

# Pregunta 3b

...

El Iterador Contraataca

## Código en Python

## Imprime

```
def misterio(ls):
0   if ls:
1       for m in misterio(ls[1:]):
2           for i in ins(ls[0], m):
3               yield i
4       else:
5           yield []
5 for m in misterio([1,2,3]):
6     print m
```

Global	
misterio	proc
m	-
pc	5

# Código en Python

# Imprime

```
def misterio(ls):
0   if ls:
1       for m in misterio(ls[1:]):
2           for i in ins(ls[0], m):
3               yield i
4       else:
5           yield []
6   for m in misterio([1,2,3]):
7       print m
```

Global	
misterio	proc
m	-
pc	5

misterio	ls	[1,2,3]
	pc	0

Código en Python

Imprime

```
def misterio(ls):
0   if ls:
1       for m in misterio(ls[1:]):
2           for i in ins(ls[0], m):
3               yield i
        else:
4           yield []

5 for m in misterio([1,2,3]):
6     print m
```

Global	
misterio	proc
m	-
pc	5

misterio	m	-
	ls	[1,2,3]
	pc	Ø 1

# Código en Python

# Imprime

```
def misterio(ls):
0   if ls:
1       for m in misterio(ls[1:]):
2           for i in ins(ls[0], m):
3               yield i
4       else:
5           yield []
6   for m in misterio([1,2,3]):
7       print m
```

Global	
misterio	proc
m	-
pc	5

misterio	ls	[2,3]
	pc	0
misterio	m	-
	ls	[1,2,3]
	pc	Ø 1

# Código en Python

# Imprime

```
def misterio(ls):
0   if ls:
1       for m in misterio(ls[1:]):
2           for i in ins(ls[0], m):
3               yield i
    else:
4       yield []

5 for m in misterio([1,2,3]):
6     print m
```

Global	
misterio	proc
m	-
pc	5

misterio	m	-
	ls	[2,3]
	pc	Ø 1
misterio	m	-
	ls	[1,2,3]
	pc	Ø 1

# Código en Python

# Imprime

```
def misterio(ls):
0   if ls:
1       for m in misterio(ls[1:]):
2           for i in ins(ls[0], m):
3               yield i
4       else:
5           yield []
6   for m in misterio([1,2,3]):
7       print m
```

Global	
misterio	proc
m	-
pc	5

misterio	ls	[3]
	pc	0
misterio	m	-
	ls	[2,3]
	pc	Ø 1
misterio	m	-
	ls	[1,2,3]
	pc	Ø 1

# Código en Python

```
def misterio(ls):
0   if ls:
1       for m in misterio(ls[1:]):
2           for i in ins(ls[0], m):
3               yield i
4       else:
5           yield []
6   for m in misterio([1,2,3]):
7       print m
```

Global	
misterio	proc
m	-
pc	5

# Imprime

misterio	m	-
	ls	[3]
	pc	<del>0</del> 1
misterio	m	-
	ls	[2,3]
	pc	<del>0</del> 1
misterio	m	-
	ls	[1,2,3]
	pc	<del>0</del> 1



# Código en Python

# Imprime

```
def misterio(ls):
0   if ls:
1       for m in misterio(ls[1:]):
2           for i in ins(ls[0], m):
3               yield i
4       else:
5           yield []
6   for m in misterio([1,2,3]):
7       print m
```

Global	
misterio	proc
m	-
pc	5

misterio	ls	[]
	pc	0
misterio	m	-
	ls	[3]
	pc	<del>0</del> 1
misterio	m	-
	ls	[2,3]
	pc	<del>0</del> 1
misterio	m	-
	ls	[1,2,3]
	pc	<del>0</del> 1

Código en Python

```
def misterio(ls):
0   if ls:
1       for m in misterio(ls[1:]):
2           for i in ins(ls[0], m):
3               yield i
         else:
4           yield []
5   for m in misterio([1,2,3]):
6       print m
```

Global	
misterio	proc
m	-
pc	5

Imprime

misterio	ls	[]
	pc	<del>0</del> 4
misterio	m	-
	ls	[3]
	pc	<del>0</del> 1
misterio	m	-
	ls	[2,3]
	pc	<del>0</del> 1
misterio	m	-
	ls	[1,2,3]
	pc	<del>0</del> 1

# Código en Python

```
def misterio(ls):
0   if ls:
1       for m in misterio(ls[1:]):
2           for i in ins(ls[0], m):
3               yield i
4       else:
5           yield []
6   for m in misterio([1,2,3]):
7       print m
```

Global	
misterio	proc
m	-
pc	5

# Imprime

misterio	ls	[]
	pc	<del>0</del> 4
misterio	i	[3]
	m	[]
	ls	[3]
	pc	<del>0</del> 1 2
misterio	m	-
	ls	[2,3]
	pc	<del>0</del> 1
misterio	m	-
	ls	[1,2,3]
	pc	<del>0</del> 1

Código en Python

Imprime

```
def misterio(ls):
0   if ls:
1       for m in misterio(ls[1:]):
2           for i in ins(ls[0], m):
3               yield i
4       else:
5           yield []
6   for m in misterio([1,2,3]):
7       print m
```

Global	
misterio	proc
m	-
pc	5

misterio	ls	[]
	pc	<del>0</del> 4
misterio	i	[3]
	m	[]
	ls	[3]
	pc	<del>0</del> 1 <del>2</del> 3
misterio	m	-
	ls	[2,3]
	pc	<del>0</del> 1
misterio	m	-
	ls	[1,2,3]
	pc	<del>0</del> 1

# Código en Python

```
def misterio(ls):
0   if ls:
1       for m in misterio(ls[1:]):
2           for i in ins(ls[0], m):
3               yield i
4       else:
5           yield []
6   for m in misterio([1,2,3]):
7       print m
```

Global	
misterio	proc
m	-
pc	5

# Imprime

misterio	ls	[]
	pc	<del>0</del> 4
misterio	i	[3]
	m	[]
	ls	[3]
	pc	<del>0</del> <del>1</del> <del>2</del> 3
misterio	i	[2,3] [3,2]
	m	[3]
	ls	[2,3]
	pc	<del>0</del> <del>1</del> 2
misterio	m	-
	ls	[1,2,3]
	pc	<del>0</del> 1

Código en Python

```
def misterio(ls):
0   if ls:
1       for m in misterio(ls[1:]):
2           for i in ins(ls[0], m):
3               yield i
4       else:
5           yield []
6   for m in misterio([1,2,3]):
7       print m
```

Global	
misterio	proc
m	-
pc	5

Imprime

misterio	ls	[]
	pc	<del>0</del> 4
misterio	i	[3]
	m	[]
	ls	[3]
	pc	<del>0</del> <del>1</del> <del>2</del> 3
misterio	i	[2,3] [3,2]
	m	[3]
	ls	[2,3]
	pc	<del>0</del> <del>1</del> <del>2</del> 3
misterio	m	-
	ls	[1,2,3]
	pc	<del>0</del> 1

Código en Python

```
def misterio(ls):
0   if ls:
1       for m in misterio(ls[1:]):
2           for i in ins(ls[0], m):
3               yield i
4       else:
5           yield []
6   for m in misterio([1,2,3]):
7       print m
```

Global	
misterio	proc
m	-
pc	5

Imprime

misterio	ls	[]
	pc	<del>0</del> 4
misterio	i	[3]
	m	[]
	ls	[3]
	pc	<del>0</del> <del>1</del> <del>2</del> 3
misterio	i	[2,3] [3,2]
	m	[3]
	ls	[2,3]
	pc	<del>0</del> <del>1</del> <del>2</del> 3
misterio	i	[1, 2, 3] [2, 1, 3] [2, 3, 1]
	m	[2,3]
	ls	[1,2,3]
	pc	<del>0</del> <del>1</del> 2

# Código en Python

```
def misterio(ls):
0   if ls:
1       for m in misterio(ls[1:]):
2           for i in ins(ls[0], m):
3               yield i
4       else:
5           yield []
6   for m in misterio([1,2,3]):
7       print m
```

Global	
misterio	proc
m	-
pc	5

# Imprime

misterio	ls	[]
	pc	<del>0</del> 4
misterio	i	[3]
	m	[]
	ls	[3]
	pc	<del>0</del> <del>1</del> <del>2</del> 3
misterio	i	[2,3] [3,2]
	m	[3]
	ls	[2,3]
	pc	<del>0</del> <del>1</del> <del>2</del> 3
misterio	i	[1, 2, 3] [2, 1, 3] [2, 3, 1]
	m	[2,3]
	ls	[1,2,3]
	pc	<del>0</del> <del>1</del> <del>2</del> 3



Código en Python

```
def misterio(ls):
0   if ls:
1       for m in misterio(ls[1:]):
2           for i in ins(ls[0], m):
3               yield i
4       else:
5           yield []
6   for m in misterio([1,2,3]):
7       print m
```

Global	
misterio	proc
m	[1,2,3]
pc	<del>5</del> 6

Imprime

[1,2,3]

misterio	ls	[]
	pc	<del>0</del> 4
misterio	i	[3]
	m	[]
	ls	[3]
	pc	<del>0</del> 1 <del>2</del> 3
misterio	i	[2,3] [3,2]
	m	[3]
	ls	[2,3]
	pc	<del>0</del> 1 <del>2</del> 3
misterio	i	[1, 2, 3] [2, 1, 3] [2, 3, 1]
	m	[2,3]
	ls	[1,2,3]
	pc	<del>0</del> 1 <del>2</del> 3

Código en Python

```
def misterio(ls):
0   if ls:
1       for m in misterio(ls[1:]):
2           for i in ins(ls[0], m):
3               yield i
4       else:
5           yield []
6   for m in misterio([1,2,3]):
7       print m
```

Imprime

[1,2,3]

Global	
misterio	proc
m	[1,2,3]
pc	<del>5</del> 6 <del>5</del>

misterio	ls	[]
	pc	<del>0</del> 4
misterio	i	[3]
	m	[]
	ls	[3]
	pc	<del>0</del> 1 <del>2</del> 3
misterio	i	[2,3] [3,2]
	m	[3]
	ls	[2,3]
	pc	<del>0</del> 1 <del>2</del> 3
misterio	i	[1, 2, 3] [2, 1, 3] [2, 3, 1]
	m	[2,3]
	ls	[1,2,3]
	pc	<del>0</del> 1 <del>2</del> 3

Código en Python

```
def misterio(ls):
0   if ls:
1       for m in misterio(ls[1:]):
2           for i in ins(ls[0], m):
3               yield i
4       else:
5           yield []
6   for m in misterio([1,2,3]):
7       print m
```

Global	
misterio	proc
m	[1,2,3]
pc	<del>5</del> 6 5

Imprime

[1,2,3]

misterio	ls	[]
	pc	<del>0</del> 4
misterio	i	[3]
	m	[]
	ls	[3]
	pc	<del>0</del> 1 <del>2</del> 3
misterio	i	[2,3] [3,2]
	m	[3]
	ls	[2,3]
	pc	<del>0</del> 1 <del>2</del> 3
misterio	i	[1, 2, 3] [2, 1, 3] [2, 3, 1]
	m	[2,3]
	ls	[1,2,3]
	pc	<del>0</del> 1 <del>2</del> 3 2

Código en Python

```
def misterio(ls):
0   if ls:
1       for m in misterio(ls[1:]):
2           for i in ins(ls[0], m):
3               yield i
4       else:
5           yield []
6   for m in misterio([1,2,3]):
7       print m
```

Global	
misterio	proc
m	[1,2,3]
pc	<del>5</del> <del>6</del> 5

Imprime

[1,2,3]

misterio	ls	[]
	pc	<del>0</del> 4
misterio	i	[3]
	m	[]
	ls	[3]
	pc	<del>0</del> <del>1</del> <del>2</del> 3
misterio	i	[2,3] [3,2]
	m	[3]
	ls	[2,3]
	pc	<del>0</del> <del>1</del> <del>2</del> 3
misterio	i	[1, 2, 3] [2, 1, 3] [2, 3, 1]
	m	[2,3]
	ls	[1,2,3]
	pc	<del>0</del> <del>1</del> <del>2</del> <del>3</del> <del>2</del> 3

Código en Python

```
def misterio(ls):
0   if ls:
1       for m in misterio(ls[1:]):
2           for i in ins(ls[0], m):
3               yield i
4       else:
5           yield []
6   for m in misterio([1,2,3]):
    print m
```

Global	
misterio	proc
m	[2,1,3]
pc	<del>5</del> <del>6</del> <del>5</del> <del>6</del>

Imprime

[1,2,3]  
[2,1,3]

misterio	ls	[]
	pc	<del>0</del> 4
misterio	i	[3]
	m	[]
	ls	[3]
	pc	<del>0</del> <del>1</del> <del>2</del> 3
misterio	i	[2,3] [3,2]
	m	[3]
	ls	[2,3]
	pc	<del>0</del> <del>1</del> <del>2</del> 3
misterio	i	[1, 2, 3] [2, 1, 3] [2, 3, 1]
	m	[2,3]
	ls	[1,2,3]
	pc	<del>0</del> <del>1</del> <del>2</del> <del>3</del> <del>2</del> 3

Código en Python

```
def misterio(ls):
0   if ls:
1       for m in misterio(ls[1:]):
2           for i in ins(ls[0], m):
3               yield i
4       else:
5           yield []
6   for m in misterio([1,2,3]):
7       print m
```

Global	
misterio	proc
m	[2,1,3]
pc	<del>5</del> <del>6</del> <del>5</del> <del>6</del> 5

Imprime

[1,2,3]  
[2,1,3]

misterio	ls	[]
	pc	<del>0</del> 4
misterio	i	[3]
	m	[]
	ls	[3]
	pc	<del>0</del> <del>1</del> <del>2</del> 3
misterio	i	[2,3] [3,2]
	m	[3]
	ls	[2,3]
	pc	<del>0</del> <del>1</del> <del>2</del> 3
misterio	i	[1, 2, 3] [2, 1, 3] [2, 3, 1]
	m	[2,3]
	ls	[1,2,3]
	pc	<del>0</del> <del>1</del> <del>2</del> <del>3</del> <del>2</del> 3

Código en Python

```
def misterio(ls):
0   if ls:
1       for m in misterio(ls[1:]):
2           for i in ins(ls[0], m):
3               yield i
4       else:
5           yield []
6   for m in misterio([1,2,3]):
7       print m
```

Imprime

[1,2,3]  
[2,1,3]

Global	
misterio	proc
m	[2,1,3]
pc	<del>5</del> <del>6</del> <del>5</del> <del>6</del> 5

misterio	ls	[]
	pc	<del>0</del> 4
misterio	i	[3]
	m	[]
	ls	[3]
	pc	<del>0</del> <del>1</del> <del>2</del> 3
misterio	i	[2,3] [3,2]
	m	[3]
	ls	[2,3]
	pc	<del>0</del> <del>1</del> <del>2</del> 3
misterio	i	[1, 2, 3] [2, 1, 3] [2, 3, 1]
	m	[2,3]
	ls	[1,2,3]
	pc	<del>0</del> <del>1</del> <del>2</del> <del>3</del> <del>2</del> <del>3</del> 2

Código en Python

```
def misterio(ls):
0   if ls:
1       for m in misterio(ls[1:]):
2           for i in ins(ls[0], m):
3               yield i
4       else:
5           yield []
6   for m in misterio([1,2,3]):
7       print m
```

Imprime

[1,2,3]  
[2,1,3]

Global	
misterio	proc
m	[2,1,3]
pc	<del>5</del> <del>6</del> <del>5</del> <del>6</del> 5

misterio	ls	[]
	pc	<del>0</del> 4
misterio	i	[3]
	m	[]
	ls	[3]
	pc	<del>0</del> <del>1</del> <del>2</del> 3
misterio	i	[2,3] [3,2]
	m	[3]
	ls	[2,3]
	pc	<del>0</del> <del>1</del> <del>2</del> 3
misterio	i	[1, 2, 3] [2, 1, 3] [2, 3, 1]
	m	[2,3]
	ls	[1,2,3]
	pc	<del>0</del> <del>1</del> <del>2</del> <del>3</del> <del>2</del> <del>3</del> <del>2</del> 3



Código en Python

```
def misterio(ls):
0   if ls:
1       for m in misterio(ls[1:]):
2           for i in ins(ls[0], m):
3               yield i
4       else:
5           yield []
6   for m in misterio([1,2,3]):
    print m
```

Imprime

[1,2,3]  
[2,1,3]  
[2,3,1]

Global	
misterio	proc
m	[2,3,1]
pc	<del>5</del> <del>6</del> <del>5</del> <del>6</del> <del>5</del> <del>6</del>

misterio	ls	[]
	pc	<del>0</del> 4
misterio	i	[3]
	m	[]
	ls	[3]
	pc	<del>0</del> <del>1</del> <del>2</del> 3
misterio	i	[2,3] [3,2]
	m	[3]
	ls	[2,3]
	pc	<del>0</del> <del>1</del> <del>2</del> 3
misterio	i	[1, 2, 3] [2, 1, 3] [2, 3, 1]
	m	[2,3]
	ls	[1,2,3]
	pc	<del>0</del> <del>1</del> <del>2</del> <del>3</del> <del>2</del> <del>3</del> <del>2</del> <del>3</del>

# Código en Python

```
def misterio(ls):  
0   if ls:  
1       for m in misterio(ls[1:]):  
2           for i in ins(ls[0], m):  
3               yield i  
       else:  
4           yield []  
  
5 for m in misterio([1,2,3]):  
6     print m
```

# Imprime

[1,2,3]  
[2,1,3]  
[2,3,1]

Global	
misterio	proc
m	[2,3,1]
pc	<del>5</del> <del>6</del> <del>5</del> <del>6</del> <del>5</del> <del>6</del> <del>5</del>

misterio	ls	[]
	pc	<del>0</del> 4
misterio	i	[3]
	m	[]
	ls	[3]
	pc	<del>0</del> <del>1</del> <del>2</del> 3
misterio	i	[2,3] [3,2]
	m	[3]
	ls	[2,3]
	pc	<del>0</del> <del>1</del> <del>2</del> 3
misterio	i	[1, 2, 3] [2, 1, 3] [2, 3, 1]
	m	[2,3]
	ls	[1,2,3]
	pc	<del>0</del> <del>1</del> <del>2</del> <del>3</del> <del>2</del> <del>3</del> <del>2</del> 3

Código en Python

```
def misterio(ls):
0   if ls:
1       for m in misterio(ls[1:]):
2           for i in ins(ls[0], m):
3               yield i
4       else:
5           yield []
6   for m in misterio([1,2,3]):
7       print m
```

Imprime

[1,2,3]  
[2,1,3]  
[2,3,1]

Global	
misterio	proc
m	[2,3,1]
pc	<del>5</del> <del>6</del> <del>5</del> <del>6</del> <del>5</del> <del>6</del> <del>5</del>

misterio	ls	[]
	pc	<del>0</del> 4
misterio	i	[3]
	m	[]
	ls	[3]
	pc	<del>0</del> <del>1</del> <del>2</del> 3
misterio	i	[2,3] [3,2]
	m	[3]
	ls	[2,3]
	pc	<del>0</del> <del>1</del> <del>2</del> 3
misterio	i	-
	m	[2,3]
	ls	[1,2,3]
	pc	<del>0</del> <del>1</del> <del>2</del> <del>3</del> <del>2</del> <del>3</del> <del>2</del> <del>3</del> 2

Código en Python

```
def misterio(ls):
0   if ls:
1       for m in misterio(ls[1:]):
2           for i in ins(ls[0], m):
3               yield i
4       else:
5           yield []
6   for m in misterio([1,2,3]):
7       print m
```

Imprime

[1,2,3]  
[2,1,3]  
[2,3,1]

Global	
misterio	proc
m	[2,3,1]
pc	<del>5</del> <del>6</del> <del>5</del> <del>6</del> <del>5</del> <del>6</del> <del>5</del>

misterio	ls	[]
	pc	<del>0</del> 4
misterio	i	[3]
	m	[]
	ls	[3]
	pc	<del>0</del> <del>1</del> <del>2</del> 3
misterio	i	[2,3] [3,2]
	m	[3]
	ls	[2,3]
	pc	<del>0</del> <del>1</del> <del>2</del> 3
misterio	i	-
	m	-
	ls	[1,2,3]
	pc	<del>0</del> <del>1</del> <del>2</del> <del>3</del> <del>2</del> <del>3</del> <del>2</del> <del>3</del> <del>2</del> 1

Código en Python

```
def misterio(ls):
0   if ls:
1       for m in misterio(ls[1:]):
2           for i in ins(ls[0], m):
3               yield i
4       else:
5           yield []
6   for m in misterio([1,2,3]):
7       print m
```

Imprime

[1,2,3]  
[2,1,3]  
[2,3,1]

Global	
misterio	proc
m	[2,3,1]
pc	<del>5</del> <del>6</del> <del>5</del> <del>6</del> <del>5</del> <del>6</del> <del>5</del>

misterio	ls	[]
	pc	<del>0</del> 4
misterio	i	[3]
	m	[]
	ls	[3]
	pc	<del>0</del> <del>1</del> <del>2</del> 3
misterio	i	[2,3] [3,2]
	m	[3]
	ls	[2,3]
	pc	<del>0</del> <del>1</del> <del>2</del> <del>3</del> 2
misterio	i	-
	m	-
	ls	[1,2,3]
	pc	<del>0</del> <del>1</del> <del>2</del> <del>3</del> <del>2</del> <del>3</del> <del>2</del> <del>3</del> <del>2</del> 1

Código en Python

```
def misterio(ls):
0   if ls:
1       for m in misterio(ls[1:]):
2           for i in ins(ls[0], m):
3               yield i
4       else:
5           yield []
6   for m in misterio([1,2,3]):
7       print m
```

Imprime

[1,2,3]  
[2,1,3]  
[2,3,1]

Global	
misterio	proc
m	[2,3,1]
pc	<del>5</del> <del>6</del> <del>5</del> <del>6</del> <del>5</del> <del>6</del> <del>5</del>

misterio	ls	[]
	pc	<del>0</del> 4
misterio	i	[3]
	m	[]
	ls	[3]
	pc	<del>0</del> <del>1</del> <del>2</del> 3
misterio	i	[2,3] [3,2]
	m	[3]
	ls	[2,3]
	pc	<del>0</del> <del>1</del> <del>2</del> <del>3</del> <del>2</del> 3
misterio	i	-
	m	-
	ls	[1,2,3]
	pc	<del>0</del> <del>1</del> <del>2</del> <del>3</del> <del>2</del> <del>3</del> <del>2</del> <del>3</del> <del>2</del> 1

Código en Python

```
def misterio(ls):
0   if ls:
1       for m in misterio(ls[1:]):
2           for i in ins(ls[0], m):
3               yield i
4       else:
5           yield []
6   for m in misterio([1,2,3]):
7       print m
```

Imprime

[1,2,3]  
[2,1,3]  
[2,3,1]

Global	
misterio	proc
m	[2,3,1]
pc	<del>5</del> <del>6</del> <del>5</del> <del>6</del> <del>5</del> <del>6</del> <del>5</del>

misterio	ls	[]
	pc	<del>0</del> 4
misterio	i	[3]
	m	[]
	ls	[3]
	pc	<del>0</del> <del>1</del> <del>2</del> 3
misterio	i	[2,3] [3,2]
	m	[3]
	ls	[2,3]
	pc	<del>0</del> <del>1</del> <del>2</del> <del>3</del> <del>2</del> 3
misterio	i	[1,3,2][3,1,2][3,2,1]
	m	[3,2]
	ls	[1,2,3]
	pc	<del>0</del> <del>1</del> <del>2</del> <del>3</del> <del>2</del> <del>3</del> <del>2</del> <del>3</del> <del>2</del> <del>1</del> 2

Código en Python

```
def misterio(ls):
0   if ls:
1       for m in misterio(ls[1:]):
2           for i in ins(ls[0], m):
3               yield i
4       else:
5           yield []
6   for m in misterio([1,2,3]):
7       print m
```

Imprime

[1,2,3]  
[2,1,3]  
[2,3,1]

Global	
misterio	proc
m	[2,3,1]
pc	<del>5</del> <del>6</del> <del>5</del> <del>6</del> <del>5</del> <del>6</del> <del>5</del>

misterio	ls	[]
	pc	<del>0</del> 4
misterio	i	[3]
	m	[]
	ls	[3]
	pc	<del>0</del> <del>1</del> <del>2</del> 3
misterio	i	[2,3] [3,2]
	m	[3]
	ls	[2,3]
	pc	<del>0</del> <del>1</del> <del>2</del> <del>3</del> <del>2</del> 3
misterio	i	[1,3,2][3,1,2][3,2,1]
	m	[3,2]
	ls	[1,2,3]
	pc	<del>0</del> <del>1</del> <del>2</del> <del>3</del> <del>2</del> <del>3</del> <del>2</del> <del>3</del> <del>2</del> <del>1</del> <del>2</del> 3



Código en Python

```
def misterio(ls):
0   if ls:
1       for m in misterio(ls[1:]):
2           for i in ins(ls[0], m):
3               yield i
4       else:
5           yield []
6   for m in misterio([1,2,3]):
7       print m
```

Imprime

[1,2,3]  
[2,1,3]  
[2,3,1]  
[1,3,2]

Global	
misterio	proc
m	[1,3,2]
pc	<del>5</del> <del>6</del> <del>5</del> <del>6</del> <del>5</del> <del>6</del> <del>5</del> <del>6</del>

misterio	ls	[]
	pc	<del>0</del> 4
misterio	i	[3]
	m	[]
	ls	[3]
	pc	<del>0</del> <del>1</del> <del>2</del> 3
misterio	i	[2,3] [3,2]
	m	[3]
	ls	[2,3]
	pc	<del>0</del> <del>1</del> <del>2</del> <del>3</del> <del>2</del> 3
misterio	i	[1,3,2][3,1,2][3,2,1]
	m	[3,2]
	ls	[1,2,3]
	pc	<del>0</del> <del>1</del> <del>2</del> <del>3</del> <del>2</del> <del>3</del> <del>2</del> <del>3</del> <del>2</del> <del>1</del> <del>2</del> 3

Código en Python

```
def misterio(ls):
0   if ls:
1       for m in misterio(ls[1:]):
2           for i in ins(ls[0], m):
3               yield i
4       else:
5           yield []
6   for m in misterio([1,2,3]):
7       print m
```

Imprime

[1,2,3]  
[2,1,3]  
[2,3,1]  
[1,3,2]

Global	
misterio	proc
m	[1,3,2]
pc	5

misterio	ls	[]
	pc	<del>0</del> 4
misterio	i	[3]
	m	[]
	ls	[3]
	pc	<del>0</del> <del>1</del> <del>2</del> 3
misterio	i	[2,3] [3,2]
	m	[3]
	ls	[2,3]
	pc	<del>0</del> <del>1</del> <del>2</del> <del>3</del> <del>2</del> 3
misterio	i	[1,3,2][3,1,2][3,2,1]
	m	[3,2]
	ls	[1,2,3]
	pc	<del>0</del> <del>1</del> <del>2</del> <del>3</del> <del>2</del> <del>3</del> <del>2</del> <del>3</del> <del>2</del> <del>1</del> <del>2</del> 3

Código en Python

```
def misterio(ls):
0   if ls:
1       for m in misterio(ls[1:]):
2           for i in ins(ls[0], m):
3               yield i
4       else:
5           yield []
6   for m in misterio([1,2,3]):
7       print m
```

Imprime

[1,2,3]  
[2,1,3]  
[2,3,1]  
[1,3,2]

Global	
misterio	proc
m	[1,3,2]
pc	5

misterio	ls	[]
	pc	<del>0</del> 4
misterio	i	[3]
	m	[]
	ls	[3]
	pc	<del>0</del> <del>1</del> <del>2</del> 3
misterio	i	[2,3] [3,2]
	m	[3]
	ls	[2,3]
	pc	<del>0</del> <del>1</del> <del>2</del> <del>3</del> 2 3
misterio	i	[1,3,2][3,1,2][3,2,1]
	m	[3,2]
	ls	[1,2,3]
	pc	2

Código en Python

```
def misterio(ls):
0   if ls:
1       for m in misterio(ls[1:]):
2           for i in ins(ls[0], m):
3               yield i
4       else:
5           yield []
6   for m in misterio([1,2,3]):
7       print m
```

Imprime

[1,2,3]  
[2,1,3]  
[2,3,1]  
[1,3,2]

Global	
misterio	proc
m	[1,3,2]
pc	5

misterio	ls	[]
	pc	<del>0</del> 4
misterio	i	[3]
	m	[]
	ls	[3]
	pc	<del>0</del> <del>1</del> <del>2</del> 3
misterio	i	[2,3] [3,2]
	m	[3]
	ls	[2,3]
	pc	<del>0</del> <del>1</del> <del>2</del> <del>3</del> <del>2</del> 3
misterio	i	[1,3,2][3,1,2][3,2,1]
	m	[3,2]
	ls	[1,2,3]
	pc	<del>2</del> 3

Código en Python

```
def misterio(ls):
0   if ls:
1       for m in misterio(ls[1:]):
2           for i in ins(ls[0], m):
3               yield i
4       else:
5           yield []
6   for m in misterio([1,2,3]):
    print m
```

Imprime

[1,2,3]  
[2,1,3]  
[2,3,1]  
[1,3,2]  
[3,1,2]

Global	
misterio	proc
m	[3,1,2]
pc	<del>5</del> 6

misterio	ls	[]
	pc	<del>0</del> 4
misterio	i	[3]
	m	[]
	ls	[3]
	pc	<del>0</del> <del>1</del> <del>2</del> 3
misterio	i	[2,3] [3,2]
	m	[3]
	ls	[2,3]
	pc	<del>0</del> <del>1</del> <del>2</del> <del>3</del> <del>2</del> 3
misterio	i	[1,3,2][3,1,2][3,2,1]
	m	[3,2]
	ls	[1,2,3]
	pc	<del>2</del> 3

Código en Python

```
def misterio(ls):
0   if ls:
1       for m in misterio(ls[1:]):
2           for i in ins(ls[0], m):
3               yield i
4       else:
5           yield []
6   for m in misterio([1,2,3]):
7       print m
```

Imprime

[1,2,3]  
[2,1,3]  
[2,3,1]  
[1,3,2]  
[3,1,2]

Global	
misterio	proc
m	[3,1,2]
pc	5

misterio	ls	[]
	pc	<del>0</del> 4
misterio	i	[3]
	m	[]
	ls	[3]
	pc	<del>0</del> 1 <del>2</del> 3
misterio	i	[2,3] [3,2]
	m	[3]
	ls	[2,3]
	pc	<del>0</del> 1 <del>2</del> 3 <del>2</del> 3
misterio	i	[1,3,2][3,1,2][3,2,1]
	m	[3,2]
	ls	[1,2,3]
	pc	<del>2</del> 3

Código en Python

```
def misterio(ls):
0   if ls:
1       for m in misterio(ls[1:]):
2           for i in ins(ls[0], m):
3               yield i
4       else:
5           yield []
6   for m in misterio([1,2,3]):
7       print m
```

Imprime

[1,2,3]  
[2,1,3]  
[2,3,1]  
[1,3,2]  
[3,1,2]

Global	
misterio	proc
m	[3,1,2]
pc	5

misterio	ls	[]
	pc	<del>0</del> 4
misterio	i	[3]
	m	[]
	ls	[3]
	pc	<del>0</del> <del>1</del> <del>2</del> 3
misterio	i	[2,3] [3,2]
	m	[3]
	ls	[2,3]
	pc	<del>0</del> <del>1</del> <del>2</del> <del>3</del> <del>2</del> 3
misterio	i	[1,3,2][3,1,2][3,2,1]
	m	[3,2]
	ls	[1,2,3]
	pc	<del>2</del> <del>3</del> 2

Código en Python

```
def misterio(ls):
0   if ls:
1       for m in misterio(ls[1:]):
2           for i in ins(ls[0], m):
3               yield i
4       else:
5           yield []
6   for m in misterio([1,2,3]):
7       print m
```

Imprime

[1,2,3]  
[2,1,3]  
[2,3,1]  
[1,3,2]  
[3,1,2]

Global	
misterio	proc
m	[3,1,2]
pc	5

misterio	ls	[]
	pc	<del>0</del> 4
misterio	i	[3]
	m	[]
	ls	[3]
	pc	<del>0</del> <del>1</del> <del>2</del> 3
misterio	i	[2,3] [3,2]
	m	[3]
	ls	[2,3]
	pc	<del>0</del> <del>1</del> <del>2</del> <del>3</del> <del>2</del> 3
misterio	i	[1,3,2][3,1,2][3,2,1]
	m	[3,2]
	ls	[1,2,3]
	pc	3



Código en Python

```
def misterio(ls):
0   if ls:
1       for m in misterio(ls[1:]):
2           for i in ins(ls[0], m):
3               yield i
4       else:
5           yield []
6   for m in misterio([1,2,3]):
7       print m
```

Imprime

[1,2,3]  
[2,1,3]  
[2,3,1]  
[1,3,2]  
[3,1,2]  
[3,2,1]

Global	
misterio	proc
m	[3,2,1]
pc	6

misterio	ls	[]
	pc	<del>0</del> 4
misterio	i	[3]
	m	[]
	ls	[3]
	pc	<del>0</del> 1 <del>2</del> 3
misterio	i	[2,3] [3,2]
	m	[3]
	ls	[2,3]
	pc	<del>0</del> 1 <del>2</del> 3 <del>2</del> 3
misterio	i	[1,3,2][3,1,2][3,2,1]
	m	[3,2]
	ls	[1,2,3]
	pc	3

Código en Python

```
def misterio(ls):
0   if ls:
1       for m in misterio(ls[1:]):
2           for i in ins(ls[0], m):
3               yield i
4       else:
5           yield []
6   for m in misterio([1,2,3]):
7       print m
```

Imprime

[1,2,3]  
[2,1,3]  
[2,3,1]  
[1,3,2]  
[3,1,2]  
[3,2,1]

Global	
misterio	proc
m	[3,2,1]
pc	5

misterio	ls	[]
	pc	<del>0</del> 4
misterio	i	[3]
	m	[]
	ls	[3]
	pc	<del>0</del> <del>1</del> <del>2</del> 3
misterio	i	[2,3] [3,2]
	m	[3]
	ls	[2,3]
	pc	<del>0</del> <del>1</del> <del>2</del> <del>3</del> <del>2</del> 3
misterio	i	[1,3,2][3,1,2][3,2,1]
	m	[3,2]
	ls	[1,2,3]
	pc	3

# Código en Python

```
def misterio(ls):  
0   if ls:  
1       for m in misterio(ls[1:]):  
2           for i in ins(ls[0], m):  
3               yield i  
       else:  
4           yield []  
  
5 for m in misterio([1,2,3]):  
6     print m
```

# Imprime

[1,2,3]  
[2,1,3]  
[2,3,1]  
[1,3,2]  
[3,1,2]  
[3,2,1]

Global	
misterio	proc
m	[3,2,1]
pc	5

misterio	ls	[]
	pc	<del>0</del> 4
misterio	i	[3]
	m	[]
	ls	[3]
	pc	<del>0</del> <del>1</del> <del>2</del> 3
misterio	i	[2,3] [3,2]
	m	[3]
	ls	[2,3]
	pc	<del>0</del> <del>1</del> <del>2</del> <del>3</del> <del>2</del> 3
misterio	i	-
	m	[3,2]
	ls	[1,2,3]
	pc	<del>3</del> 2

Código en Python

```
def misterio(ls):
0   if ls:
1       for m in misterio(ls[1:]):
2           for i in ins(ls[0], m):
3               yield i
4       else:
5           yield []
6   for m in misterio([1,2,3]):
7       print m
```

Imprime

[1,2,3]  
[2,1,3]  
[2,3,1]  
[1,3,2]  
[3,1,2]  
[3,2,1]

Global	
misterio	proc
m	[3,2,1]
pc	5

misterio	ls	[]
	pc	<del>0</del> 4
misterio	i	[3]
	m	[]
	ls	[3]
	pc	<del>0</del> <del>1</del> <del>2</del> 3
misterio	i	[2,3] [3,2]
	m	[3]
	ls	[2,3]
	pc	<del>0</del> <del>1</del> <del>2</del> <del>3</del> <del>2</del> 3
misterio	i	-
	m	-
	ls	[1,2,3]
	pc	<del>3</del> <del>2</del> 1

# Código en Python

```
def misterio(ls):
0   if ls:
1       for m in misterio(ls[1:]):
2           for i in ins(ls[0], m):
3               yield i
4       else:
5           yield []
6   for m in misterio([1,2,3]):
7       print m
```

# Imprime

[1,2,3]  
[2,1,3]  
[2,3,1]  
[1,3,2]  
[3,1,2]  
[3,2,1]

Global	
misterio	proc
m	[3,2,1]
pc	5

misterio	ls	[]
	pc	<del>0</del> 4
misterio	i	[3]
	m	[]
	ls	[3]
	pc	<del>0</del> <del>1</del> <del>2</del> 3
misterio	i	-
	m	[3]
	ls	[2,3]
	pc	<del>0</del> <del>1</del> <del>2</del> <del>3</del> <del>2</del> <del>3</del> 2
misterio	i	-
	m	-
	ls	[1,2,3]
	pc	<del>3</del> <del>2</del> 1

# Código en Python

```
def misterio(ls):
0   if ls:
1       for m in misterio(ls[1:]):
2           for i in ins(ls[0], m):
3               yield i
4       else:
5           yield []
6   for m in misterio([1,2,3]):
7       print m
```

# Imprime

[1,2,3]  
[2,1,3]  
[2,3,1]  
[1,3,2]  
[3,1,2]  
[3,2,1]

Global	
misterio	proc
m	[3,2,1]
pc	5

misterio	ls	[]
	pc	<del>0</del> 4
misterio	i	[3]
	m	[]
	ls	[3]
	pc	<del>0</del> <del>1</del> <del>2</del> 3
misterio	i	-
	m	-
	ls	[2,3]
	pc	<del>0</del> <del>1</del> <del>2</del> <del>3</del> <del>2</del> <del>3</del> <del>2</del> 1
misterio	i	-
	m	-
	ls	[1,2,3]
	pc	<del>3</del> <del>2</del> 1

Código en Python

```
def misterio(ls):
0   if ls:
1       for m in misterio(ls[1:]):
2           for i in ins(ls[0], m):
3               yield i
4       else:
5           yield []
6   for m in misterio([1,2,3]):
7       print m
```

Imprime

[1,2,3]  
[2,1,3]  
[2,3,1]  
[1,3,2]  
[3,1,2]  
[3,2,1]

Global	
misterio	proc
m	[3,2,1]
pc	5

misterio	ls	[]
	pc	<del>0</del> 4
misterio	i	-
	m	[]
	ls	[3]
	pc	<del>0</del> <del>1</del> <del>2</del> <del>3</del> 2
misterio	i	-
	m	-
	ls	[2,3]
	pc	<del>0</del> <del>1</del> <del>2</del> <del>3</del> <del>2</del> <del>3</del> <del>2</del> 1
misterio	i	-
	m	-
	ls	[1,2,3]
	pc	<del>3</del> <del>2</del> 1

Código en Python

```
def misterio(ls):
0   if ls:
1       for m in misterio(ls[1:]):
2           for i in ins(ls[0], m):
3               yield i
4       else:
5           yield []
6   for m in misterio([1,2,3]):
7       print m
```

Imprime

[1,2,3]  
[2,1,3]  
[2,3,1]  
[1,3,2]  
[3,1,2]  
[3,2,1]

Global	
misterio	proc
m	[3,2,1]
pc	5

misterio	ls	[]
	pc	<del>0</del> 4
misterio	i	-
	m	-
	ls	[3]
	pc	<del>0</del> <del>1</del> <del>2</del> <del>3</del> <del>2</del> 1
misterio	i	-
	m	-
	ls	[2,3]
	pc	<del>0</del> <del>1</del> <del>2</del> <del>3</del> <del>2</del> <del>3</del> <del>2</del> 1
misterio	i	-
	m	-
	ls	[1,2,3]
	pc	<del>3</del> <del>2</del> 1



Código en Python

```
def misterio(ls):
0   if ls:
1       for m in misterio(ls[1:]):
2           for i in ins(ls[0], m):
3               yield i
4       else:
5           yield []

5   for m in misterio([1,2,3]):
6       print m
```

Imprime

[1,2,3]  
[2,1,3]  
[2,3,1]  
[1,3,2]  
[3,1,2]  
[3,2,1]

Global	
misterio	proc
m	[3,2,1]
pc	5

misterio	ls	[]
	pc	<del>0</del> 4
misterio	i	-
	m	-
	ls	[3]
	pc	<del>0</del> <del>1</del> <del>2</del> <del>3</del> <del>2</del> <del>1</del>
misterio	i	-
	m	-
	ls	[2,3]
	pc	<del>0</del> <del>1</del> <del>2</del> <del>3</del> <del>2</del> <del>3</del> <del>2</del> <del>1</del>
misterio	i	-
	m	-
	ls	[1,2,3]
	pc	<del>3</del> <del>2</del> <del>1</del>

Código en Python

```
def misterio(ls):
0   if ls:
1       for m in misterio(ls[1:]):
2           for i in ins(ls[0], m):
3               yield i
4       else:
5           yield []
6   for m in misterio([1,2,3]):
7       print m
```

Imprime

[1,2,3]  
[2,1,3]  
[2,3,1]  
[1,3,2]  
[3,1,2]  
[3,2,1]

Global	
misterio	proc
m	[3,2,1]
pc	5

misterio	ls	[]
	pc	<del>0</del> 4
misterio	i	-
	m	-
	ls	[3]
	pc	<del>0</del> <del>1</del> <del>2</del> <del>3</del> <del>2</del> <del>1</del>
misterio	i	-
	m	-
	ls	[2,3]
	pc	<del>0</del> <del>1</del> <del>2</del> <del>3</del> <del>2</del> <del>3</del> <del>2</del> <del>1</del>
misterio	i	-
	m	-
	ls	[1,2,3]
	pc	<del>3</del> <del>2</del> <del>1</del>

Código en Python

```
def misterio(ls):
0   if ls:
1       for m in misterio(ls[1:]):
2           for i in ins(ls[0], m):
3               yield i
4       else:
5           yield []
6   for m in misterio([1,2,3]):
7       print m
```

Imprime

[1,2,3]  
[2,1,3]  
[2,3,1]  
[1,3,2]  
[3,1,2]  
[3,2,1]

Global	
misterio	proc
m	[3,2,1]
pc	5

misterio	ls	[]
	pc	<del>0</del> 4
misterio	i	-
	m	-
	ls	[3]
	pc	<del>0</del> <del>1</del> <del>2</del> <del>3</del> <del>2</del> 1
misterio	i	-
	m	-
	ls	[2,3]
	pc	<del>0</del> <del>1</del> <del>2</del> <del>3</del> <del>2</del> <del>3</del> <del>2</del> 1
misterio	i	-
	m	-
	ls	[1,2,3]
	pc	<del>3</del> <del>2</del> 1

## Código en Python

## Imprime

```
def misterio(ls):  
0   if ls:  
1       for m in misterio(ls[1:]):  
2           for i in ins(ls[0], m):  
3               yield i  
       else:  
4           yield []  
  
5 for m in misterio([1,2,3]):  
6     print m
```

```
[1,2,3]  
[2,1,3]  
[2,3,1]  
[1,3,2]  
[3,1,2]  
[3,2,1]
```

# Programa Finalizado

*“Mucho que aprender  
todavía tienes”.*

- Yoda-

# El Misterio Develado

La función Misterio es un iterador que calcula las permutaciones posibles de una lista de  $n$  elementos pasada como argumento.

Para ello, la función obtiene todas las permutaciones posibles de los últimos  $n-1$  elementos mediante una llamada recursiva. Luego apoyándose en la función `ins` que retorna cada una de las posibles inserciones de un elemento en una lista, retorna cada una de las formas posibles de insertar la cabeza de la lista recibida en cada una de las  $(n-1)!$  permutaciones obtenidas del caso recursivo.