 Problema 3 Misterio Paso 18: Pasa a la línea 8, asigna a r 1 def misterio(n): 2 if n == 0: 3 yield [1] 4 else:







Examen 2 Problema 3 Misterio n misterio Paso 19: Vuelve a la línea 7, en el ciclo for рс 14 x [1] misterio 13 n 1 def misterio(n): 12 pc **25678**7 if n == 0: 2 3 yield [1] 11 Χ 4 else: misterio 10 n 5 for x in misterio(n-1): 6 рс r = [] **7** 8 for y in suspenso(0, x): 8 Χ r = [*r, y]9 misterio ⁷ yield r n 10 for x in misterio(5): рс print(x) 11 Х **GLOBAL** misterio 4 n misterio proc рс



0

2 3

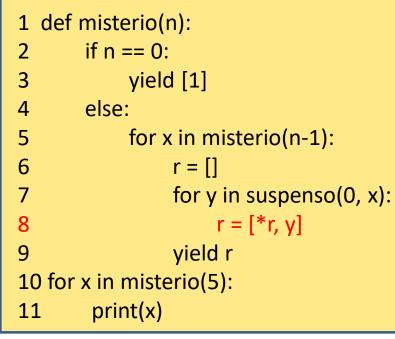
18 y

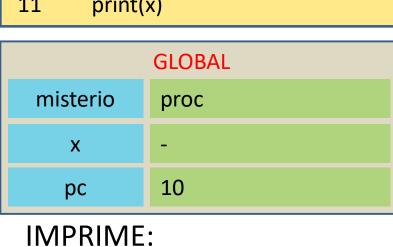
17 r

1 1

[] [1]

Examen 2 Problema 3 Misterio Paso 20: Pasa a la línea 8, asigna a r 1 def misterio(n): 2 if n == 0: 3 yield [1]







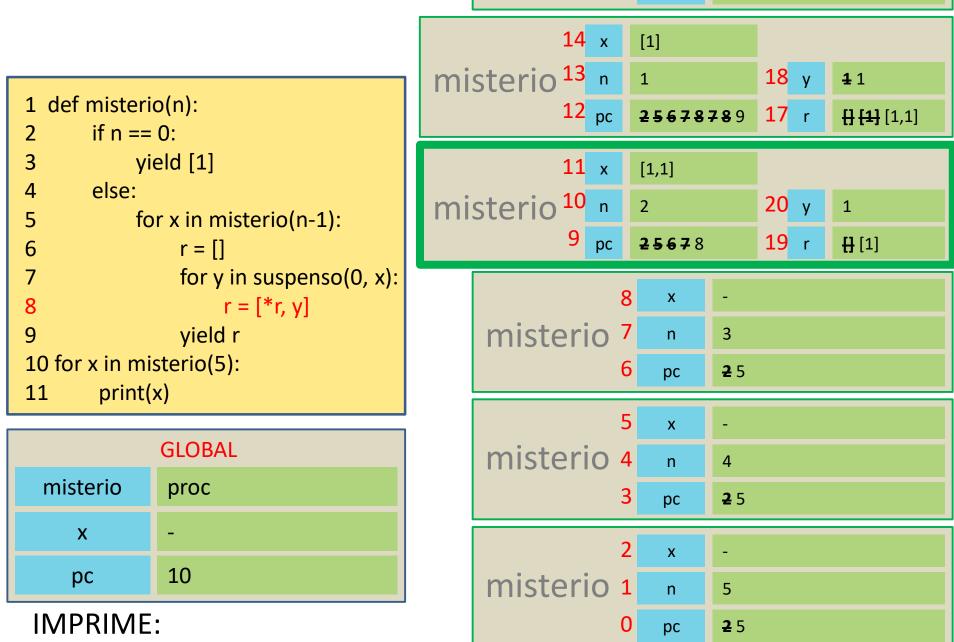
Examen 2 Problema 3 Misterio 0 n misterio Paso 21: Sale del ciclo for que itera en **2** 3 рс suspenso, pasa a la línea 9, ejecuta yield r, es decir, yield [1,1] 14 x [1] misterio 13 n 18 y 1 **1** 1 def misterio(n): 12 pc **2567878**9 17 r [] [1,1] if n == 0: 2 3 yield [1] 11 Х 4 else: misterio 10 2 n 5 for x in misterio(n-1): 6 **2** 5 рс r = []7 for y in suspenso(0, x): 8 8 Χ r = [*r, y]9 misterio 7 3 yield r n 10 for x in misterio(5): 6 **2** 5 рс print(x) 11 Х **GLOBAL** misterio 4 4 n misterio proc **2** 5 рс Χ Х 10 рс misterio 1 n 5 **IMPRIME: 2** 5 рс

Examen 2 Problema 3 Misterio 0 n misterio Paso 22: Vuelve al marco de misterio" y **2** 3 рс pasa a la línea 6, asigna a r 14 x [1] misterio 13 n 18 y 1 **1** 1 def misterio(n): 12 pc **2567878**9 17 r [] [1,1] if n == 0: 2 3 yield [1] 11 x [1,1] 4 else: misterio 10 n 5 for x in misterio(n-1): 19 r 6 **25**6 [] рс r = []7 for y in suspenso(0, x): 8 8 Х r = [*r, y]9 misterio ⁷ 3 yield r n 10 for x in misterio(5): **2** 5 рс print(x) 11 Х **GLOBAL** misterio 4 4 n misterio proc **2** 5 рс Χ Х 10 рс misterio 1 n 5 **IMPRIME: 2** 5 рс

Examen 2 Problema 3 Misterio 0 n misterio Paso 23: Pasa a la línea 7, entra al ciclo for, **2** 3 рс ejecuta *suspenso(0,x)*, es decir, suspenso(0,[1,1]) 14 x [1] misterio 13 n 18 y 1 **1** 1 def misterio(n): 12 pc **2567878**9 17 r [] [1,1] if n == 0: 2 3 yield [1] 11 [1,1] Х 4 else: misterio 10 n 20 y 5 for x in misterio(n-1): 19 r 6 2567 []рс r = []for y in suspenso(0, x): 8 8 Х r = [*r, y]9 misterio 7 3 yield r n 10 for x in misterio(5): 6 **2** 5 рс print(x) 11 Х **GLOBAL** misterio 4 4 n misterio proc **2** 5 рс Χ Х 10 рс misterio 1 n 5 **IMPRIME: 2** 5 рс

Examen 2 Problema 3 Misterio 0 n misterio Paso 24: Sabemos que suspenso(0,[1,1]) **2** 3 рс primero hace yield 0+1, luego hace yield 1+1, y luego hace yield 1 por lo tanto, y itera 14 x [1] en esos valores misterio 13 n 18 y 1 **1** 1 def misterio(n): 12 pc **2567878**9 17 r [] [1,1] if n == 0: 2 3 yield [1] 11 [1,1] Х 4 else: misterio 20 y 1 for x in misterio(n-1): 5 **256**7 19 r [] 6 рс r = []for y in suspenso(0, x): 8 8 Х r = [*r, y]9 misterio 7 3 yield r n 10 for x in misterio(5): **2** 5 рс print(x) 11 Х **GLOBAL** misterio 4 4 n misterio proc **2** 5 рс Χ Х 10 рс misterio 1 n 5 **IMPRIME: 2** 5 рс

Paso 25: Pasa a la línea 8, asigna a r



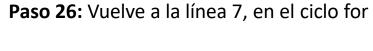
0

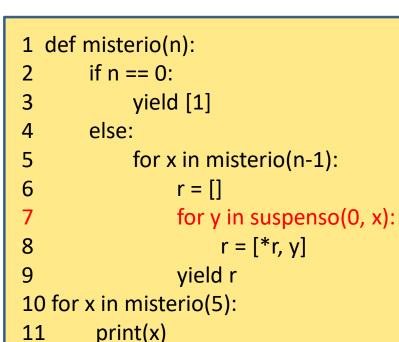
2 3

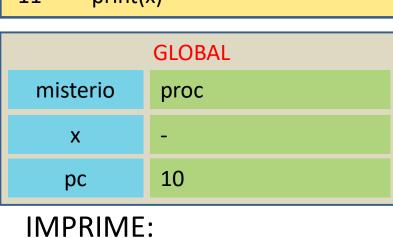
n

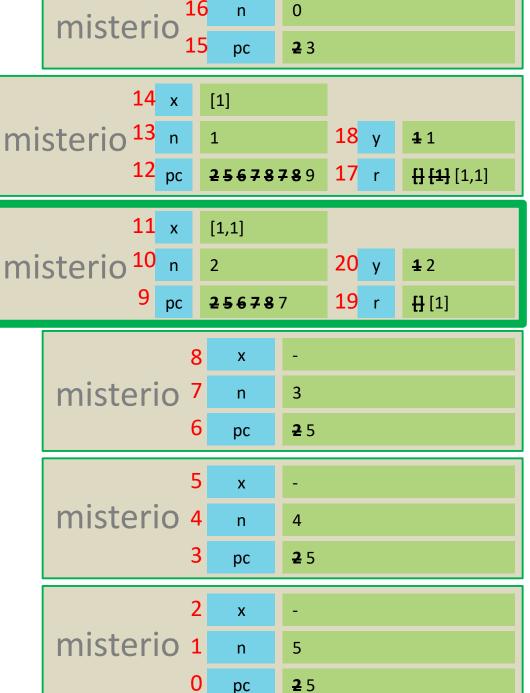
рс

misterio

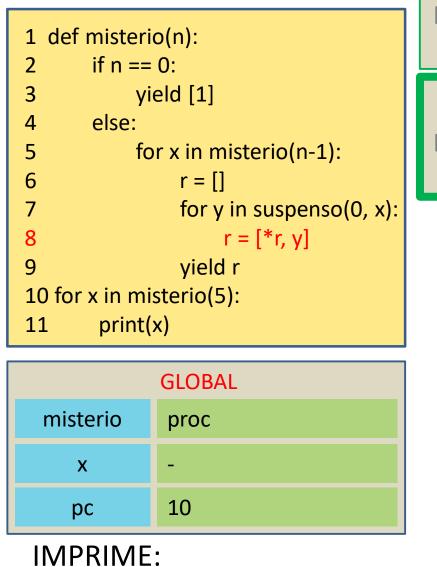


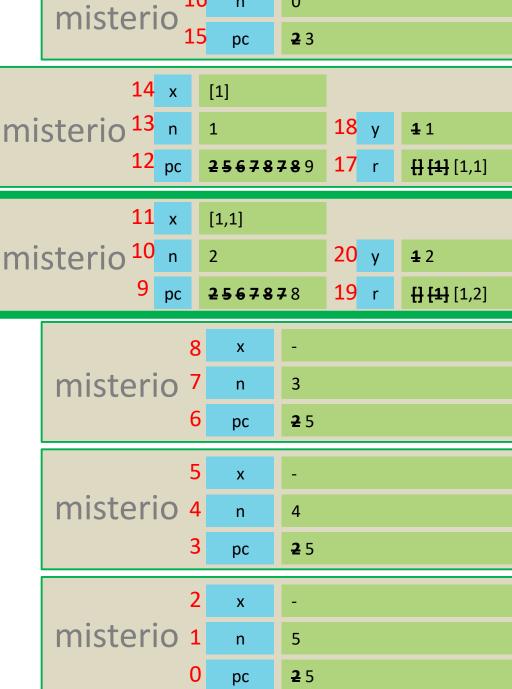






Paso 27: Pasa a la línea 8, asigna a r

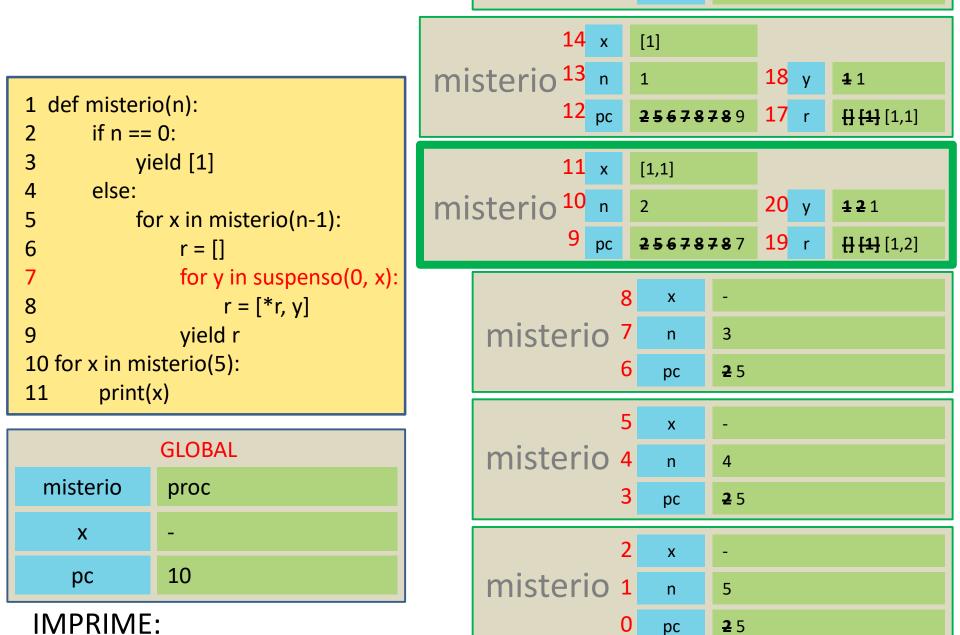




0

n

Paso 28: Vuelve a la línea 7, en el ciclo for



0

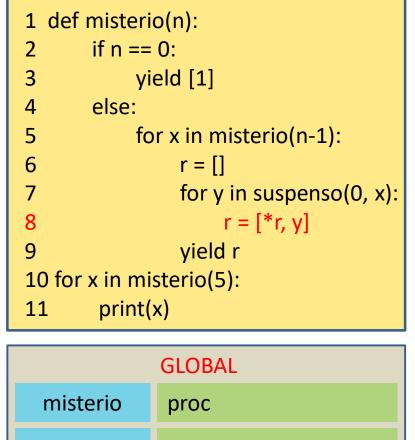
2 3

n

рс

misterio

Paso 29: Pasa a la línea 8, asigna a r

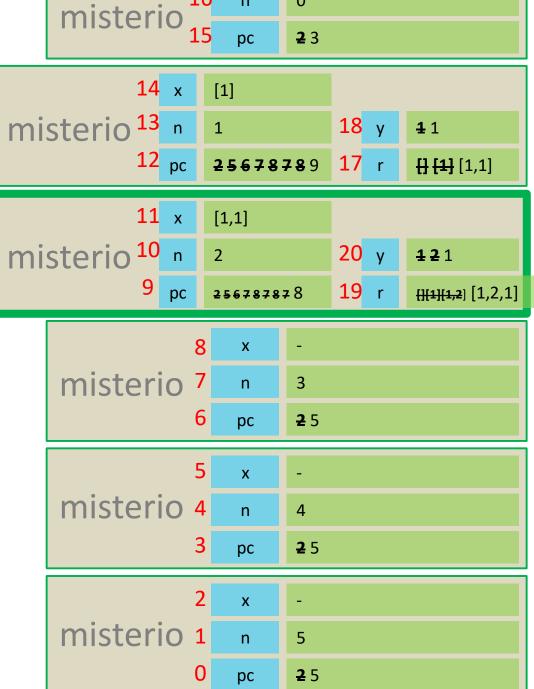


10

Χ

рс

IMPRIME:



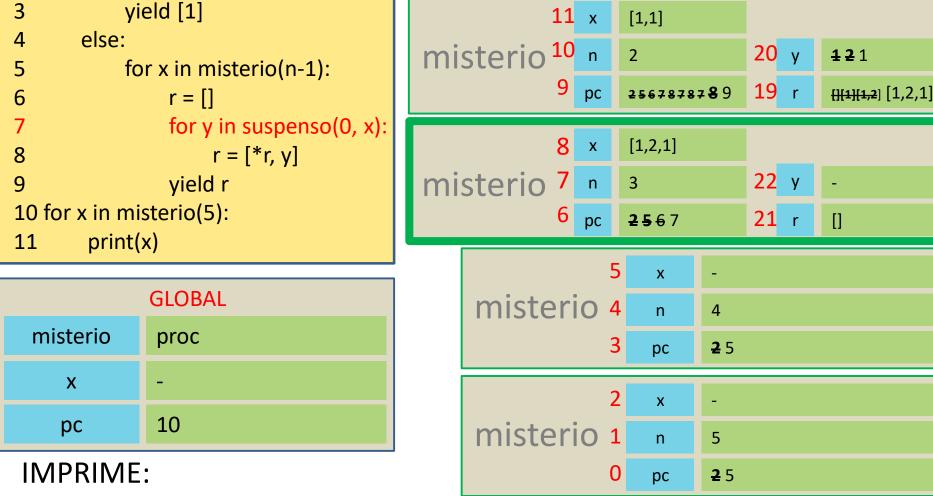
0

n

Examen 2 Problema 3 Misterio 0 n misterio Paso 30: Sale del ciclo for que itera en **2** 3 рс suspenso, pasa a la línea 9, ejecuta yield r, es decir, yield [1,2,1] 14 x [1] misterio 13 n 18 y 1 **1**1 def misterio(n): 12 pc **2567878**9 17 r [] [1,1] if n == 0: 2 3 yield [1] X 11 [1,1] 4 else: misterio 10 n 20 y **12**1 for x in misterio(n-1): 5 19 r 25678787**8**9 6 рс [][1,2][1,2][1,2,1]r = []7 for y in suspenso(0, x): 8 8 Х r = [*r, y]9 misterio 7 3 yield r n 10 for x in misterio(5): **2** 5 рс print(x) 11 Х **GLOBAL** misterio 4 4 n misterio proc **2** 5 рс Χ Х 10 рс misterio 1 n 5 **IMPRIME: 2** 5 рс

Examen 2 Problema 3 Misterio 0 n misterio Paso 31: Vuelve al marco de misterio" y **2** 3 рс pasa a la línea 6, asigna a r 14 x [1] misterio 13 n 18 y 1 **1** 1 def misterio(n): **12** pc **2567878**9 17 r [] [1,1] if n == 0: 2 3 yield [1] 11 x [1,1] 4 else: misterio 10 n 20 y **12**1 5 for x in misterio(n-1): 6 19 r **256787878**9 [][1][1,2][1,2,1]рс r = []7 for y in suspenso(0, x): 8 [1,2,1] 8 Х r = [*r, y]9 misterio ⁷ yield r 3 10 for x in misterio(5): 21 r **25**6 []рс print(x) 11 Х **GLOBAL** misterio 4 4 n misterio proc **2** 5 рс Χ Х 10 рс misterio 1 n 5 **IMPRIME: 2** 5 рс

Examen 2 Problema 3 Misterio n misterio Paso 32: Pasa a la línea 7, entra al ciclo for, рс ejecuta *suspenso(0,x)*, es decir, suspenso(0,[1,2,1]) 14 x [1] misterio 13 n 1 def misterio(n): **12** pc **2567878**9 if n == 0: 2 3 yield [1] 11 x [1,1] 4 else: misterio 10 n 5 for x in misterio(n-1): 6 рс r = []for y in suspenso(0, x): 8 8 Х r = [*r, y]9 misterio ⁷ 3 yield r 10 for x in misterio(5): рс print(x) 11 **GLOBAL** misterio 4 misterio proc



0

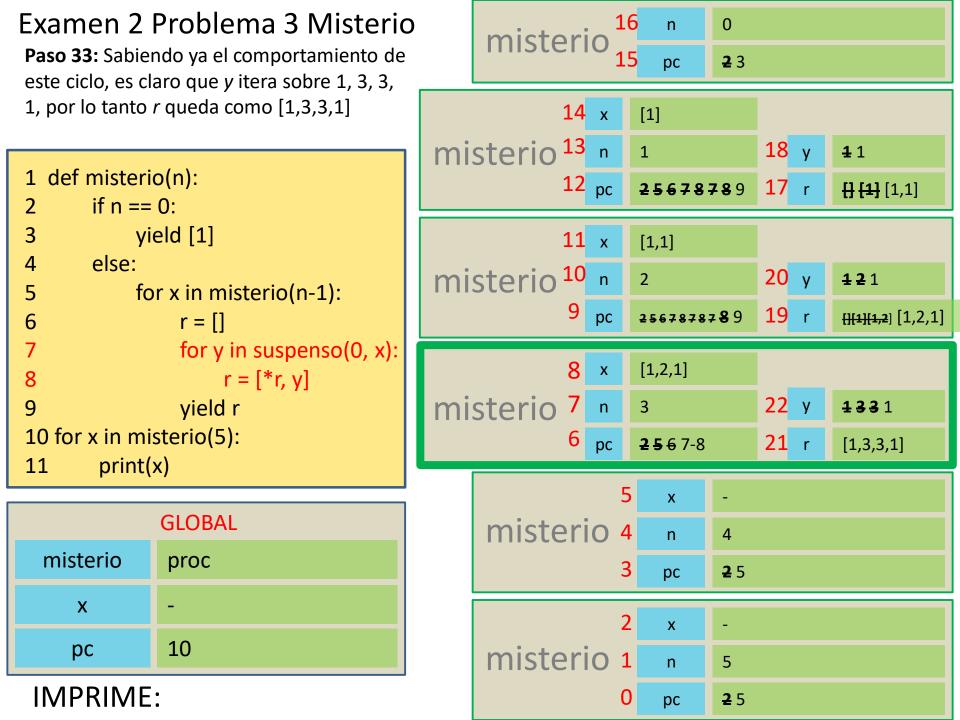
2 3

18 y

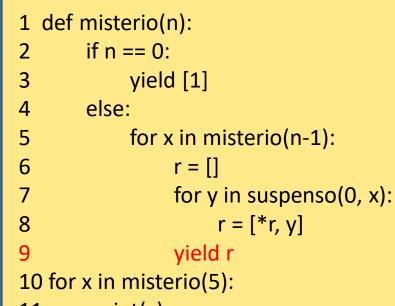
17 r

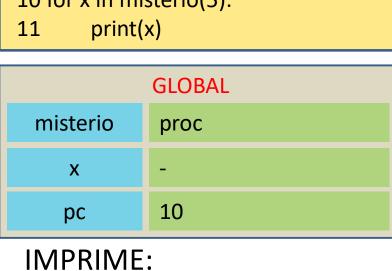
11

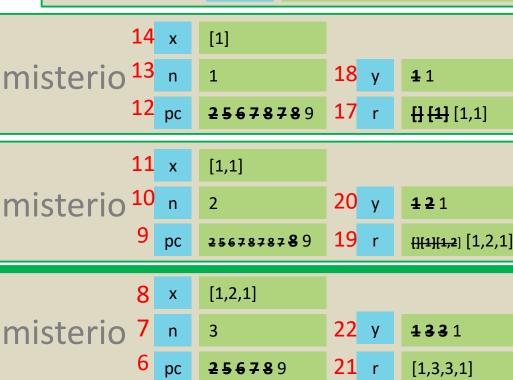
[] [1,1]



Examen 2 Problema 3 Misterio Paso 34: Sale del ciclo for que itera en suspenso, pasa a la línea 9, ejecuta yield r, es decir, yield [1,3,3,1] def misterio(n): if n == 0: 2 3 yield [1]







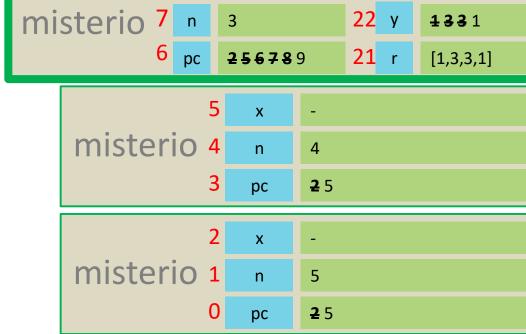
0

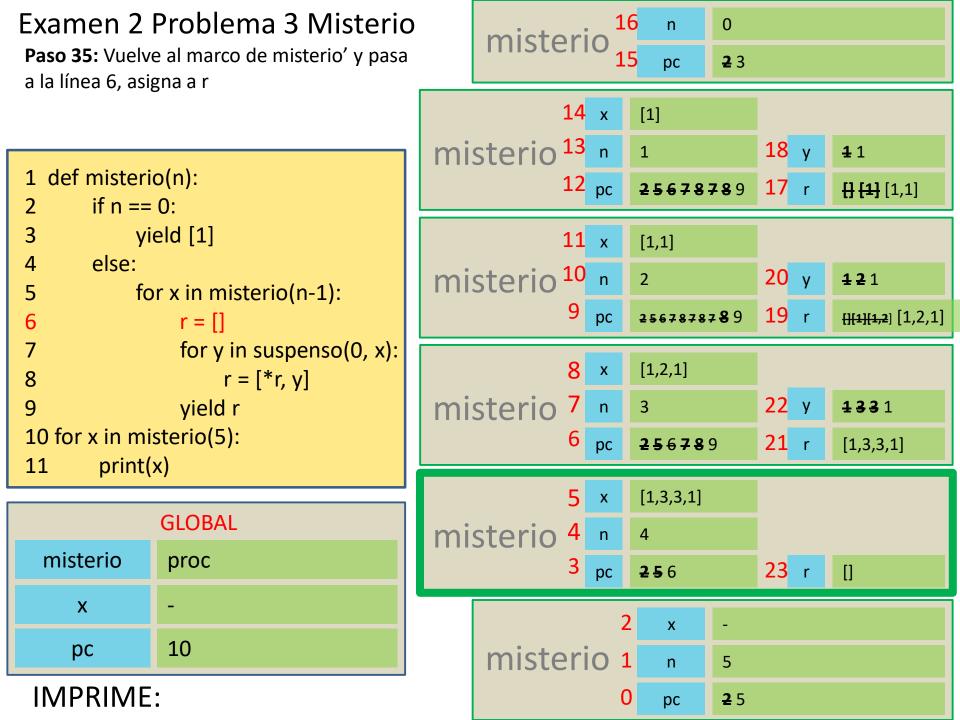
2 3

n

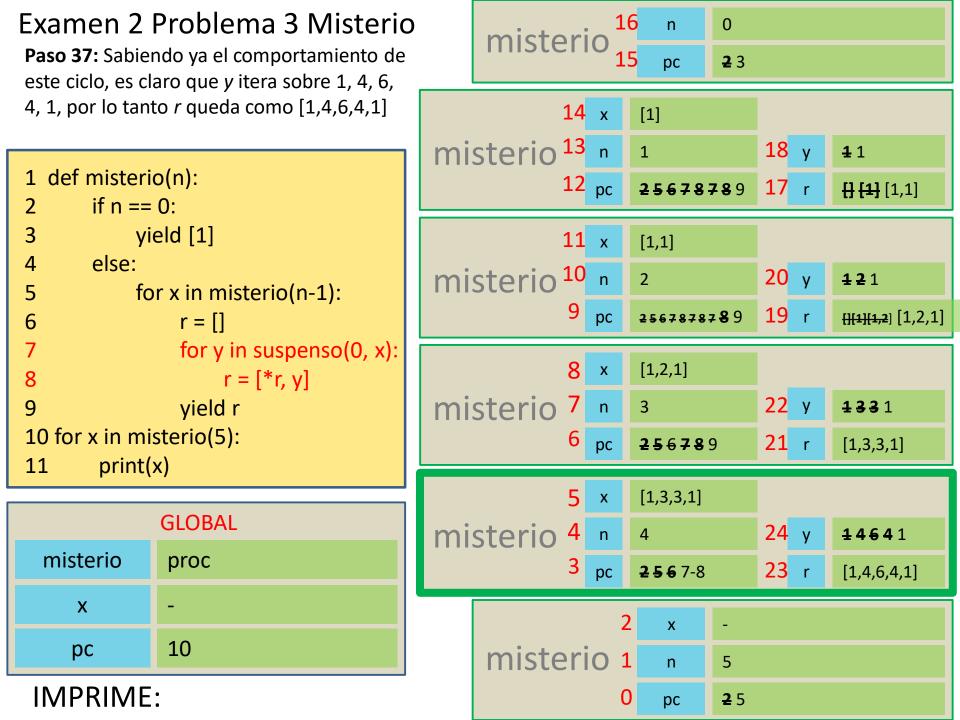
рс

misterio



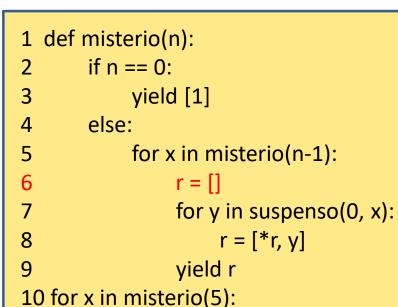


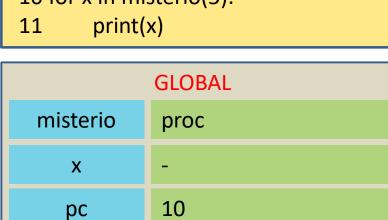
Examen 2 Problema 3 Misterio 0 n misterio Paso 36: Pasa a la línea 7, entra al ciclo for, **2** 3 рс ejecuta suspenso(0,x), es decir, suspenso(0,[1,3,3,1]) 14 x [1] misterio 13 n 18 y 1 **1**1 def misterio(n): **12** pc **2567878**9 17 r [] [1,1] if n == 0: 2 3 yield [1] 11 x [1,1] 4 else: misterio 10 n 20 y **12**1 5 for x in misterio(n-1): 19 r 25678787**8**9 6 [][1,2][1,2][1,2,1]рс r = []7 for y in suspenso(0, x): [1,2,1] Х 8 r = [*r, y]9 misterio ⁷ **22** y **133**1 yield r 3 10 for x in misterio(5): 6 **21** r **25678**9 [1,3,3,1] рс print(x) 11 5 [1,3,3,1] Χ **GLOBAL** misterio **24** y misterio proc 23 r **256**7 []рс Χ Х 10 рс misterio 1 n 5 **IMPRIME: 2** 5 рс



Examen 2 Problema 3 Misterio 0 n misterio Paso 38: Sale del ciclo for que itera en **2** 3 рс suspenso, pasa a la línea 9, ejecuta yield r, es decir, yield [1,4,6,4,1] 14 x [1] misterio 13 n 18 y 1 **1**1 def misterio(n): **12** pc **2567878**9 17 r [] [1,1] if n == 0: 2 3 yield [1] 11 x [1,1] 4 else: misterio 10 n 20 y **12**1 5 for x in misterio(n-1): 19 r 25678787**8**9 6 [][1,2][1,2][1,2,1]рс r = []7 for y in suspenso(0, x): [1,2,1] Х 8 r = [*r, y]**22** y 9 misterio ⁷ **133**1 yield r 3 10 for x in misterio(5): 6 **25678**9 **21** r [1,3,3,1] рс 11 print(x) 5 [1,3,3,1] Χ **GLOBAL** misterio 24 y 14641 misterio proc **25678**9 23 r рс [1,4,6,4,1] Χ Х 10 рс misterio 1 n 5 **IMPRIME: 2** 5 рс

Paso 39: Vuelve al marco de misterio y pasa a la línea 6, asigna a r





IMPRIME:

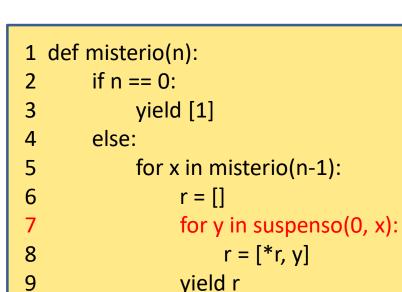
misterio **2** 3 рс 14 x [1] misterio 13 n 18 y 1 **1** 1 **12** pc **2567878**9 17 r [] [1,1] 11 x [1,1] misterio 10 n 20 y **12**1 19 r 25678787**8**9 [][1,2][1,2][1,2,1]рс [1,2,1] Х misterio ⁷ **22** y **133**1 3 21 r **25678**9 [1,3,3,1] рс [1,3,3,1] Χ misterio 24 y 14641 23 r **25678**9 рс [1,4,6,4,1]

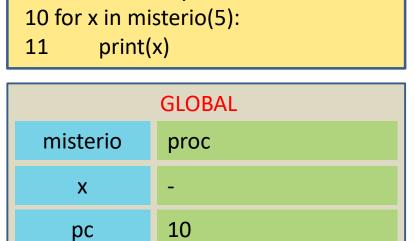
0

n

[1,4,6,4,1] Χ misterio 5 25 r **25**6 рс

Paso 40: Pasa a la línea 7, entra al ciclo for, ejecuta *suspenso(0,x)*, es decir, suspenso(0,[1,4,6,4,1])





IMPRIME:

misterio **2** 3 рс 14 x [1] misterio 13 n 18 y 1 **1**1 **12** pc **2567878**9 17 r [] [1,1] 11 x [1,1] misterio 10 n 20 y **12**1 19 r 25678787**8**9 [][1,2][1,2][1,2,1]рс [1,2,1] Х **22** y misterio ⁷ **133**1 3 256789 21 r [1,3,3,1] рс [1,3,3,1] Χ misterio 24 y 14641 23 r **25678**9 рс [1,4,6,4,1] [1,4,6,4,1] Χ

5

рс

2567

25 r

[]

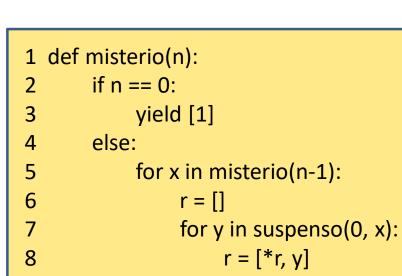
misterio

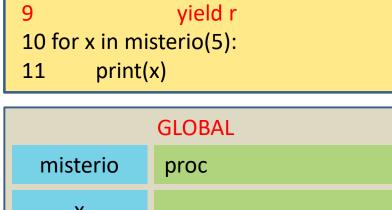
0

n

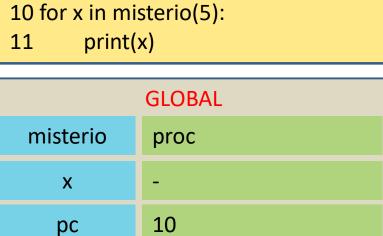
Examen 2 Problema 3 Misterio 0 n misterio Paso 41: Sabiendo ya el comportamiento de **2** 3 рс este ciclo, es claro que y itera sobre 1, 5, 10, 10, 5, 1, por lo tanto r queda como 14 x [1] [1,5,10,10,5,1] misterio 13 n 18 y 1 **1** 1 def misterio(n): **12** pc 25678789 17 r [] [1,1] if n == 0: 2 3 yield [1] 11 x [1,1] 4 else: misterio 10 n 20 y **42**1 5 for x in misterio(n-1): 19 r 25678787**8**9 6 [][1,2][1,2][1,2,1]рс r = []7 for y in suspenso(0, x): [1,2,1] 8 Х r = [*r, y]**22** y 9 misterio ⁷ **133**1 yield r 3 10 for x in misterio(5): 21 r **25678**9 [1,3,3,1] рс 11 print(x) [1,3,3,1] Χ **GLOBAL** misterio 24 v 14641 misterio proc 23 r **25678**9 рс [1,4,6,4,1] Χ [1,4,6,4,1] Χ 10 рс misterio 26 y 4 5 10 10 5 1 5 **IMPRIME:** 25 r [1,5,10,10,5,1] **256**7-8 рс

Paso 42: Sale del ciclo for que itera en suspenso, pasa a la línea 9, ejecuta yield r, es decir, yield [1,5,10,10,5,1]





IMPRIME:

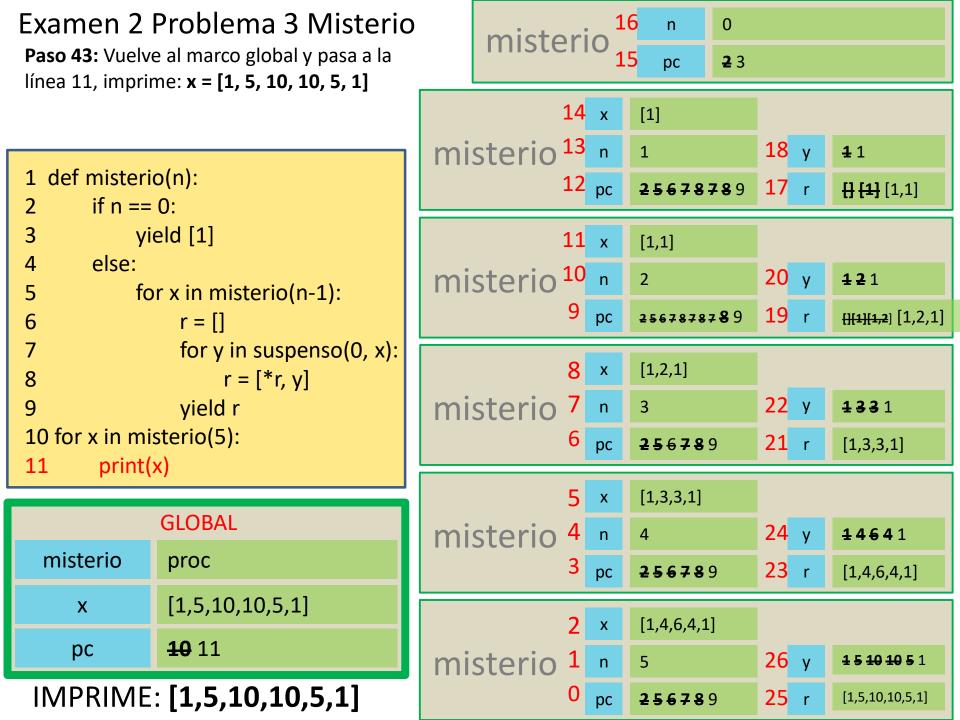


misterio **2** 3 рс 14 x [1] misterio 13 n 18 y 1 **1**1 12 pc **2567878**9 17 r [] [1,1] 11 x [1,1] misterio 10 n 20 y **12**1 19 r 25678787**8**9 [][1,2][1,2][1,2,1]рс [1,2,1] Х **22** y **133**1 3 21 r **25678**9 [1,3,3,1] рс [1,3,3,1] Χ 24 v 14641 23 r **25678**9 рс [1,4,6,4,1]

0

n

misterio ⁷ misterio [1,4,6,4,1] Χ misterio 26 y 4 5 10 10 5 1 5 25 r [1,5,10,10,5,1] **25678**9 рс



Examen 2 Problema 3 Misterio Paso 44: Vuelve a la línea 10, en el ciclo for

for x in misterio(n-1):

for y in suspenso(0, x):

r = [*r, y]

r = []

yield r

GLOBAL

10 11 10

IMPRIME: [1,5,10,10,5,1]

[1,5,10,10,5,1]

proc

def misterio(n):

else:

2

3

4

5

6

7

8

9

11

misterio

Χ

рс

if n == 0:

10 for x in misterio(5):

print(x)

yield [1]



misterio

misterio 13 n

misterio 10 n

misterio ⁷

misterio

misterio

14 x

12 pc

11 x

рс

Х

рс

Χ

рс

Χ

2 3

n

рс

25678789

[1]

1

[1,1]

[1,2,1]

256789

[1,3,3,1]

256789

[1,4,6,4,1]

256789

3

0

18 y

17 r

- - **1** 1
 - [] [1,1]
 - **12**1
- **133**1

- **21** r

- 25678787**8**9
- 20 y
- 19 r
- [][1,2][1,2][1,2,1]
- **22** y

 - [1,3,3,1]

 - **24** y

 - 14641

15101051

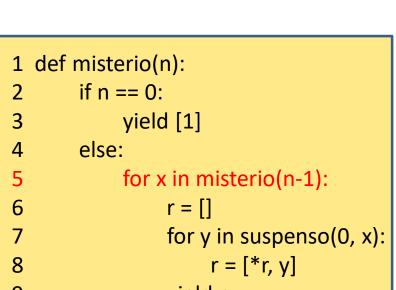
[1,5,10,10,5,1]

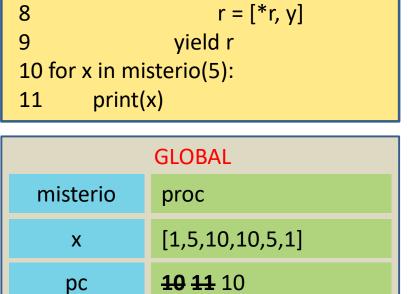
23 r [1,4,6,4,1]

26 y

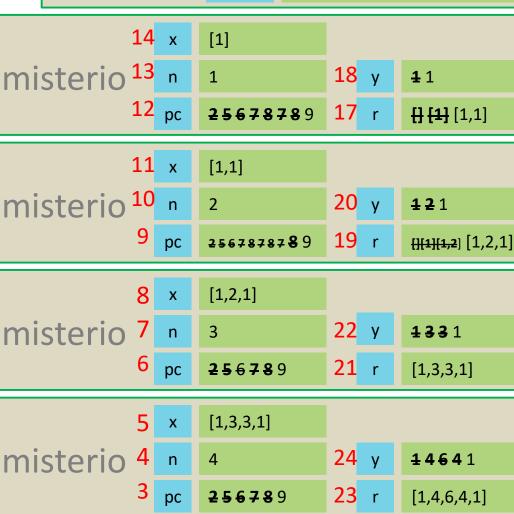
25 r

Examen 2 Problema 3 Misterio Paso 45: Vuelve al marco de misterio, vuelve a la línea 5 en el ciclo for def misterio(n): if n == 0: 2 3 yield [1]





IMPRIME: [1,5,10,10,5,1]



0

2 3

n

рс

misterio

misterio [1,4,6,4,1] Χ misterio 26 y **1510105**1 25 r [1,5,10,10,5,1] 2567895 рс

for x in misterio(n-1):

r = [*r, y]

r = []

yield r

GLOBAL

[1,5,10,10,5,1]

proc

Paso 46: Vuelve al marco de misterio', vuelve a la línea 5 en el ciclo for

def misterio(n):

else:

2

3

4

5

6

7

8

9

11

misterio

Χ

if n == 0:

10 for x in misterio(5):

print(x)

yield [1]



14 x misterio 13 n 11 Х

misterio

18 y **2567878**9

0

2 3

n

рс

[1]

1

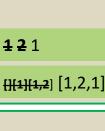
[1,1]

[1,2,1]

25678787**8**9

17 r [] [1,1] 20 y 19 r

1 1



1331

[1,3,3,1]

for y in suspenso(0, x):

misterio 10 n рс Х misterio ⁷ рс Χ misterio рс

misterio

256789 [1,3,3,1] **256789**5 [1,4,6,4,1] Χ

2567895

24 v 23 r

22 y

21 r

26 y

25 r

14641 [1,4,6,4,1] **1510105**1 [1,5,10,10,5,1]

10 11 10 рс IMPRIME: [1,5,10,10,5,1]

for x in misterio(n-1):

r = [*r, y]

r = []

yield r

Paso 47: Vuelve al marco de misterio", vuelve a la línea 5 en el ciclo for

def misterio(n):

else:

2

3

4

5

6

7

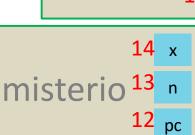
8

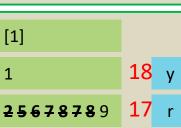
9

if n == 0:

yield [1]







20 y

0

2 3

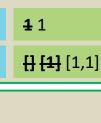
n

рс

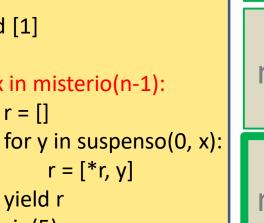
[1]

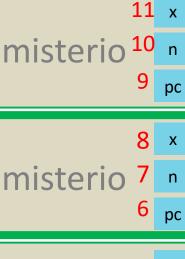
1

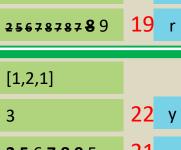
[1,1]

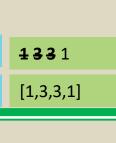


421

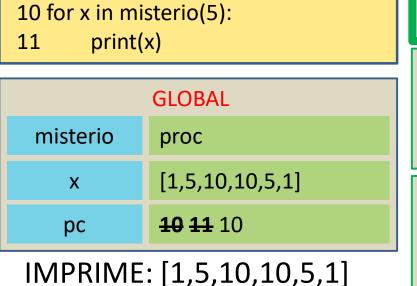


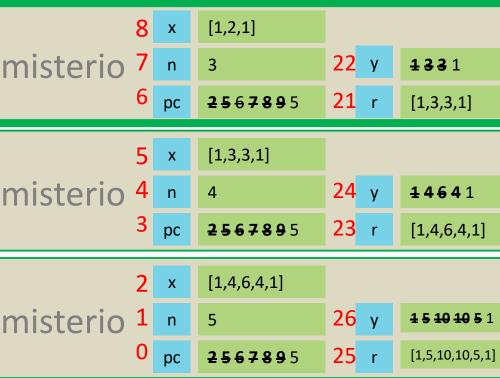






[][1,2][1,2][1,2,1]





for x in misterio(n-1):

for y in suspenso(0, x):

r = [*r, y]

r = []

yield r

GLOBAL

10 11 10

IMPRIME: [1,5,10,10,5,1]

[1,5,10,10,5,1]

proc

Paso 48: Vuelve al marco de misterio", vuelve a la línea 5 en el ciclo for

def misterio(n):

else:

2

3

4

5

6

7

8

9

11

misterio

Χ

рс

if n == 0:

10 for x in misterio(5):

print(x)

yield [1]



14 x misterio 13 n **12** pc

11 x

18 y **2567878**9 17 r

0

2 3

n

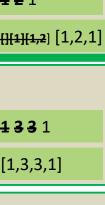
рс

[1]

[1,1]

[] [1] [1,1]

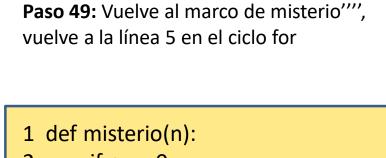
1 1

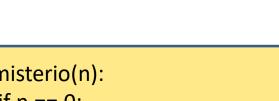


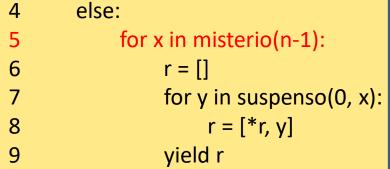


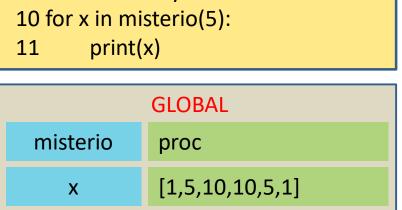
misterio ¹⁰	n	2	20 y	12 1
9	рс	2567878789 5	19 r	[][1,2,2] [1,2,2]
8	х	[1,2,1]		
misterio 7	n	3	22 y	133 1
	рс	256789 5	21 r	[1,3,3,1]
5	х	[1,3,3,1]		
misterio 4 3	n	4	24 y	1464 1
	рс	256789 5	23 r	[1,4,6,4,1]
2	Х	[1,4,6,4,1]		
misterio 1 0	n	5	26 y	1510105 1
	рс	256789 5	25 r	[1,5,10,10,5,1]

Paso 49: Vuelve al marco de misterio"",



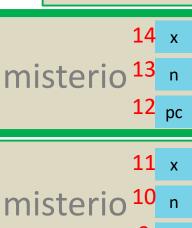




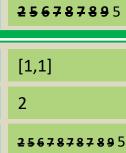


10 11 10

рс



misterio



[1,2,1]

2567895

рс

Х

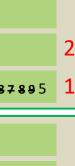
рс

Χ

n

рс

[1]



0

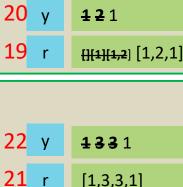
2 3

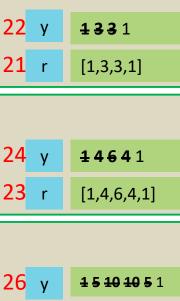
18 y

17 r

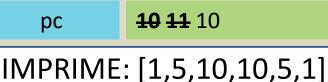
1 1

[] [1,1]



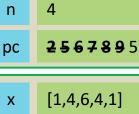


[1,5,10,10,5,1]





misterio ⁷



2567895

[1,3,3,1]

24 v 23 r

25 r

Examen 2 Problema 3 Misterio 0 n misterio Paso 50: Vuelve al marco de misterio"", 23 рс termina la ejecución de misterio() 14 x [1] misterio 13 n **18** y 1 **1** 1 def misterio(n): 12 pc **25678789**5 17 r [] [1,1] if n == 0: 2 3 yield [1] 11 x [1,1] 4 else: misterio 10 n 20 y **12**1 5 for x in misterio(n-1): 19 r 6 25678787895 [][1,2][1,2][1,2,1]рс r = []7 for y in suspenso(0, x): [1,2,1] Х 8 r = [*r, y]misterio ⁷ **22** y 9 1331 yield r 3 10 for x in misterio(5): 2567895 **21** r [1,3,3,1] рс 11 print(x) [1,3,3,1] Χ **GLOBAL** misterio 24 v 14641 misterio proc **256789**5 23 r рс [1,4,6,4,1] [1,5,10,10,5,1] Χ [1,4,6,4,1] Χ **10 11** 10 рс misterio 26 y **1510105**1 IMPRIME: [1,5,10,10,5,1] 25 r [1,5,10,10,5,1] 2567895

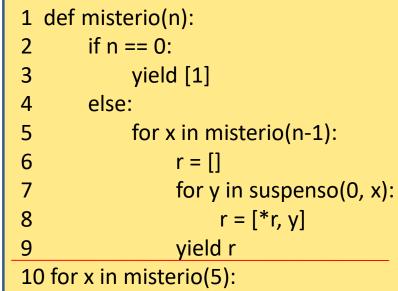
Examen 2 Problema 3 Misterio Paso 51: Se libera el marco de misterio"". Vuelve al marco de misterio"", termina el ciclo for, termina la ejecución de misterio() 14 x [1] misterio 13 n 18 _V 1 **1** 1 def misterio(n): **12** pc 256787895 17 r [] [1,1] if n == 0: 2 3 yield [1] 11 [1,1] Х 4 else: misterio 10 n 20 v **42**1 5 for x in misterio(n-1): 19 r 25678787895 [][1,2][1,2][1,2,1]рс 6 r = []7 for y in suspenso(0, x): [1,2,1] Х 8 r = [*r, y]misterio 7 **22** y 9 1331 yield r 3 10 for x in misterio(5): **21** r **256789**5 [1,3,3,1] рс 11 print(x) [1,3,3,1] Χ **GLOBAL** misterio 24 v 14641 misterio proc **256789**5 23 r рс [1,4,6,4,1] [1,5,10,10,5,1]Χ [1,4,6,4,1] Χ **10 11** 10 рс misterio 26 y **1510105**1

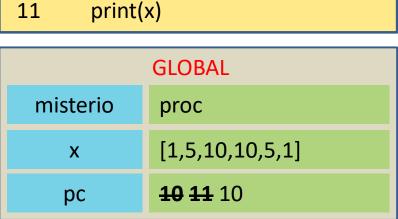
IMPRIME: [1,5,10,10,5,1] 25 r [1,5,10,10,5,1] 2567895

Examen 2 Problema 3 Misterio Paso 52: Se libera el marco de misterio". Vuelve al marco de misterio", termina el ciclo for, termina la ejecución de misterio() def misterio(n): 256787895 if n == 0: 2 3 yield [1] 11 Х [1,1] else: 4 misterio 20 v **42**1 for x in misterio(n-1): 5 19 r 25678787895 [][1,2][1,2][1,2,1]рс 6 r = []for y in suspenso(0, x): [1,2,1] Х 8 r = [*r, y]**22** y misterio ⁷ 9 1331 yield r 3 10 for x in misterio(5): 21 r **256789**5 [1,3,3,1] рс 11 print(x) [1,3,3,1] Χ **GLOBAL** misterio 24 v 14641 misterio proc **256789**5 23 r рс [1,4,6,4,1] [1,5,10,10,5,1]Χ [1,4,6,4,1] Χ **10 11** 10 рс misterio 26 y **1510105**1 IMPRIME: [1,5,10,10,5,1] 25 r [1,5,10,10,5,1] 2567895

Examen 2 Problema 3 Misterio Paso 53: Se libera el marco de misterio". Vuelve al marco de misterio", termina el ciclo for, termina la ejecución de misterio() def misterio(n): 256787895 if n == 0: 3 yield [1] [1,1]else: for x in misterio(n-1): 5 6 r = []for y in suspenso(0, x): [1,2,1]Х 8 r = [*r, y]misterio **22** y 9 1331 yield r 3 10 for x in misterio(5): **21** r 2567895 [1,3,3,1] рс 11 print(x) [1,3,3,1] Χ **GLOBAL** misterio 24 v 14641 misterio proc **256789**5 23 r рс [1,4,6,4,1] [1,5,10,10,5,1]Χ [1,4,6,4,1] **10 11** 10 рс misterio 26 y **1510105**1 IMPRIME: [1,5,10,10,5,1] 25 r [1,5,10,10,5,1] 2567895

Paso 54: Se libera el marco de misterio". Vuelve al marco de misterio, termina el ciclo for, termina la ejecución de misterio()

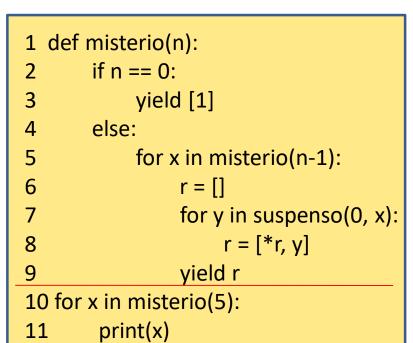


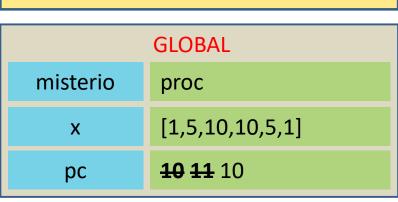


IMPRIME: [1,5,10,10,5,1]



Paso 55: Se libera el marco de misterio'. Vuelve al marco de misterio, termina el ciclo for, termina la ejecución de misterio()



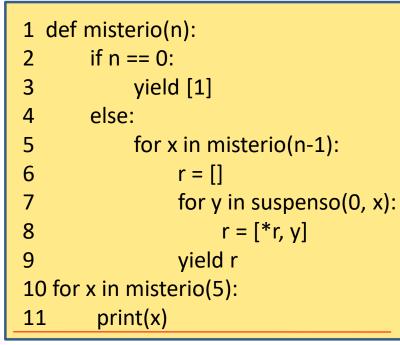


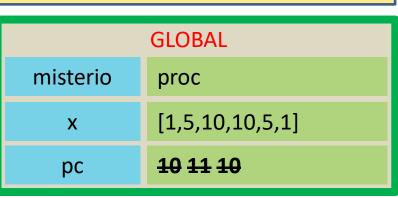
IMPRIME: [1,5,10,10,5,1]



Paso 56: Se libera el marco de misterio.

Vuelve al marco global, termina el ciclo for





IMPRIME: [1,5,10,10,5,1]



Paso 57: Termina la ejecución del programa

