

EJERCICIO 2

BY LUIS PARRA

INICIAMOS EL LA CORRIDA EN ESTE ESTADO

Imprime:

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

GLOBAL

| | |
|-----|------|
| ins | proc |
| i | - |
| pc | 7 |

Entramos por primera vez a ins y ejecutamos el yield de la linea 2

Imprime:

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

GLOBAL

| | |
|-----|------|
| ins | proc |
| i | - |
| pc | 7 |

ins

| | | |
|---|----|---------|
| 2 | ls | [1,2,3] |
| 1 | e | 0 |
| 0 | pc | 2 |

El yield retorna [e, *ls] que es [0,1,2,3]. Y lo almacenamos en i global

Imprime:

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

GLOBAL

| | |
|-----|-----------|
| ins | proc |
| i | [0,1,2,3] |
| pc | 7 |

ins

| | | |
|---|----|---------|
| 2 | ls | [1,2,3] |
| 1 | e | 0 |
| 0 | pc | 2 |

Imprimimos i

Imprime: [0,1,2,3]

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

GLOBAL

| ins | proc |
|-----|-----------|
| i | [0,1,2,3] |
| pc | ≠ 8 |

ins

| | | |
|---|----|---------|
| 2 | ls | [1,2,3] |
| 1 | e | 0 |
| 0 | pc | 2 |

Chequeamos el if de la linea 3, se cumple.

Imprime: [0,1,2,3]

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

GLOBAL

| | |
|-----|-------------|
| ins | proc |
| i | [0,1,2,3] - |
| pc | 7 8 7 |

ins

| | | |
|---|----|---------|
| 2 | ls | [1,2,3] |
| 1 | e | 0 |
| 0 | pc | 2 3 |

Empezamos el for de la linea 4. Agregamos una nueva variable i

Imprime: [0,1,2,3]

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

GLOBAL

| | |
|-----|-------------|
| ins | proc |
| i | [0,1,2,3] - |
| pc | 7 8 7 |

ins

| | | |
|---|----|---------|
| 3 | i | - |
| 2 | ls | [1,2,3] |
| 1 | e | 0 |
| 0 | pc | 2 3 4 |

Se entra al nuevo ins y se ejecuta el yield de la linea 2

Imprime: [0,1,2,3]

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

GLOBAL

| | |
|-----|-------------|
| ins | proc |
| i | [0,1,2,3] - |
| pc | 7 8 7 |

ins

| | ls | [2,3] |
|---|----|-------|
| 6 | e | 0 |
| 5 | pc | 2 |

ins

| | | |
|---|----|---------|
| 3 | i | - |
| 2 | ls | [1,2,3] |
| 1 | e | 0 |
| 0 | pc | 2 3 4 |

El resultado se almacena en la i3

Imprime: [0,1,2,3]

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

GLOBAL

| | |
|-----|-------------|
| ins | proc |
| i | [0,1,2,3] - |
| pc | 7 8 7 |

ins

| 6 | ls | [2,3] |
|---|----|-------|
| 5 | e | 0 |
| 4 | pc | 2 |

ins

| 3 | i | [0,2,3] |
|---|----|---------|
| 2 | ls | [1,2,3] |
| 1 | e | 0 |
| 0 | pc | 2 3 4 |

Se ejecuta el yield de la linea 5 y el valor [1,0,2,3] se guarda en i global

Imprime: [0,1,2,3]

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

| ins | | |
|-----|----|-------|
| 6 | ls | [2,3] |
| 5 | e | 0 |
| 4 | pc | 2 |

| GLOBAL | | |
|--------|---------------------|--|
| ins | proc | |
| i | [0,1,2,3] [1,0,2,3] | |
| pc | 7 8 7 | |

| ins | | |
|-----|----|---------|
| 3 | i | [0,2,3] |
| 2 | ls | [1,2,3] |
| 1 | e | 0 |
| 0 | pc | 2 3 4 5 |

Se ejecuta la linea 8 y se imprime i global

Imprime: [0,1,2,3], [1,0,2,3]

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

GLOBAL

| | |
|-----|---------------------|
| ins | proc |
| i | [0,1,2,3] [1,0,2,3] |
| pc | 7 8 7 8 |

ins

| | | ins |
|---|----|-------|
| 6 | ls | [2,3] |
| 5 | e | 0 |
| 4 | pc | 2 |

ins

| | | ins |
|---|----|---------|
| 3 | i | [0,2,3] |
| 2 | ls | [1,2,3] |
| 1 | e | 0 |
| 0 | pc | 2 3 4 5 |

Se entra a ins de la linea 7 nuevamente

Imprime: [0,1,2,3], [1,0,2,3]

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

GLOBAL

| | |
|-----|----------------------------------|
| ins | proc |
| i | [0,1,2,3] [1,0,2,3] - |
| pc | 7 8 7 8 7 |

ins

| ins | | |
|-----|----|-------|
| 6 | ls | [2,3] |
| 5 | e | 0 |
| 4 | pc | 2 |

ins

| ins | | |
|-----|----|---------|
| 3 | i | [0,2,3] |
| 2 | ls | [1,2,3] |
| 1 | e | 0 |
| 0 | pc | 2 3 4 5 |

Se entra al ins de la linea 4

Imprime: [0,1,2,3], [1,0,2,3]

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

GLOBAL

| | |
|-----|-----------------------|
| ins | proc |
| i | [0,1,2,3] [1,0,2,3] - |
| pc | 7 8 7 8 7 |

ins

| ins | | |
|-----|----|-------|
| 6 | ls | [2,3] |
| 5 | e | 0 |
| 4 | pc | 2 |

ins

| ins | | |
|-----|----|-----------|
| 3 | i | [0,2,3] - |
| 2 | ls | [1,2,3] |
| 1 | e | 0 |
| 0 | pc | 2 3 4 5 4 |

Se verifica el If de la linea 3, se cumple.

Imprime: [0,1,2,3], [1,0,2,3]

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

| ins | | |
|-----|----|-------|
| 6 | ls | [2,3] |
| 5 | e | 0 |
| 4 | pc | 2 3 |

| GLOBAL | |
|--------|-----------------------|
| ins | proc |
| i | [0,1,2,3] [1,0,2,3] - |
| pc | 7 8 7 8 7 |

| ins | | |
|-----|----|-----------|
| 3 | i | [0,2,3] - |
| 2 | ls | [1,2,3] |
| 1 | e | 0 |
| 0 | pc | 2 3 4 5 4 |

Se comienza el for de la linea 4, se agrega una nueva variable i

Imprime: [0,1,2,3], [1,0,2,3]

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

| ins | | |
|-----|----|-------|
| 7 | i | - |
| 6 | ls | [2,3] |
| 5 | e | 0 |
| 4 | pc | 2 3 4 |

| GLOBAL | | |
|--------|-----------------------|--|
| ins | proc | |
| i | [0,1,2,3] [1,0,2,3] - | |
| pc | 7 8 7 8 7 | |

| ins | | |
|-----|----|-----------|
| 3 | i | [0,2,3] - |
| 2 | ls | [1,2,3] |
| 1 | e | 0 |
| 0 | pc | 2 3 4 5 4 |

Se entra a ins y se ejecuta el yield de la linea 2

Imprime: [0,1,2,3], [1,0,2,3]

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

| GLOBAL | | |
|--------|-----------------------|--|
| ins | proc | |
| i | [0,1,2,3] [1,0,2,3] - | |
| pc | 7 8 7 8 7 | |

| ins | | |
|-----|----|-----|
| 10 | ls | [3] |
| 9 | e | 0 |
| 8 | pc | 2 |

| ins | | |
|-----|----|-------|
| 7 | i | - |
| 6 | ls | [2,3] |
| 5 | e | 0 |
| 4 | pc | 2 3 4 |

| ins | | |
|-----|----|-----------|
| 3 | i | [0,2,3] - |
| 2 | ls | [1,2,3] |
| 1 | e | 0 |
| 0 | pc | 2 3 4 5 4 |

Se almacena el yield en i7 con valor [0,3]

Imprime: [0,1,2,3], [1,0,2,3]

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

| GLOBAL | | |
|--------|----------------------------------|--|
| ins | proc | |
| i | [0,1,2,3] [1,0,2,3] - | |
| pc | 7 8 7 8 7 | |

| ins | | |
|-----|----|-----|
| 10 | ls | [3] |
| 9 | e | 0 |
| 8 | pc | 2 |

| ins | | |
|-----|----|-------|
| 7 | i | [0,3] |
| 6 | ls | [2,3] |
| 5 | e | 0 |
| 4 | pc | 2 3 4 |

| ins | | |
|-----|----|-----------|
| 3 | i | [0,2,3] - |
| 2 | ls | [1,2,3] |
| 1 | e | 0 |
| 0 | pc | 2 3 4 5 4 |

Se ejecuta el yield de la linea 5 y se almcena el resultado en i3

Imprime: [0,1,2,3], [1,0,2,3]

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

| GLOBAL | | |
|--------|-----------------------|--|
| ins | proc | |
| i | [0,1,2,3] [1,0,2,3] - | |
| pc | 7 8 7 8 7 | |

| ins | | |
|-----|----|-----|
| 10 | ls | [3] |
| 9 | e | 0 |
| 8 | pc | 2 |

| ins | | |
|-----|----|---------|
| 7 | i | [0,3] |
| 6 | ls | [2,3] |
| 5 | e | 0 |
| 4 | pc | 2 3 4 5 |

| ins | | |
|-----|----|-----------------|
| 3 | i | [0,2,3] [2,0,3] |
| 2 | ls | [1,2,3] |
| 1 | e | 0 |
| 0 | pc | 2 3 4 5 4 |

Se ejecuta el yield de la linea 5 nuevamente y se almacena el resultado en i global

Imprime: [0,1,2,3], [1,0,2,3]

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

| ins | | |
|-----|----|-----|
| 10 | ls | [3] |
| 9 | e | 0 |
| 8 | pc | 2 |

| ins | | |
|-----|----|---------|
| 7 | i | [0,3] |
| 6 | ls | [2,3] |
| 5 | e | 0 |
| 4 | pc | 2 3 4 5 |

| GLOBAL | |
|--------|---------------------|
| ins | proc |
| i | [1,0,2,3] [1,2,0,3] |
| pc | 7 8 7 8 7 |

| ins | | |
|-----|----|-----------------|
| 3 | i | [0,2,3] [2,0,3] |
| 2 | ls | [1,2,3] |
| 1 | e | 0 |
| 0 | pc | 2 3 4 5 4 5 |

Se imprime el valor de i global

Imprime: [0,1,2,3], [1,0,2,3], [1,2,0,3]

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

| ins | | |
|-----|----|-----|
| 10 | ls | [3] |
| 9 | e | 0 |
| 8 | pc | 2 |

| ins | | |
|-----|----|---------|
| 7 | i | [0,3] |
| 6 | ls | [2,3] |
| 5 | e | 0 |
| 4 | pc | 2 3 4 5 |

| GLOBAL | | |
|--------|---------------------|--|
| ins | proc | |
| i | [1,0,2,3] [1,2,0,3] | |
| pc | 7 8 7 8 7 8 | |

| ins | | |
|-----|----|-----------------|
| 3 | i | [0,2,3] [2,0,3] |
| 2 | ls | [1,2,3] |
| 1 | e | 0 |
| 0 | pc | 2 3 4 5 4 5 |

Se entra en el ins de la linea 7 y se resume ejecucion

Imprime: [0,1,2,3], [1,0,2,3], [1,2,0,3]

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

| ins | | |
|-----|----|-----|
| 10 | ls | [3] |
| 9 | e | 0 |
| 8 | pc | 2 |

| ins | | |
|-----|----|---------|
| 7 | i | [0,3] |
| 6 | ls | [2,3] |
| 5 | e | 0 |
| 4 | pc | 2 3 4 5 |

GLOBAL

| ins | proc |
|-----|---------------|
| i | [1,2,0,3] - |
| pc | 7 8 7 8 7 8 7 |

| ins | | |
|-----|----|-----------------|
| 3 | i | [0,2,3] [2,0,3] |
| 2 | ls | [1,2,3] |
| 1 | e | 0 |
| 0 | pc | 2 3 4 5 4 5 |

Se entra en el ins de la linea 4 y se resume ejecucion

Imprime: [0,1,2,3], [1,0,2,3], [1,2,0,3]

```

1  def ins(e, ls):
2      yield [e, *ls]
3      if ls:
4          for i in ins(e, ls[1:]):
5              yield [ls[0], *i]
6
7  for i in ins(0, [1, 2, 3]):
8      print i

```

| ins | | |
|-----|----|-----|
| 10 | ls | [3] |
| 9 | e | 0 |
| 8 | pc | 2 |

| ins | | |
|-----|----|---------|
| 7 | i | [0,3] |
| 6 | ls | [2,3] |
| 5 | e | 0 |
| 4 | pc | 2 3 4 5 |

GLOBAL

| ins | proc |
|-----|---------------|
| i | [1,2,0,3] - |
| pc | 7 8 7 8 7 8 7 |

| ins | | |
|-----|----|---------------|
| 3 | i | [2,0,3] - |
| 2 | ls | [1,2,3] |
| 1 | e | 0 |
| 0 | pc | 2 3 4 5 4 5 4 |

Se entra en el ins de la linea 4 (Again) y se resume ejecucion

Imprime: [0,1,2,3], [1,0,2,3], [1,2,0,3]

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

| ins | | |
|-----|----|-----|
| 10 | ls | [3] |
| 9 | e | 0 |
| 8 | pc | 2 |

| ins | | |
|-----|----|-----------|
| 7 | i | [0,3] - |
| 6 | ls | [2,3] |
| 5 | e | 0 |
| 4 | pc | 2 3 4 5 4 |

| GLOBAL | | |
|--------|---------------|--|
| ins | proc | |
| i | [1,2,0,3] - | |
| pc | 7 8 7 8 7 8 7 | |

| ins | | |
|-----|----|---------------|
| 3 | i | [2,0,3] - |
| 2 | ls | [1,2,3] |
| 1 | e | 0 |
| 0 | pc | 2 3 4 5 4 5 4 |

Se chequea el if de la linea 3, se cumple. Y se agrega una nueva variable i

Imprime: [0,1,2,3], [1,0,2,3], [1,2,0,3]

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

| ins | | |
|-----|----|-----|
| 11 | i | - |
| 10 | ls | [3] |
| 9 | e | 0 |
| 8 | pc | 2 3 |

| ins | | |
|-----|----|-----------|
| 7 | i | [0,3] - |
| 6 | ls | [2,3] |
| 5 | e | 0 |
| 4 | pc | 2 3 4 5 4 |

| GLOBAL | | |
|--------|---------------|--|
| ins | proc | |
| i | [1,2,0,3] - | |
| pc | 7 8 7 8 7 8 7 | |

| ins | | |
|-----|----|---------------|
| 3 | i | [2,0,3] - |
| 2 | ls | [1,2,3] |
| 1 | e | 0 |
| 0 | pc | 2 3 4 5 4 5 4 |

Se entra en el ins de la linea 4 y este retorna ejecuta el yield de la linea 2

Imprime: [0,1,2,3], [1,0,2,3], [1,2,0,3]

```

1 def ins(e, ls):
2     yield [e, *ls]
3     if ls:
4         for i in ins(e, ls[1:]):
5             yield [ls[0], *i]
6
7 for i in ins(0, [1, 2, 3]):
8     print i

```

GLOBAL

| | ins | proc |
|----|---------------|------|
| i | [1,2,0,3] - | |
| pc | 7 8 7 8 7 8 7 | |

| ins | | |
|-----|----|-------|
| 11 | i | - |
| 10 | ls | [3] |
| 9 | e | 0 |
| 8 | pc | 2 3 4 |

| ins | | |
|-----|----|----|
| 14 | ls | [] |
| 13 | e | 0 |
| 12 | pc | 2 |

| ins | | |
|-----|----|-----------|
| 7 | i | [0,3] - |
| 6 | ls | [2,3] |
| 5 | e | 0 |
| 4 | pc | 2 3 4 5 4 |

| ins | | |
|-----|----|---------------|
| 3 | i | [2,0,3] - |
| 2 | ls | [1,2,3] |
| 1 | e | 0 |
| 0 | pc | 2 3 4 5 4 5 4 |

El resultado del yield se guarda en i11

Imprime: [0,1,2,3], [1,0,2,3], [1,2,0,3]

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

GLOBAL

| ins | proc |
|-----|---------------|
| i | [1,2,0,3] - |
| pc | 7 8 7 8 7 8 7 |

| ins | | |
|-----|----|-------|
| 11 | i | [0] |
| 10 | ls | [3] |
| 9 | e | 0 |
| 8 | pc | 2 3 4 |

| ins | | |
|-----|----|----|
| 14 | ls | [] |
| 13 | e | 0 |
| 12 | pc | 2 |

| ins | | |
|-----|----|-----------|
| 7 | i | [0,3] - |
| 6 | ls | [2,3] |
| 5 | e | 0 |
| 4 | pc | 2 3 4 5 4 |

| ins | | |
|-----|----|---------------|
| 3 | i | [2,0,3] - |
| 2 | ls | [1,2,3] |
| 1 | e | 0 |
| 0 | pc | 2 3 4 5 4 5 4 |

Se procede a realizar el yield de la linea 5 y se guarda el resultado en i7

Imprime: [0,1,2,3], [1,0,2,3], [1,2,0,3]

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

GLOBAL

| | |
|-----|---------------|
| ins | proc |
| i | [1,2,0,3] - |
| pc | 7 8 7 8 7 8 7 |

| ins | | |
|-----|----|---------|
| 11 | i | [0] |
| 10 | ls | [3] |
| 9 | e | 0 |
| 8 | pc | 2 3 4 5 |

| ins | | |
|-----|----|----|
| 14 | ls | [] |
| 13 | e | 0 |
| 12 | pc | 2 |

| ins | | |
|-----|----|-------------|
| 7 | i | [0,3] [3,0] |
| 6 | ls | [2,3] |
| 5 | e | 0 |
| 4 | pc | 2 3 4 5 4 |

| ins | | |
|-----|----|---------------|
| 3 | i | [2,0,3] - |
| 2 | ls | [1,2,3] |
| 1 | e | 0 |
| 0 | pc | 2 3 4 5 4 5 4 |

Se procede a realizar el yield de la linea 5 y se guarda el resultado en i3

Imprime: [0,1,2,3], [1,0,2,3], [1,2,0,3]

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

GLOBAL

| ins | proc |
|-----|---------------|
| i | [1,2,0,3] - |
| pc | 7 8 7 8 7 8 7 |

| ins | | |
|-----|----|---------|
| 11 | i | [0] |
| 10 | ls | [3] |
| 9 | e | 0 |
| 8 | pc | 2 3 4 5 |

| ins | | |
|-----|----|----|
| 14 | ls | [] |
| 13 | e | 0 |
| 12 | pc | 2 |

| ins | | |
|-----|----|-------------|
| 7 | i | [0,3] [3,0] |
| 6 | ls | [2,3] |
| 5 | e | 0 |
| 4 | pc | 2 3 4 5 4 5 |

| ins | | |
|-----|----|-----------------|
| 3 | i | [2,0,3] [2,3,0] |
| 2 | ls | [1,2,3] |
| 1 | e | 0 |
| 0 | pc | 2 3 4 5 4 5 4 |

Se procede a realizar el yield de la linea 5 y se guarda el resultado en i global

Imprime: [0,1,2,3], [1,0,2,3], [1,2,0,3]

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

| ins | | |
|-----|----|---------|
| 11 | i | [0] |
| 10 | ls | [3] |
| 9 | e | 0 |
| 8 | pc | 2 3 4 5 |

| ins | | |
|-----|----|----|
| 14 | ls | [] |
| 13 | e | 0 |
| 12 | pc | 2 |

| ins | | |
|-----|----|-------------|
| 7 | i | [0,3] [3,0] |
| 6 | ls | [2,3] |
| 5 | e | 0 |
| 4 | pc | 2 3 4 5 4 5 |

| ins | | |
|-----|----|-----------------|
| 3 | i | [2,0,3] [2,3,0] |
| 2 | ls | [1,2,3] |
| 1 | e | 0 |
| 0 | pc | 2 3 4 5 4 5 4 5 |

| GLOBAL | |
|--------|---------------------|
| ins | proc |
| i | [1,2,0,3] [1,2,3,0] |
| pc | 7 8 7 8 7 8 7 |

Se imprime i global

Imprime: [0,1,2,3]; [1,0,2,3]; [1,2,0,3]; [1,2,3,0]

```

1  def ins(e, ls):
2      yield [e, *ls]
3      if ls:
4          for i in ins(e, ls[1:]):
5              yield [ls[0], *i]
6
7  for i in ins(0, [1, 2, 3]):
8      print i

```

| GLOBAL | |
|--------|---------------------|
| ins | proc |
| i | [1,2,0,3] [1,2,3,0] |
| pc | 7 8 7 8 7 8 7 8 |

| ins | | |
|-----|----|---------|
| 11 | i | [0] |
| 10 | ls | [3] |
| 9 | e | 0 |
| 8 | pc | 2 3 4 5 |

| ins | | |
|-----|----|-------------|
| 7 | i | [0,3] [3,0] |
| 6 | ls | [2,3] |
| 5 | e | 0 |
| 4 | pc | 2 3 4 5 4 5 |

| ins | | |
|-----|----|-----------------|
| 3 | i | [2,0,3] [2,3,0] |
| 2 | ls | [1,2,3] |
| 1 | e | 0 |
| 0 | pc | 2 3 4 5 4 5 4 5 |

Y seguimos otra vez en el ins de la linea 7

Imprime: [0,1,2,3]; [1,0,2,3]; [1,2,0,3]; [1,2,3,0]

```

1  def ins(e, ls):
2      yield [e, *ls]
3      if ls:
4          for i in ins(e, ls[1:]):
5              yield [ls[0], *i]
6
7  for i in ins(0, [1, 2, 3]):
8      print i

```

| GLOBAL | |
|--------|-------------------|
| ins | proc |
| i | [1,2,3,0] - |
| pc | 7 8 7 8 7 8 7 8 7 |

| ins | | |
|-----|----|---------|
| 11 | i | [0] |
| 10 | ls | [3] |
| 9 | e | 0 |
| 8 | pc | 2 3 4 5 |

| ins | | |
|-----|----|-------------|
| 7 | i | [0,3] [3,0] |
| 6 | ls | [2,3] |
| 5 | e | 0 |
| 4 | pc | 2 3 4 5 4 5 |

| ins | | |
|-----|----|-----------------|
| 3 | i | [2,0,3] [2,3,0] |
| 2 | ls | [1,2,3] |
| 1 | e | 0 |
| 0 | pc | 2 3 4 5 4 5 4 5 |

Volvemos al ins de la linea 4

Imprime: [0,1,2,3]; [1,0,2,3]; [1,2,0,3]; [1,2,3,0]

```

1 def ins(e, ls):
2     yield [e, *ls]
3     if ls:
4         for i in ins(e, ls[1:]):
5             yield [ls[0], *i]
6
7 for i in ins(0, [1, 2, 3]):
8     print i

```

GLOBAL

| ins | proc |
|-----|-------------------|
| i | [1,2,3,0] - |
| pc | 7 8 7 8 7 8 7 8 7 |

| ins | | |
|-----|----|---------|
| 11 | i | [0] |
| 10 | ls | [3] |
| 9 | e | 0 |
| 8 | pc | 2 3 4 5 |

| ins | | |
|-----|----|----|
| 14 | ls | [] |
| 13 | e | 0 |
| 12 | pc | 2 |

| ins | | |
|-----|----|-------------|
| 7 | i | [0,3] [3,0] |
| 6 | ls | [2,3] |
| 5 | e | 0 |
| 4 | pc | 2 3 4 5 4 5 |

| ins | | |
|-----|----|-------------------|
| 3 | i | [2,0,3] [2,3,0] - |
| 2 | ls | [1,2,3] |
| 1 | e | 0 |
| 0 | pc | 2 3 4 5 4 5 4 5 4 |

Volvemos al ins de la linea 4 (again)

Imprime: [0,1,2,3]; [1,0,2,3]; [1,2,0,3]; [1,2,3,0]

```

1  def ins(e, ls):
2      yield [e, *ls]
3      if ls:
4          for i in ins(e, ls[1:]):
5              yield [ls[0], *i]
6
7  for i in ins(0, [1, 2, 3]):
8      print i

```

GLOBAL

| ins | proc |
|-----|-------------------|
| i | [1,2,3,0] - |
| pc | 7 8 7 8 7 8 7 8 7 |

| ins | | |
|-----|----|---------|
| 11 | i | [0] |
| 10 | ls | [3] |
| 9 | e | 0 |
| 8 | pc | 2 3 4 5 |

| ins | | |
|-----|----|---------------|
| 7 | i | [0,3] [3,0] - |
| 6 | ls | [2,3] |
| 5 | e | 0 |
| 4 | pc | 2 3 4 5 4 5 4 |

| ins | | |
|-----|----|-------------------|
| 3 | i | [2,0,3] [2,3,0] - |
| 2 | ls | [1,2,3] |
| 1 | e | 0 |
| 0 | pc | 2 3 4 5 4 5 4 |

Volvemos al ins de la linea 4 (again) (again)

Imprime: [0,1,2,3]; [1,0,2,3]; [1,2,0,3]; [1,2,3,0]

```

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

GLOBAL

| ins | proc |
|-----|-------------------|
| i | [1,2,3,0] - |
| pc | 7 8 7 8 7 8 7 8 7 |

| ins | | |
|-----|----|-----------|
| 11 | i | [θ] - |
| 10 | ls | [3] |
| 9 | e | 0 |
| 8 | pc | 2 3 4 5 4 |

| ins | | |
|-----|----|----|
| 14 | ls | [] |
| 13 | e | 0 |
| 12 | pc | 2 |

| ins | | |
|-----|----|---------------|
| 7 | i | [θ,3] [3,0] - |
| 6 | ls | [2,3] |
| 5 | e | 0 |
| 4 | pc | 2 3 4 5 4 5 4 |

| ins | | |
|-----|----|-------------------|
| 3 | i | [2,0,3] [2,3,0] - |
| 2 | ls | [1,2,3] |
| 1 | e | 0 |
| 0 | pc | 2 3 4 5 4 5 4 5 4 |

Se evalua if de la linea 3. No se cumple, ins termina ejecucion :c

Imprime: [0,1,2,3]; [1,0,2,3]; [1,2,0,3]; [1,2,3,0]

```

1  def ins(e, ls):
2      yield [e, *ls]
3      if ls:
4          for i in ins(e, ls[1:]):
5              yield [ls[0], *i]
6
7  for i in ins(0, [1, 2, 3]):
8      print i

```

| GLOBAL | |
|--------|-------------------|
| ins | proc |
| i | [1,2,3,0] - |
| pc | 7 8 7 8 7 8 7 8 7 |

| ins | | |
|-----|----|-----------|
| 11 | i | [θ] - |
| 10 | ls | [3] |
| 9 | e | 0 |
| 8 | pc | 2 3 4 5 4 |

| ins | | |
|-----|----|----|
| 14 | ls | [] |
| 13 | 0 | 0 |
| 12 | pc | 2 |

| ins | | |
|-----|----|---------------|
| 7 | i | [θ,3] [3,0] - |
| 6 | ls | [2,3] |
| 5 | e | 0 |
| 4 | pc | 2 3 4 5 4 5 4 |

| ins | | |
|-----|----|-------------------|
| 3 | i | [2,0,3] [2,3,0] - |
| 2 | ls | [1,2,3] |
| 1 | e | 0 |
| 0 | pc | 2 3 4 5 4 5 4 |

Ins no retorna nada por tanto termina ejecucion el for y a su vez ins

Imprime: [0,1,2,3]; [1,0,2,3]; [1,2,0,3]; [1,2,3,0]

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

GLOBAL

| ins | proc |
|-----|-------------------|
| i | [1,2,3,0] - |
| pc | 7 8 7 8 7 8 7 8 7 |

| ins | |
|-----|-----|
| 11 | |
| 10 | [3] |
| 9 | |
| 8 | pc |

| ins | |
|-----|----|
| 14 | [] |
| 13 | 0 |
| 12 | pc |
| 2 | |

| ins | |
|-----|----|
| 7 | i |
| 6 | ls |
| 5 | e |
| 4 | pc |

[0,3] [3,0] -
[2,3]
0
2 3 4 5 4 5 4

| ins | |
|-----|----|
| 3 | i |
| 2 | ls |
| 1 | e |
| 0 | pc |

[2,0,3] [2,3,0] -
[1,2,3]
0
2 3 4 5 4 5 4 5 4

Ins no retorna nada por tanto termina ejecucion el for y a su vez ins(again)

Imprime: [0,1,2,3]; [1,0,2,3]; [1,2,0,3]; [1,2,3,0]

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

GLOBAL

| | |
|-----|-------------------|
| ins | proc |
| i | [1,2,3,0] - |
| pc | 7 8 7 8 7 8 7 8 7 |

| ins | |
|-----|-----|
| 11 | [3] |
| 10 | pc |
| 9 | z |
| 8 | 5 4 |
| 14 | ins |
| 13 | [] |
| 12 | 0 |
| 2 | pc |

| ins | |
|-----|---------|
| 7 | [2,0] - |
| 6 | [2,3] |
| 5 | 0 |
| 4 | pc |
| 2 | z |
| 1 | 4 5 4 |

| ins | |
|-----|-------------------|
| 3 | i |
| 2 | ls |
| 1 | e |
| 0 | pc |
| 2 | [2,0,3] [2,3,0] - |
| 3 | [1,2,3] |
| 4 | 0 |
| 5 | 2 3 4 5 4 5 4 5 4 |

Ins no retorna nada por tanto termina ejecucion el for y a su vez ins(again)(again)

Imprime: [0,1,2,3]; [1,0,2,3]; [1,2,0,3]; [1,2,3,0]

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

GLOBAL

| | |
|-----|-------------------|
| ins | proc |
| i | [1,2,3,0] - |
| pc | 7 8 7 8 7 8 7 8 7 |



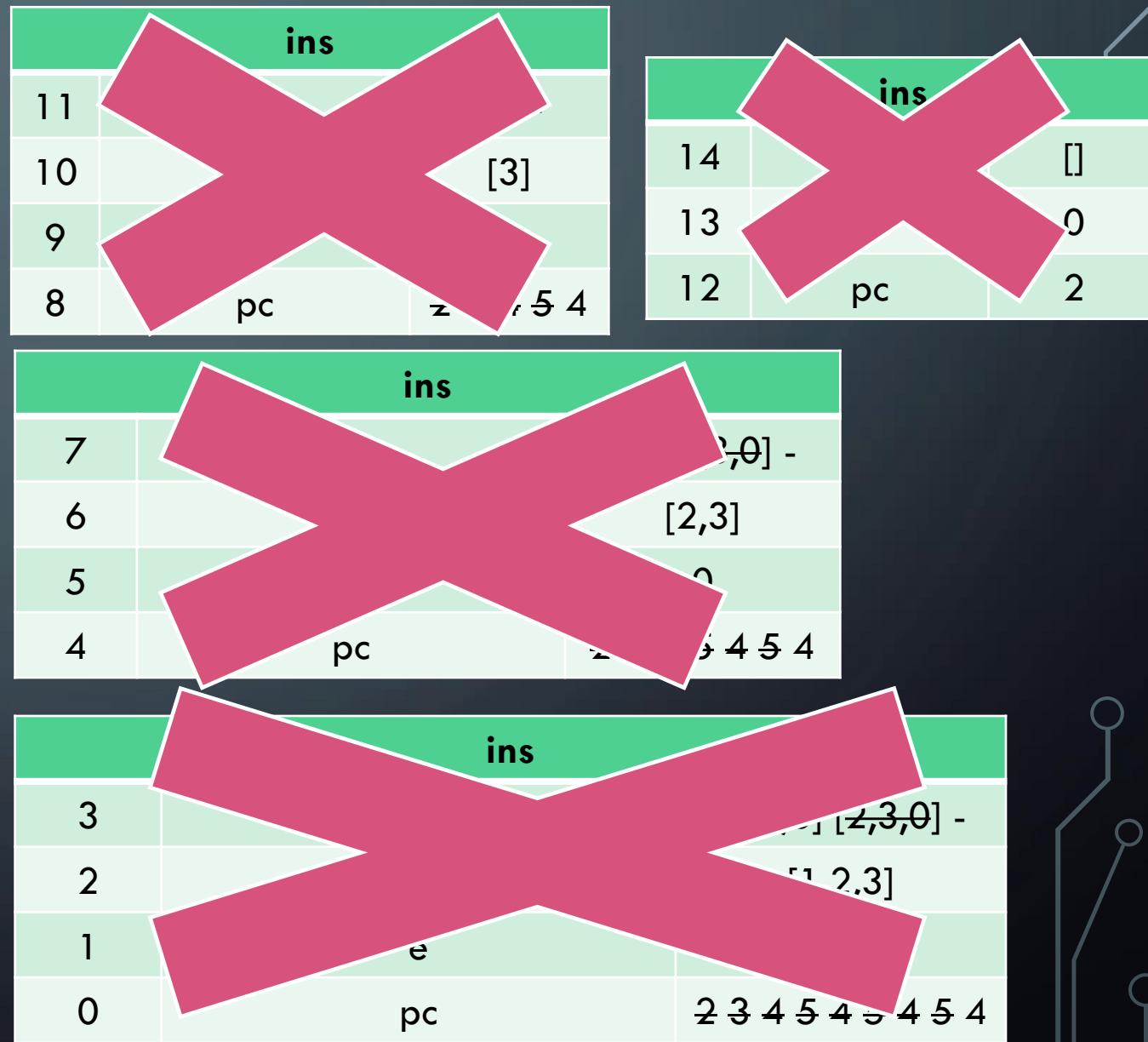
Al ins de la linea 7 no retornar nada. Termina ejecución el programa.

Imprime: [0,1,2,3]; [1,0,2,3]; [1,2,0,3]; [1,2,3,0]

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

GLOBAL

| | |
|-----|-------------------|
| ins | proc |
| i | [1,2,3,0] - |
| pc | 7 8 7 8 7 8 7 8 7 |



El programa al final Imprime

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

Imprime:

[0,1,2,3];

[1,0,2,3];

[1,2,0,3];

[1,2,3,0]

El algoritmo ins, lo que hace es que dado un valor e y una lista ls, devuelve las listas con e en todas las posiciones.

EJECUTEMOS AHORA ESTE MISTERIO

```
1 def misterio(ls):
2     if ls:
3         for m in misterio(ls[1:]):
4             for i in ins(ls[0], m):
5                 yield i
6     else:
7         yield []
8
9 for m in misterio([1,2,3]):
10    print m
```

No ejecutaremos el marco de ins, ya que eso esta explicado en el corrida anterior.

INICIAMOS EL LA CORRIDA EN ESTE ESTADO

Imprime:

```
1 def misterio(ls):
2     if ls:
3         for m in misterio(ls[1:]):
4             for i in ins(ls[0], m):
5                 yield i
6     else:
7         yield []
8
9 for m in misterio([1,2,3]):
10    print m
```

GLOBAL

| misterio | proc |
|----------|------|
| m | - |
| pc | 9 |

Entramos al misterio de la linea 9

Imprime:

```
1 def misterio(ls):
2     if ls:
3         for m in misterio(ls[1:]):
4             for i in ins(ls[0], m):
5                 yield i
6     else:
7         yield []
8
9 for m in misterio([1,2,3]):
10    print m
```

GLOBAL

| | |
|----------|------|
| misterio | proc |
| m | - |
| pc | 9 |

misterio

| | | |
|---|----|---------|
| 1 | ls | [1,2,3] |
| 0 | pc | 2 |

Verificamos el if de la linea 2. Se cumple, y se crea una nueva variable m

Imprime:

```
1 def misterio(ls):
2     if ls:
3         for m in misterio(ls[1:]):
4             for i in ins(ls[0], m):
5                 yield i
6     else:
7         yield []
8
9 for m in misterio([1,2,3]):
10    print m
```

GLOBAL

| | |
|----------|------|
| misterio | proc |
|----------|------|

| | |
|---|---|
| m | - |
|---|---|

| | |
|----|---|
| pc | 9 |
|----|---|

misterio

| | | |
|---|---|---|
| 2 | m | - |
|---|---|---|

| | | |
|---|----|---------|
| 1 | ls | [1,2,3] |
|---|----|---------|

| | | |
|---|----|---|
| 0 | pc | 2 |
|---|----|---|

Llamamos al misterio de la linea 3

Imprime:

```
1 def misterio(ls):
2     if ls:
3         for m in misterio(ls[1:]):
4             for i in ins(ls[0], m):
5                 yield i
6     else:
7         yield []
8
9 for m in misterio([1,2,3]):
10    print m
```

| misterio | | |
|----------|----|-------|
| 4 | ls | [2,3] |
| 3 | pc | 2 |

| GLOBAL | |
|----------|------|
| misterio | proc |
| m | - |
| pc | 9 |

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

Verificamos el if de la linea 2. Se cumple y se crea una nueva variable m

Imprime:

```
1 def misterio(ls):
2     if ls:
3         for m in misterio(ls[1:]):
4             for i in ins(ls[0], m):
5                 yield i
6     else:
7         yield []
8
9 for m in misterio([1,2,3]):
10    print m
```

| misterio | | |
|----------|----|-------|
| 5 | m | - |
| 4 | ls | [2,3] |
| 3 | pc | 2 |

| GLOBAL | |
|----------|------|
| misterio | proc |
| m | - |
| pc | 9 |

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

Se ejecuta el misterio de la linea 3

Imprime:

```
1 def misterio(ls):
2     if ls:
3         for m in misterio(ls[1:]):
4             for i in ins(ls[0], m):
5                 yield i
6     else:
7         yield []
8
9 for m in misterio([1,2,3]):
10    print m
```

| misterio | | |
|----------|----|-----|
| 7 | ls | [3] |
| 6 | pc | 2 |

| misterio | | |
|----------|----|-------|
| 5 | m | - |
| 4 | ls | [2,3] |
| 3 | pc | 2 3 |

| GLOBAL | | |
|----------|--|------|
| misterio | | proc |
| m | | - |
| pc | | 9 |

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

Se verifica el if de la linea 2. Se cumple y se crea una nueva m

Imprime:

```
1 def misterio(ls):
2     if ls:
3         for m in misterio(ls[1:]):
4             for i in ins(ls[0], m):
5                 yield i
6     else:
7         yield []
8
9 for m in misterio([1,2,3]):
10    print m
```

| misterio | | |
|----------|----|-----|
| 8 | m | - |
| 7 | ls | [3] |
| 6 | pc | 2 |

| misterio | | |
|----------|----|-------|
| 5 | m | - |
| 4 | ls | [2,3] |
| 3 | pc | 2 3 |

| GLOBAL | | |
|----------|------|--|
| misterio | proc | |
| m | - | |
| pc | 9 | |

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

Se llama nuevamente al misterio de la linea 3 (Parece esto una pelicula)

Imprime:

```
1 def misterio(ls):
2     if ls:
3         for m in misterio(ls[1:]):
4             for i in ins(ls[0], m):
5                 yield i
6     else:
7         yield []
8
9 for m in misterio([1,2,3]):
10    print m
```

| misterio | | |
|----------|----|-----|
| 8 | m | - |
| 7 | ls | [3] |
| 6 | pc | 2 3 |

| misterio | | |
|----------|----|-------|
| 5 | m | - |
| 4 | ls | [2,3] |
| 3 | pc | 2 3 |

| GLOBAL | | |
|----------|------|--|
| misterio | proc | |
| m | - | |
| pc | 9 | |

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

| misterio | | |
|----------|----|----|
| 10 | ls | [] |
| 9 | pc | 2 |

Se verifica el if de la linea 2. No se cumple. Se retorna a m8 [] y se muere un misterio

Imprime:

```
1 def misterio(ls):
2     if ls:
3         for m in misterio(ls[1:]):
4             for i in ins(ls[0], m):
5                 yield i
6     else:
7         yield []
8
9 for m in misterio([1,2,3]):
10    print m
```

| misterio | | |
|----------|----|-----|
| 8 | m | [] |
| 7 | ls | [3] |
| 6 | pc | 2 3 |

| misterio | | |
|----------|----|-------|
| 5 | m | - |
| 4 | ls | [2,3] |
| 3 | pc | 2 3 |

| GLOBAL | | |
|----------|------|--|
| misterio | proc | |
| m | - | |
| pc | 9 | |

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

| misterio | | |
|----------|----|---|
| 10 | ls | |
| 9 | pc | 2 |

Al m tener un valor obtenemos una variable i que depende de ins

Imprime:

```
1 def misterio(ls):
2     if ls:
3         for m in misterio(ls[1:]):
4             for i in ins(ls[0], m):
5                 yield i
6     else:
7         yield []
8
9 for m in misterio([1,2,3]):
10    print m
```

| misterio | | |
|----------|----|-----|
| 9 | i | - |
| 8 | m | [] |
| 7 | ls | [3] |
| 6 | pc | 2 3 |

| misterio | | |
|----------|----|-------|
| 5 | m | - |
| 4 | ls | [2,3] |
| 3 | pc | 2 3 |

| GLOBAL | | |
|----------|------|--|
| misterio | proc | |
| m | - | |
| pc | 9 | |

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

Ins retorna en su primera llamada [3] a i9

Imprime:

```
1 def misterio(ls):
2     if ls:
3         for m in misterio(ls[1:]):
4             for i in ins(ls[0], m):
5                 yield i
6     else:
7         yield []
8
9 for m in misterio([1,2,3]):
10    print m
```

| misterio | | |
|----------|----|-------|
| 9 | i | [3] |
| 8 | m | [] |
| 7 | ls | [3] |
| 6 | pc | 2 3 4 |

| misterio | | |
|----------|----|-------|
| 5 | m | - |
| 4 | ls | [2,3] |
| 3 | pc | 2 3 |

| GLOBAL | | |
|----------|--|------|
| misterio | | proc |
| m | | - |
| pc | | 9 |

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

Se ejecuta la linea 5 y se retorna el valor de i9 que se almacena en m5

Imprime:

```
1 def misterio(ls):
2     if ls:
3         for m in misterio(ls[1:]):
4             for i in ins(ls[0], m):
5                 yield i
6     else:
7         yield []
8
9 for m in misterio([1,2,3]):
10    print m
```

| misterio | | | |
|----------|----|--|---------|
| 9 | i | | [3] |
| 8 | m | | [] |
| 7 | ls | | [3] |
| 6 | pc | | 2 3 4 5 |

| misterio | | | |
|----------|----|--|-------|
| 5 | m | | [3] |
| 4 | ls | | [2,3] |
| 3 | pc | | 2 3 |

| GLOBAL | | | |
|----------|--|------|--|
| misterio | | proc | |
| m | | - | |
| pc | | 9 | |

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

Se ejecuta la linea 4 al ya poseer valor de m5 y se obtiene nueva variable i10

Imprime:

```
1 def misterio(ls):
2     if ls:
3         for m in misterio(ls[1:]):
4             for i in ins(ls[0], m):
5                 yield i
6     else:
7         yield []
8
9 for m in misterio([1,2,3]):
10    print m
```

GLOBAL

| | |
|----------|------|
| misterio | proc |
| m | - |
| pc | 9 |

misterio

| | | |
|---|----|---------|
| 9 | i | [3] |
| 8 | m | [] |
| 7 | ls | [3] |
| 6 | pc | 2 3 4 5 |

misterio

| | | |
|----|----|-------|
| 10 | i | - |
| 5 | m | [3] |
| 4 | ls | [2,3] |
| 3 | pc | 2 3 4 |

misterio

| | | |
|---|----|---------|
| 2 | m | - |
| 1 | ls | [1,2,3] |
| 0 | pc | 2 3 |

Se ejecuta el ins de la linea 4 y retorna [2,3] que se almacena en i10

Imprime:

```
1 def misterio(ls):
2     if ls:
3         for m in misterio(ls[1:]):
4             for i in ins(ls[0], m):
5                 yield i
6     else:
7         yield []
8
9 for m in misterio([1,2,3]):
10    print m
```

| misterio | | | |
|----------|----|---------|--|
| 9 | i | [3] | |
| 8 | m | [] | |
| 7 | ls | [3] | |
| 6 | pc | 2 3 4 5 | |

| misterio | | | |
|----------|----|-------|--|
| 10 | i | [2,3] | |
| 5 | m | [3] | |
| 4 | ls | [2,3] | |
| 3 | pc | 2 3 4 | |

| GLOBAL | | | |
|----------|--|------|--|
| misterio | | proc | |
| m | | - | |
| pc | | 9 | |

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

Se ejecuta la linea 5 y se retorna el valor de i10 y se almacena en m2

Imprime:

```
1 def misterio(ls):
2     if ls:
3         for m in misterio(ls[1:]):
4             for i in ins(ls[0], m):
5                 yield i
6     else:
7         yield []
8
9 for m in misterio([1,2,3]):
10    print m
```

GLOBAL

| | |
|----------|------|
| misterio | proc |
| m | - |
| pc | 9 |

misterio

| | | |
|---|----|---------|
| 9 | i | [3] |
| 8 | m | [] |
| 7 | ls | [3] |
| 6 | pc | 2 3 4 5 |

misterio

| | | |
|----|----|---------|
| 10 | i | [2,3] |
| 5 | m | [3] |
| 4 | ls | [2,3] |
| 3 | pc | 2 3 4 5 |

misterio

| | | |
|---|----|---------|
| 2 | m | [2,3] |
| 1 | ls | [1,2,3] |
| 0 | pc | 2 3 |

Se ejecuta la linea 4. Obteniendo una nueva variable i11

Imprime:

```
1 def misterio(ls):
2     if ls:
3         for m in misterio(ls[1:]):
4             for i in ins(ls[0], m):
5                 yield i
6     else:
7         yield []
8
9 for m in misterio([1,2,3]):
10    print m
```

GLOBAL

| | |
|----------|------|
| misterio | proc |
| m | - |
| pc | 9 |

| misterio | | | |
|----------|----|---------|--|
| 9 | i | [3] | |
| 8 | m | [] | |
| 7 | ls | [3] | |
| 6 | pc | 2 3 4 5 | |

| misterio | | | |
|----------|----|---------|--|
| 10 | i | [2,3] | |
| 5 | m | [3] | |
| 4 | ls | [2,3] | |
| 3 | pc | 2 3 4 5 | |

| misterio | | | |
|----------|----|---------|--|
| 11 | i | - | |
| 2 | m | [2,3] | |
| 1 | ls | [1,2,3] | |
| 0 | pc | 2 3 4 | |

Se llama a ins y este retorna [1,2,3] que se guarda en i11

Imprime:

```
1 def misterio(ls):
2     if ls:
3         for m in misterio(ls[1:]):
4             for i in ins(ls[0], m):
5                 yield i
6     else:
7         yield []
8
9 for m in misterio([1,2,3]):
10    print m
```

GLOBAL

| | |
|----------|------|
| misterio | proc |
| m | - |
| pc | 9 |

| misterio | | | |
|----------|----|---------|--|
| 9 | i | [3] | |
| 8 | m | [] | |
| 7 | ls | [3] | |
| 6 | pc | 2 3 4 5 | |

| misterio | | | |
|----------|----|---------|--|
| 10 | i | [2,3] | |
| 5 | m | [3] | |
| 4 | ls | [2,3] | |
| 3 | pc | 2 3 4 5 | |

| misterio | | | |
|----------|----|---------|--|
| 11 | i | [1,2,3] | |
| 2 | m | [2,3] | |
| 1 | ls | [1,2,3] | |
| 0 | pc | 2 3 4 | |

Se llama al yield de la linea 5 y se guarda el valor de i11 en m global

Imprime:

```
1 def misterio(ls):
2     if ls:
3         for m in misterio(ls[1:]):
4             for i in ins(ls[0], m):
5                 yield i
6     else:
7         yield []
8
9 for m in misterio([1,2,3]):
10    print m
```

GLOBAL

| | |
|----------|---------|
| misterio | proc |
| m | [1,2,3] |
| pc | 9 |

| misterio | | | |
|----------|----|---------|--|
| 9 | i | [3] | |
| 8 | m | [] | |
| 7 | ls | [3] | |
| 6 | pc | 2 3 4 5 | |

| misterio | | | |
|----------|----|---------|--|
| 10 | i | [2,3] | |
| 5 | m | [3] | |
| 4 | ls | [2,3] | |
| 3 | pc | 2 3 4 5 | |

| misterio | | | |
|----------|----|---------|--|
| 11 | i | [1,2,3] | |
| 2 | m | [2,3] | |
| 1 | ls | [1,2,3] | |
| 0 | pc | 2 3 4 5 | |

Imprimimos m global

Imprime: [1,2,3]

```
1 def misterio(ls):
2     if ls:
3         for m in misterio(ls[1:]):
4             for i in ins(ls[0], m):
5                 yield i
6     else:
7         yield []
8
9 for m in misterio([1,2,3]):
10    print m
```

GLOBAL

| | |
|----------|---------|
| misterio | proc |
| m | [1,2,3] |
| pc | ♀ 10 |

misterio

| | | |
|---|----|---------|
| 9 | i | [3] |
| 8 | m | [] |
| 7 | ls | [3] |
| 6 | pc | ♀ 3 4 5 |

misterio

| | | |
|----|----|---------|
| 10 | i | [2,3] |
| 5 | m | [3] |
| 4 | ls | [2,3] |
| 3 | pc | ♀ 3 4 5 |

misterio

| | | |
|----|----|---------|
| 11 | i | [1,2,3] |
| 2 | m | [2,3] |
| 1 | ls | [1,2,3] |
| 0 | pc | ♀ 3 4 5 |

Volvemos al misterio de la linea 9

Imprime: [1,2,3]

```
1 def misterio(ls):
2     if ls:
3         for m in misterio(ls[1:]):
4             for i in ins(ls[0], m):
5                 yield i
6     else:
7         yield []
8
9 for m in misterio([1,2,3]):
10    print m
```

GLOBAL

| | |
|----------|-----------|
| misterio | proc |
| m | [1,2,3] - |
| pc | 9 10 9 |

| misterio | | | |
|----------|----|---------|--|
| 9 | i | [3] | |
| 8 | m | [] | |
| 7 | ls | [3] | |
| 6 | pc | 2 3 4 5 | |

| misterio | | | |
|----------|----|---------|--|
| 10 | i | [2,3] | |
| 5 | m | [3] | |
| 4 | ls | [2,3] | |
| 3 | pc | 2 3 4 5 | |

| misterio | | | |
|----------|----|---------|--|
| 11 | i | [1,2,3] | |
| 2 | m | [2,3] | |
| 1 | ls | [1,2,3] | |
| 0 | pc | 2 3 4 5 | |

Volvemos a revisar ins de la linea 4. Esta vez retorna [2,1,3]

Imprime: [1,2,3]

```
1 def misterio(ls):
2     if ls:
3         for m in misterio(ls[1:]):
4             for i in ins(ls[0], m):
5                 yield i
6     else:
7         yield []
8
9 for m in misterio([1,2,3]):
10    print m
```

GLOBAL

| | |
|----------|-----------|
| misterio | proc |
| m | [1,2,3] - |
| pc | 9 10 9 |

| misterio | | | |
|----------|----|---------|--|
| 9 | i | [3] | |
| 8 | m | [] | |
| 7 | ls | [3] | |
| 6 | pc | 2 3 4 5 | |

| misterio | | | |
|----------|----|---------|--|
| 10 | i | [2,3] | |
| 5 | m | [3] | |
| 4 | ls | [2,3] | |
| 3 | pc | 2 3 4 5 | |

| misterio | | | |
|----------|----|-----------------|--|
| 11 | i | [1,2,3] [2,1,3] | |
| 2 | m | [2,3] | |
| 1 | ls | [1,2,3] | |
| 0 | pc | 2 3 4 5 4 | |

Se le retorna por la linea 5 el valor de i11 a m global

Imprime: [1,2,3]

```
1 def misterio(ls):
2     if ls:
3         for m in misterio(ls[1:]):
4             for i in ins(ls[0], m):
5                 yield i
6     else:
7         yield []
8
9 for m in misterio([1,2,3]):
10    print m
```

GLOBAL

| | |
|----------|-----------------|
| misterio | proc |
| m | [1,2,3] [2,1,3] |
| pc | 9 10 9 |

| misterio | | | |
|----------|----|---------|--|
| 9 | i | [3] | |
| 8 | m | [] | |
| 7 | ls | [3] | |
| 6 | pc | 2 3 4 5 | |

| misterio | | | |
|----------|----|---------|--|
| 10 | i | [2,3] | |
| 5 | m | [3] | |
| 4 | ls | [2,3] | |
| 3 | pc | 2 3 4 5 | |

| misterio | | | |
|----------|----|-----------------|--|
| 11 | i | [1,2,3] [2,1,3] | |
| 2 | m | [2,3] | |
| 1 | ls | [1,2,3] | |
| 0 | pc | 2 3 4 5 4 5 | |

Imprimimos m global

Imprime: [1,2,3] , [2,1,3]

```
1 def misterio(ls):
2     if ls:
3         for m in misterio(ls[1:]):
4             for i in ins(ls[0], m):
5                 yield i
6     else:
7         yield []
8
9 for m in misterio([1,2,3]):
10    print m
```

GLOBAL

| | |
|----------|-----------------|
| misterio | proc |
| m | [1,2,3] [2,1,3] |
| pc | 9 10 9 10 |

| misterio | | | |
|----------|----|---------|--|
| 9 | i | [3] | |
| 8 | m | [] | |
| 7 | ls | [3] | |
| 6 | pc | 2 3 4 5 | |

| misterio | | | |
|----------|----|---------|--|
| 10 | i | [2,3] | |
| 5 | m | [3] | |
| 4 | ls | [2,3] | |
| 3 | pc | 2 3 4 5 | |

| misterio | | | |
|----------|----|-----------------|--|
| 11 | i | [1,2,3] [2,1,3] | |
| 2 | m | [2,3] | |
| 1 | ls | [1,2,3] | |
| 0 | pc | 2 3 4 5 4 5 | |

Volvemos con el misterio de la linea 9

Imprime: [1,2,3] , [2,1,3]

```
1 def misterio(ls):
2     if ls:
3         for m in misterio(ls[1:]):
4             for i in ins(ls[0], m):
5                 yield i
6     else:
7         yield []
8
9 for m in misterio([1,2,3]):
10    print m
```

GLOBAL

| | |
|----------|-------------------|
| misterio | proc |
| m | [1,2,3] [2,1,3] - |
| pc | 9 10 9 10 9 |

| misterio | | | |
|----------|----|---------|--|
| 9 | i | [3] | |
| 8 | m | [] | |
| 7 | ls | [3] | |
| 6 | pc | 2 3 4 5 | |

| misterio | | | |
|----------|----|---------|--|
| 10 | i | [2,3] | |
| 5 | m | [3] | |
| 4 | ls | [2,3] | |
| 3 | pc | 2 3 4 5 | |

| misterio | | | |
|----------|----|-----------------|--|
| 11 | i | [1,2,3] [2,1,3] | |
| 2 | m | [2,3] | |
| 1 | ls | [1,2,3] | |
| 0 | pc | 2 3 4 5 4 5 | |

Se llama nuevamente al ins de la linea 4. Esta vez retorna [2,3,1]

Imprime: [1,2,3] , [2,1,3]

```
1 def misterio(ls):
2     if ls:
3         for m in misterio(ls[1:]):
4             for i in ins(ls[0], m):
5                 yield i
6     else:
7         yield []
8
9 for m in misterio([1,2,3]):
10    print m
```

GLOBAL

| | |
|----------|-------------------|
| misterio | proc |
| m | [1,2,3] [2,1,3] - |
| pc | 9 10 9 10 9 |

misterio

| | | |
|---|----|---------|
| 9 | i | [3] |
| 8 | m | [] |
| 7 | ls | [3] |
| 6 | pc | 2 3 4 5 |

misterio

| | | |
|----|----|---------|
| 10 | i | [2,3] |
| 5 | m | [3] |
| 4 | ls | [2,3] |
| 3 | pc | 2 3 4 5 |

misterio

| | | |
|----|----|-------------------------|
| 11 | i | [1,2,3] [2,1,3] [2,3,1] |
| 2 | m | [2,3] |
| 1 | ls | [1,2,3] |
| 0 | pc | 2 3 4 5 4 5 4 |

Se retorna el valor de i11 a m_e global por la linea 5

Imprime: [1,2,3] , [2,1,3]

```
1 def misterio(ls):
2     if ls:
3         for m in misterio(ls[1:]):
4             for i in ins(ls[0], m):
5                 yield i
6     else:
7         yield []
8
9 for m in misterio([1,2,3]):
10    print m
```

GLOBAL

| | |
|----------|-------------------------|
| misterio | proc |
| m | [1,2,3] [2,1,3] [2,3,1] |
| pc | 9 10 9 10 9 |

misterio

| | | |
|---|----|---------|
| 9 | i | [3] |
| 8 | m | [] |
| 7 | ls | [3] |
| 6 | pc | 2 3 4 5 |

misterio

| | | |
|----|----|---------|
| 10 | i | [2,3] |
| 5 | m | [3] |
| 4 | ls | [2,3] |
| 3 | pc | 2 3 4 5 |

misterio

| | | |
|----|----|-------------------------|
| 11 | i | [1,2,3] [2,1,3] [2,3,1] |
| 2 | m | [2,3] |
| 1 | ls | [1,2,3] |
| 0 | pc | 2 3 4 5 4 5 4 5 |

Se imprime m global

Imprime: [1,2,3] , [2,1,3] , [2,3,1]

```
1 def misterio(ls):
2     if ls:
3         for m in misterio(ls[1:]):
4             for i in ins(ls[0], m):
5                 yield i
6     else:
7         yield []
8
9 for m in misterio([1,2,3]):
10    print m
```

GLOBAL

| | |
|----------|-------------------------|
| misterio | proc |
| m | [1,2,3] [2,1,3] [2,3,1] |
| pc | 9 10 9 10 9 10 |

misterio

| | | |
|---|----|---------|
| 9 | i | [3] |
| 8 | m | [] |
| 7 | ls | [3] |
| 6 | pc | 2 3 4 5 |

misterio

| | | |
|----|----|---------|
| 10 | i | [2,3] |
| 5 | m | [3] |
| 4 | ls | [2,3] |
| 3 | pc | 2 3 4 5 |

misterio

| | | |
|----|----|-------------------------|
| 11 | i | [1,2,3] [2,1,3] [2,3,1] |
| 2 | m | [2,3] |
| 1 | ls | [1,2,3] |
| 0 | pc | 2 3 4 5 4 5 4 5 |

Volvemos al misterio de la linea 9

Imprime: [1,2,3] , [2,1,3] , [2,3,1]

```
1 def misterio(ls):
2     if ls:
3         for m in misterio(ls[1:]):
4             for i in ins(ls[0], m):
5                 yield i
6     else:
7         yield []
8
9 for m in misterio([1,2,3]):
10    print m
```

GLOBAL

| | |
|----------|---------------------------|
| misterio | proc |
| m | [1,2,3] [2,1,3] [2,3,1] - |
| pc | 9 10 9 10 9 10 9 |

| misterio | | |
|----------|----|---------|
| 9 | i | [3] |
| 8 | m | [] |
| 7 | ls | [3] |
| 6 | pc | 2 3 4 5 |

| misterio | | |
|----------|----|---------|
| 10 | i | [2,3] |
| 5 | m | [3] |
| 4 | ls | [2,3] |
| 3 | pc | 2 3 4 5 |

| misterio | | |
|----------|----|-------------------------|
| 11 | i | [1,2,3] [2,1,3] [2,3,1] |
| 2 | m | [2,3] |
| 1 | ls | [1,2,3] |
| 0 | pc | 2 3 4 5 4 5 4 5 |

Volvemos a llamar a ins. Pero esta vez no retorna nada (termina su ejecucion)

Imprime: [1,2,3] , [2,1,3] , [2,3,1]

```
1 def misterio(ls):
2     if ls:
3         for m in misterio(ls[1:]):
4             for i in ins(ls[0], m):
5                 yield i
6     else:
7         yield []
8
9 for m in misterio([1,2,3]):
10    print m
```

GLOBAL

| | |
|----------|---------------------------|
| misterio | proc |
| m | [1,2,3] [2,1,3] [2,3,1] - |
| pc | 9 10 9 10 9 10 9 |

| misterio | | | |
|----------|----|---------|--|
| 9 | i | [3] | |
| 8 | m | [] | |
| 7 | ls | [3] | |
| 6 | pc | 2 3 4 5 | |

| misterio | | | |
|----------|----|---------|--|
| 10 | i | [2,3] | |
| 5 | m | [3] | |
| 4 | ls | [2,3] | |
| 3 | pc | 2 3 4 5 | |

| misterio | | | |
|----------|----|-------------------------|--|
| 11 | i | [1,2,3] [2,1,3] [2,3,1] | |
| 2 | m | [2,3] | |
| 1 | ls | [1,2,3] | |
| 0 | pc | 2 3 4 5 4 5 4 | |

Al no poder hacer más nada volvemos al misterio de la linea 3

Imprime: [1,2,3] , [2,1,3] , [2,3,1]

```
1 def misterio(ls):
2     if ls:
3         for m in misterio(ls[1:]):
4             for i in ins(ls[0], m):
5                 yield i
6     else:
7         yield []
8
9 for m in misterio([1,2,3]):
10    print m
```

GLOBAL

| | |
|----------|---------------------------|
| misterio | proc |
| m | [1,2,3] [2,1,3] [2,3,1] - |
| pc | 9 10 9 10 9 10 9 |

| misterio | | |
|----------|----|---------|
| 9 | i | [3] |
| 8 | m | [] |
| 7 | ls | [3] |
| 6 | pc | 2 3 4 5 |

| misterio | | |
|----------|----|---------|
| 10 | i | [2,3] |
| 5 | m | [3] |
| 4 | ls | [2,3] |
| 3 | pc | 2 3 4 5 |

| misterio | | |
|----------|----|---------------------|
| 11 | i | - |
| 2 | m | [2,3] - |
| 1 | ls | [1,2,3] |
| 0 | pc | 2 3 4 5 4 5 4 5 4 3 |

Se ejecuta nuevamente el ins de la linea 4. Retorna [3,2] y se guarda en i10

Imprime: [1,2,3], [2,1,3], [2,3,1]

```
1 def misterio(ls):
2     if ls:
3         for m in misterio(ls[1:]):
4             for i in ins(ls[0], m):
5                 yield i
6     else:
7         yield []
8
9 for m in misterio([1,2,3]):
10    print m
```

GLOBAL

| | |
|----------|---------------------------|
| misterio | proc |
| m | [1,2,3] [2,1,3] [2,3,1] - |
| pc | 9+10 9+10 9+10 9 |

misterio

| | | |
|---|----|---------|
| 9 | i | [3] |
| 8 | m | [] |
| 7 | ls | [3] |
| 6 | pc | 2 3 4 5 |

misterio

| | | |
|----|----|-------------|
| 10 | i | [2,3] [3,2] |
| 5 | m | [3] |
| 4 | ls | [2,3] |
| 3 | pc | 2 3 4 5 4 |

misterio

| | | |
|----|----|---------------------|
| 11 | i | - |
| 2 | m | [2,3] - |
| 1 | ls | [1,2,3] |
| 0 | pc | 2 3 4 5 4 5 4 5 4 3 |

Se retorna el valor de i10 y se guarda en m2

Imprime: [1,2,3] , [2,1,3] , [2,3,1]

```
1 def misterio(ls):
2     if ls:
3         for m in misterio(ls[1:]):
4             for i in ins(ls[0], m):
5                 yield i
6     else:
7         yield []
8
9 for m in misterio([1,2,3]):
10    print m
```

GLOBAL

| | |
|----------|---------------------------|
| misterio | proc |
| m | [1,2,3] [2,1,3] [2,3,1] - |
| pc | 9+10 9+10 9+10 9 |

| misterio | | | |
|----------|----|---------|--|
| 9 | i | [3] | |
| 8 | m | [] | |
| 7 | ls | [3] | |
| 6 | pc | 2 3 4 5 | |

| misterio | | | |
|----------|----|-------------|--|
| 10 | i | [2,3] [3,2] | |
| 5 | m | [3] | |
| 4 | ls | [2,3] | |
| 3 | pc | 2 3 4 5 4 5 | |

| misterio | | | |
|----------|----|---------------------|--|
| 11 | i | - | |
| 2 | m | [2,3] [3,2] | |
| 1 | ls | [1,2,3] | |
| 0 | pc | 2 3 4 5 4 5 4 5 4 3 | |

Se llama a ins de la linea 4. Retorna [1,3,2] y se guarda en i11

Imprime: [1,2,3], [2,1,3], [2,3,1]

```
1 def misterio(ls):
2     if ls:
3         for m in misterio(ls[1:]):
4             for i in ins(ls[0], m):
5                 yield i
6     else:
7         yield []
8
9 for m in misterio([1,2,3]):
10    print m
```

GLOBAL

| | |
|----------|---------------------------|
| misterio | proc |
| m | [1,2,3] [2,1,3] [2,3,1] - |
| pc | 9 10 9 10 9 10 9 |

| misterio | | | |
|----------|----|---------|--|
| 9 | i | [3] | |
| 8 | m | [] | |
| 7 | ls | [3] | |
| 6 | pc | 2 3 4 5 | |

| misterio | | | |
|----------|----|-------------|--|
| 10 | i | [2,3] [3,2] | |
| 5 | m | [3] | |
| 4 | ls | [2,3] | |
| 3 | pc | 2 3 4 5 4 5 | |

| misterio | | | |
|----------|----|-----------------------|--|
| 11 | i | [1,3,2] | |
| 2 | m | [2,3] [3,2] | |
| 1 | ls | [1,2,3] | |
| 0 | pc | 2 3 4 5 4 5 4 5 4 3 4 | |

Se retorna por la linea 5 el valir de i 11 a m global

Imprime: [1,2,3] , [2,1,3] , [2,3,1]

```
1 def misterio(ls):
2     if ls:
3         for m in misterio(ls[1:]):
4             for i in ins(ls[0], m):
5                 yield i
6     else:
7         yield []
8
9 for m in misterio([1,2,3]):
10    print m
```

GLOBAL

| | |
|----------|---------------------------------|
| misterio | proc |
| m | [1,2,3] [2,1,3] [2,3,1] [1,3,2] |
| pc | 9 10 9 10 9 10 9 |

misterio

| | | |
|---|----|---------|
| 9 | i | [3] |
| 8 | m | [] |
| 7 | ls | [3] |
| 6 | pc | 2 3 4 5 |

misterio

| | | |
|----|----|-------------|
| 10 | i | [2,3] [3,2] |
| 5 | m | [3] |
| 4 | ls | [2,3] |
| 3 | pc | 2 3 4 5 4 5 |

misterio

| | | |
|----|----|-------------------------|
| 11 | i | [1,3,2] |
| 2 | m | [2,3] [3,2] |
| 1 | ls | [1,2,3] |
| 0 | pc | 2 3 4 5 4 5 4 5 4 3 4 5 |

Se imprime m global

Imprime: [1,2,3] , [2,1,3] , [2,3,1], [1,3,2]

```
1 def misterio(ls):
2     if ls:
3         for m in misterio(ls[1:]):
4             for i in ins(ls[0], m):
5                 yield i
6     else:
7         yield []
8
9 for m in misterio([1,2,3]):
10    print m
```

GLOBAL

| | |
|----------|---------------------------------|
| misterio | proc |
| m | [1,2,3] [2,1,3] [2,3,1] [1,3,2] |
| pc | 9 10 9 10 9 10 9 10 |

misterio

| | | |
|---|----|---------|
| 9 | i | [3] |
| 8 | m | [] |
| 7 | ls | [3] |
| 6 | pc | 2 3 4 5 |

misterio

| | | |
|----|----|-------------|
| 10 | i | [2,3] [3,2] |
| 5 | m | [3] |
| 4 | ls | [2,3] |
| 3 | pc | 2 3 4 5 4 5 |

misterio

| | | |
|----|----|-------------------------|
| 11 | i | [1,3,2] |
| 2 | m | [2,3] [3,2] |
| 1 | ls | [1,2,3] |
| 0 | pc | 2 3 4 5 4 5 4 5 4 3 4 5 |

Y Volvemos con el misterio de la linea 9

Imprime: [1,2,3] , [2,1,3] , [2,3,1], [1,3,2]

```
1 def misterio(ls):
2     if ls:
3         for m in misterio(ls[1:]):
4             for i in ins(ls[0], m):
5                 yield i
6     else:
7         yield []
8
9 for m in misterio([1,2,3]):
10    print m
```

| misterio | | | |
|----------|----|---------|--|
| 9 | i | [3] | |
| 8 | m | [] | |
| 7 | ls | [3] | |
| 6 | pc | 2 3 4 5 | |

| misterio | | | |
|----------|----|-------------|--|
| 10 | i | [2,3] [3,2] | |
| 5 | m | [3] | |
| 4 | ls | [2,3] | |
| 3 | pc | 2 3 4 5 4 5 | |

| GLOBAL | | | |
|----------|--|-----------------------------------|--|
| misterio | | proc | |
| m | | [1,2,3] [2,1,3] [2,3,1] [1,3,2] - | |
| pc | | 9 10 9 10 9 10 9 10 9 | |

| misterio | | | |
|----------|----|-------------------------|--|
| 11 | i | [1,3,2] | |
| 2 | m | [2,3] [3,2] | |
| 1 | ls | [1,2,3] | |
| 0 | pc | 2 3 4 5 4 5 4 5 4 3 4 5 | |

Volvemos a ejecutar el ins de la linea 4 y retorna [3,1,2]. Se almacena en i1d

Imprime: [1,2,3] , [2,1,3] , [2,3,1], [1,3,2]

```
1 def misterio(ls):
2     if ls:
3         for m in misterio(ls[1:]):
4             for i in ins(ls[0], m):
5                 yield i
6     else:
7         yield []
8
9 for m in misterio([1,2,3]):
10    print m
```

| misterio | | | |
|----------|----|---------|--|
| 9 | i | [3] | |
| 8 | m | [] | |
| 7 | ls | [3] | |
| 6 | pc | 2 3 4 5 | |

| misterio | | | |
|----------|----|-------------|--|
| 10 | i | [2,3] [3,2] | |
| 5 | m | [3] | |
| 4 | ls | [2,3] | |
| 3 | pc | 2 3 4 5 4 5 | |

| GLOBAL | | | |
|----------|--|-----------------------------------|--|
| misterio | | proc | |
| m | | [1,2,3] [2,1,3] [2,3,1] [1,3,2] - | |
| pc | | 9 10 9 10 9 10 9 10 9 | |

| misterio | | | |
|----------|----|---------------------------|--|
| 11 | i | [1,3,2] [3,1,2] | |
| 2 | m | [2,3] [3,2] | |
| 1 | ls | [1,2,3] | |
| 0 | pc | 2 3 4 5 4 5 4 5 4 3 4 5 4 | |

Se retorna por la linea 5 el valor de i11 a m global

Imprime: [1,2,3] , [2,1,3] , [2,3,1], [1,3,2]

```
1 def misterio(ls):
2     if ls:
3         for m in misterio(ls[1:]):
4             for i in ins(ls[0], m):
5                 yield i
6     else:
7         yield []
8
9 for m in misterio([1,2,3]):
10    print m
```

| misterio | | | |
|----------|----|---------|--|
| 9 | i | [3] | |
| 8 | m | [] | |
| 7 | ls | [3] | |
| 6 | pc | 2 3 4 5 | |

| misterio | | | |
|----------|----|-------------|--|
| 10 | i | [2,3] [3,2] | |
| 5 | m | [3] | |
| 4 | ls | [2,3] | |
| 3 | pc | 2 3 4 5 4 5 | |

| GLOBAL | | | |
|----------|--|---|--|
| misterio | | proc | |
| m | | [1,2,3] [2,1,3] [2,3,1] [1,3,2] [3,1,2] | |
| pc | | 9 10 9 10 9 10 9 10 9 | |

| misterio | | | |
|----------|----|-----------------------------|--|
| 11 | i | [1,3,2] [3,1,2] | |
| 2 | m | [2,3] [3,2] | |
| 1 | ls | [1,2,3] | |
| 0 | pc | 2 3 4 5 4 5 4 5 4 3 4 5 4 5 | |

Se imprime m global

Imprime: [1,2,3] , [2,1,3] , [2,3,1],
[1,3,2], [3,1,2]

```
1 def misterio(ls):
2     if ls:
3         for m in misterio(ls[1:]):
4             for i in ins(ls[0], m):
5                 yield i
6     else:
7         yield []
8
9 for m in misterio([1,2,3]):
10    print m
```

| misterio | | | |
|----------|----|---------|--|
| 9 | i | [3] | |
| 8 | m | [] | |
| 7 | ls | [3] | |
| 6 | pc | 2 3 4 5 | |

| misterio | | | |
|----------|----|-------------|--|
| 10 | i | [2,3] [3,2] | |
| 5 | m | [3] | |
| 4 | ls | [2,3] | |
| 3 | pc | 2 3 4 5 4 5 | |

| GLOBAL | | | |
|----------|---|------|--|
| misterio | | proc | |
| m | [1,2,3] [2,1,3] [2,3,1] [1,3,2] [3,1,2] | | |
| pc | 9 10 9 10 9 10 9 10 9 10 | | |

| misterio | | | |
|----------|----|-----------------------------|--|
| 11 | i | [1,3,2] [3,1,2] | |
| 2 | m | [2,3] [3,2] | |
| 1 | ls | [1,2,3] | |
| 0 | pc | 2 3 4 5 4 5 4 5 4 3 4 5 4 5 | |

Volvemos con el misterio de la linea 9

Imprime: [1,2,3] , [2,1,3] , [2,3,1],
 [1,3,2], [3,1,2]

```

1 def misterio(ls):
2     if ls:
3         for m in misterio(ls[1:]):
4             for i in ins(ls[0], m):
5                 yield i
6     else:
7         yield []
8
9 for m in misterio([1,2,3]):
10    print m

```

| misterio | | | |
|----------|----|---------|--|
| 9 | i | [3] | |
| 8 | m | [] | |
| 7 | ls | [3] | |
| 6 | pc | 2 3 4 5 | |

| misterio | | | |
|----------|----|-------------|--|
| 10 | i | [2,3] [3,2] | |
| 5 | m | [3] | |
| 4 | ls | [2,3] | |
| 3 | pc | 2 3 4 5 4 5 | |

| GLOBAL | | | |
|----------|---|--|--|
| misterio | proc | | |
| m | [1,2,3] [2,1,3] [2,3,1] [1,3,2] [3,1,2] - | | |
| pc | 9 10 9 10 9 10 9 10 9 10 9 | | |

| misterio | | | |
|----------|----|-----------------------------|--|
| 11 | i | [1,3,2] [3,1,2] | |
| 2 | m | [2,3] [3,2] | |
| 1 | ls | [1,2,3] | |
| 0 | pc | 2 3 4 5 4 5 4 5 4 3 4 5 4 5 | |

Volvemos con el ins de la linea 4. Retorna [3,2,1] y se guarda en i11

Imprime: [1,2,3], [2,1,3], [2,3,1],
[1,3,2], [3,1,2]

```
1 def misterio(ls):
2     if ls:
3         for m in misterio(ls[1:]):
4             for i in ins(ls[0], m):
5                 yield i
6     else:
7         yield []
8
9 for m in misterio([1,2,3]):
10    print m
```

| misterio | | | |
|----------|----|---------|--|
| 9 | i | [3] | |
| 8 | m | [] | |
| 7 | ls | [3] | |
| 6 | pc | 2 3 4 5 | |

| misterio | | | |
|----------|----|-------------|--|
| 10 | i | [2,3] [3,2] | |
| 5 | m | [3] | |
| 4 | ls | [2,3] | |
| 3 | pc | 2 3 4 5 4 5 | |

| GLOBAL | | | |
|----------|---|------|--|
| misterio | | proc | |
| m | [1,2,3] [2,1,3] [2,3,1] [1,3,2] [3,1,2] - | | |
| pc | 9 10 9 10 9 10 9 10 9 10 9 | | |

| misterio | | | |
|----------|----|-------------------------------|--|
| 11 | i | [1,3,2] [3,1,2] [3,2,1] | |
| 2 | m | [2,3] [3,2] | |
| 1 | ls | [1,2,3] | |
| 0 | pc | 2 3 4 5 4 5 4 5 4 3 4 5 4 5 4 | |

Por la linea 5 se retorna el valor de i11 a m global

Imprime: [1,2,3] , [2,1,3] , [2,3,1],
[1,3,2], [3,1,2]

```
1 def misterio(ls):
2     if ls:
3         for m in misterio(ls[1:]):
4             for i in ins(ls[0], m):
5                 yield i
6     else:
7         yield []
8
9 for m in misterio([1,2,3]):
10    print m
```

| misterio | | | |
|----------|----|---------|--|
| 9 | i | [3] | |
| 8 | m | [] | |
| 7 | ls | [3] | |
| 6 | pc | 2 3 4 5 | |

| misterio | | | |
|----------|----|-------------|--|
| 10 | i | [2,3] [3,2] | |
| 5 | m | [3] | |
| 4 | ls | [2,3] | |
| 3 | pc | 2 3 4 5 4 5 | |

| GLOBAL | | | |
|----------|---|------|--|
| misterio | | proc | |
| m | [1,2,3] [2,1,3] [2,3,1] [1,3,2] [3,1,2] [3,2,1] | | |
| pc | 9 10 9 10 9 10 9 10 9 10 9 | | |

| misterio | | | |
|----------|----|---------------------------------|--|
| 11 | i | [1,3,2] [3,1,2] [3,2,1] | |
| 2 | m | [2,3] [3,2] | |
| 1 | ls | [1,2,3] | |
| 0 | pc | 2 3 4 5 4 5 4 5 4 3 4 5 4 5 4 5 | |

Se imprime el valor de m global

Imprime: [1,2,3] , [2,1,3] , [2,3,1],
[1,3,2], [3,1,2], [3,2,1]

```
1 def misterio(ls):
2     if ls:
3         for m in misterio(ls[1:]):
4             for i in ins(ls[0], m):
5                 yield i
6     else:
7         yield []
8
9 for m in misterio([1,2,3]):
10    print m
```

| misterio | | | |
|----------|----|---------|--|
| 9 | i | [3] | |
| 8 | m | [] | |
| 7 | ls | [3] | |
| 6 | pc | 2 3 4 5 | |

| misterio | | | |
|----------|----|-------------|--|
| 10 | i | [2,3] [3,2] | |
| 5 | m | [3] | |
| 4 | ls | [2,3] | |
| 3 | pc | 2 3 4 5 4 5 | |

| GLOBAL | | | |
|----------|---|------|--|
| misterio | | proc | |
| m | [1,2,3] [2,1,3] [2,3,1] [1,3,2] [3,1,2] [3,2,1] | | |
| pc | 9 10 9 10 9 10 9 10 9 10 9 10 | | |

| misterio | | | |
|----------|----|---------------------------------|--|
| 11 | i | [1,3,2] [3,1,2] [3,2,1] | |
| 2 | m | [2,3] [3,2] | |
| 1 | ls | [1,2,3] | |
| 0 | pc | 2 3 4 5 4 5 4 5 4 3 4 5 4 5 4 5 | |

Volvemos una vez mas con el misterio de la linea 9

Imprime: [1,2,3] , [2,1,3] , [2,3,1],
[1,3,2], [3,1,2], [3,2,1]

```
1 def misterio(ls):
2     if ls:
3         for m in misterio(ls[1:]):
4             for i in ins(ls[0], m):
5                 yield i
6     else:
7         yield []
8
9 for m in misterio([1,2,3]):
10    print m
```

| misterio | | | |
|----------|----|---------|--|
| 9 | i | [3] | |
| 8 | m | [] | |
| 7 | ls | [3] | |
| 6 | pc | 2 3 4 5 | |

| misterio | | | |
|----------|----|-------------|--|
| 10 | i | [2,3] [3,2] | |
| 5 | m | [3] | |
| 4 | ls | [2,3] | |
| 3 | pc | 2 3 4 5 4 5 | |

| GLOBAL | | | |
|----------|---|------|--|
| misterio | | proc | |
| m | [1,2,3] [2,1,3] [2,3,1] [1,3,2] [3,1,2] [3,2,1] - | | |
| pc | 9 10 9 10 9 10 9 10 9 10 9 10 9 | | |

| misterio | | | |
|----------|----|---------------------------------|--|
| 11 | i | [1,3,2] [3,1,2] [3,2,1] | |
| 2 | m | [2,3] [3,2] | |
| 1 | ls | [1,2,3] | |
| 0 | pc | 2 3 4 5 4 5 4 5 4 3 4 5 4 5 4 5 | |

Volvemos con ins. Esta vez no retorna nada ya que termine su ejecucion

Imprime: [1,2,3] , [2,1,3] , [2,3,1],
[1,3,2], [3,1,2], [3,2,1]

```
1 def misterio(ls):
2     if ls:
3         for m in misterio(ls[1:]):
4             for i in ins(ls[0], m):
5                 yield i
6     else:
7         yield []
8
9 for m in misterio([1,2,3]):
10    print m
```

| misterio | | | |
|----------|----|---------|--|
| 9 | i | [3] | |
| 8 | m | [] | |
| 7 | ls | [3] | |
| 6 | pc | 2 3 4 5 | |

| misterio | | | |
|----------|----|-------------|--|
| 10 | i | [2,3] [3,2] | |
| 5 | m | [3] | |
| 4 | ls | [2,3] | |
| 3 | pc | 2 3 4 5 4 5 | |

| GLOBAL | | | |
|----------|---|------|--|
| misterio | | proc | |
| m | [1,2,3] [2,1,3] [2,3,1] [1,3,2] [3,1,2] [3,2,1] | - | |
| pc | 9 10 9 10 9 10 9 10 9 10 9 10 9 | | |

| misterio | | | |
|----------|----|-----------------------------------|--|
| 11 | i | - | |
| 2 | m | [2,3] [3,2] | |
| 1 | ls | [1,2,3] | |
| 0 | pc | 2 3 4 5 4 5 4 5 4 3 4 5 4 5 4 5 4 | |

Retomamos misterio linea 3.

Imprime: [1,2,3] , [2,1,3] , [2,3,1],
 [1,3,2], [3,1,2], [3,2,1]

```

1 def misterio(ls):
2     if ls:
3         for m in misterio(ls[1:]):
4             for i in ins(ls[0], m):
5                 yield i
6     else:
7         yield []
8
9 for m in misterio([1,2,3]):
10    print m

```

| misterio | | | |
|----------|----|---------|--|
| 9 | i | [3] | |
| 8 | m | [] | |
| 7 | ls | [3] | |
| 6 | pc | 2 3 4 5 | |

| misterio | | | |
|----------|----|-------------|--|
| 10 | i | [2,3] [3,2] | |
| 5 | m | [3] | |
| 4 | ls | [2,3] | |
| 3 | pc | 2 3 4 5 4 5 | |

| GLOBAL | | | |
|----------|---|------|--|
| misterio | | proc | |
| m | [1,2,3] [2,1,3] [2,3,1] [1,3,2] [3,1,2] [3,2,1] | - | |
| pc | 9 10 9 10 9 10 9 10 9 10 9 10 9 | | |

| misterio | | | |
|----------|----|-------------------------|---|
| 11 | i | - | |
| 2 | m | [2,3] [3,2] | - |
| 1 | ls | [1,2,3] | |
| 0 | pc | 4 5 4 3 4 5 4 5 4 5 4 3 | |

Llamamos a ins. Pero no termino su ejecucion.

Imprime: [1,2,3] , [2,1,3] , [2,3,1],
 [1,3,2], [3,1,2], [3,2,1]

```

1 def misterio(ls):
2     if ls:
3         for m in misterio(ls[1:]):
4             for i in ins(ls[0], m):
5                 yield i
6     else:
7         yield []
8
9 for m in misterio([1,2,3]):
10    print m

```

| misterio | | | |
|----------|----|---------|--|
| 9 | i | [3] | |
| 8 | m | [] | |
| 7 | ls | [3] | |
| 6 | pc | 2 3 4 5 | |

| misterio | | | |
|----------|----|---------------|--|
| 10 | i | - | |
| 5 | m | [3] | |
| 4 | ls | [2,3] | |
| 3 | pc | 2 3 4 5 4 5 4 | |

| GLOBAL | | | |
|----------|---|--|--|
| misterio | proc | | |
| m | [1,2,3] [2,1,3] [2,3,1] [1,3,2] [3,1,2] [3,2,1] - | | |
| pc | 9 10 9 10 9 10 9 10 9 10 9 10 9 | | |

| misterio | | | |
|----------|----|-------------------------|--|
| 11 | i | - | |
| 2 | m | [2,3] [3,2] - | |
| 1 | ls | [1,2,3] | |
| 0 | pc | 4 5 4 3 4 5 4 5 4 5 4 3 | |

Procedemos a llamar al misterio de la linea 3

Imprime: [1,2,3] , [2,1,3] , [2,3,1],
 [1,3,2], [3,1,2], [3,2,1]

```

1 def misterio(ls):
2     if ls:
3         for m in misterio(ls[1:]):
4             for i in ins(ls[0], m):
5                 yield i
6     else:
7         yield []
8
9 for m in misterio([1,2,3]):
10    print m

```

| misterio | | | |
|----------|----|---------|--|
| 9 | i | [3] | |
| 8 | m | [] | |
| 7 | ls | [3] | |
| 6 | pc | 2 3 4 5 | |

| misterio | | | |
|----------|----|-----------------|--|
| 10 | i | - | |
| 5 | m | [3] - | |
| 4 | ls | [2,3] | |
| 3 | pc | 2 3 4 5 4 5 4 3 | |

| GLOBAL | | | |
|----------|---|------|--|
| misterio | | proc | |
| m | [1,2,3] [2,1,3] [2,3,1] [1,3,2] [3,1,2] [3,2,1] | - | |
| pc | 9 10 9 10 9 10 9 10 9 10 9 10 9 | | |

| misterio | | | |
|----------|----|-------------------------|--|
| 11 | i | - | |
| 2 | m | [2,3] [3,2] - | |
| 1 | ls | [1,2,3] | |
| 0 | pc | 4 5 4 3 4 5 4 5 4 5 4 3 | |

Llamamos a ins. Linea 4 Pero Tambien termino su ejecucion

Imprime: [1,2,3] , [2,1,3] , [2,3,1],
 [1,3,2], [3,1,2], [3,2,1]

```

1 def misterio(ls):
2     if ls:
3         for m in misterio(ls[1:]):
4             for i in ins(ls[0], m):
5                 yield i
6     else:
7         yield []
8
9 for m in misterio([1,2,3]):
10    print m

```

| misterio | | | |
|----------|----|-----|-----------|
| 9 | i | ls | pc |
| | - | [3] | 2 3 4 5 4 |
| 8 | m | [] | |
| 7 | ls | | |
| 6 | pc | | |

| misterio | | | |
|----------|----|-----------------|----|
| 10 | i | ls | pc |
| | - | | |
| 5 | m | [3] - | |
| 4 | ls | [2,3] | |
| 3 | pc | 2 3 4 5 4 5 4 3 | |

| GLOBAL | | | |
|----------|---|---|--|
| misterio | proc | | |
| m | [1,2,3] [2,1,3] [2,3,1] [1,3,2] [3,1,2] [3,2,1] | - | |
| pc | 9 10 9 10 9 10 9 10 9 10 9 10 9 | | |

| misterio | | | |
|----------|----|-------------------------|----|
| 11 | i | ls | pc |
| | - | | |
| 2 | m | [2,3] [3,2] - | |
| 1 | ls | [1,2,3] | |
| 0 | pc | 4 5 4 3 4 5 4 5 4 5 4 3 | |

Buscamos apoyo en misterio linea 3 pero Tambien termino su ejecucion

Imprime: [1,2,3] , [2,1,3] , [2,3,1],
[1,3,2], [3,1,2], [3,2,1]

```
1 def misterio(ls):
2     if ls:
3         for m in misterio(ls[1:]):
4             for i in ins(ls[0], m):
5                 yield i
6     else:
7         yield []
8
9 for m in misterio([1,2,3]):
10    print m
```

| misterio | | | |
|----------|----|-------------|--|
| 9 | i | - | |
| 8 | m | [] | |
| 7 | ls | [3] | |
| 6 | pc | 2 3 4 5 4 3 | |

| misterio | | | |
|----------|----|-----------------|--|
| 10 | i | - | |
| 5 | m | [3] - | |
| 4 | ls | [2,3] | |
| 3 | pc | 2 3 4 5 4 5 4 3 | |

| GLOBAL | | | |
|----------|---|------|--|
| misterio | | proc | |
| m | [1,2,3] [2,1,3] [2,3,1] [1,3,2] [3,1,2] [3,2,1] | - | |
| pc | 9 10 9 10 9 10 9 10 9 10 9 10 9 | | |

| misterio | | | |
|----------|----|-------------------------|--|
| 11 | i | - | |
| 2 | m | [2,3] [3,2] - | |
| 1 | ls | [1,2,3] | |
| 0 | pc | 4 5 4 3 4 5 4 5 4 5 4 3 | |

Como ya se termino la ejecucion de misterio linea 3. Pues se acabo la ejecucion de misterio

Imprime: [1,2,3] , [2,1,3] , [2,3,1],
[1,3,2], [3,1,2], [3,2,1]

```

1 def misterio(ls):
2     if ls:
3         for m in misterio(ls[1:]):
4             for i in ins(ls[0], m):
5                 yield i
6     else:
7         yield []
8
9 for m in misterio([1,2,3]):
10    print m

```

| misterio | | |
|----------|---|---------|
| 9 | m | |
| 8 | | |
| 7 | | |
| 6 | | 2 3 4 5 |

| misterio | | |
|----------|----|-----------------|
| 10 | i | - |
| 5 | m | [3] - |
| 4 | ls | [2,3] |
| 3 | pc | 2 3 4 5 4 5 4 3 |

| GLOBAL | | |
|----------|---|---|
| misterio | proc | |
| m | [1,2,3] [2,1,3] [2,3,1] [1,3,2] [3,1,2] [3,2,1] | - |
| pc | 9 10 9 10 9 10 9 10 9 10 9 10 9 | |

| misterio | | |
|----------|----|---------------------|
| 11 | i | - |
| 2 | m | [2,3] [3,2] - |
| 1 | ls | [1,2,3] |
| 0 | pc | 4 5 4 3 4 5 4 5 4 3 |

Subsecuentemente, al terminarse el misterio. m5 no consigue valor y termina el misterio

Imprime: [1,2,3] , [2,1,3] , [2,3,1],
[1,3,2], [3,1,2], [3,2,1]

```
1 def misterio(ls):
2     if ls:
3         for m in misterio(ls[1:]):
4             for i in ins(ls[0], m):
5                 yield i
6     else:
7         yield []
8
9 for m in misterio([1,2,3]):
10    print m
```

| GLOBAL | |
|----------|---|
| misterio | proc |
| m | [1,2,3] [2,1,3] [2,3,1] [1,3,2] [3,1,2] [3,2,1] - |
| pc | 9 10 9 10 9 10 9 10 9 10 9 10 9 |

| misterio | |
|----------|----|
| 10 | m |
| 5 | m |
| 4 | m |
| 3 | pc |

| misterio | |
|----------|----|
| 9 | |
| 8 | m |
| 7 | m |
| 6 | pc |

| misterio | |
|----------|----|
| 11 | i |
| 2 | m |
| 1 | ls |
| 0 | pc |

Y como el misterio del programa principal termino. Tambien el programa principal termina

Imprime: [1,2,3] , [2,1,3] , [2,3,1],
[1,3,2], [3,1,2], [3,2,1]

```
1 def misterio(ls):
2     if ls:
3         for m in misterio(ls[1:]):
4             for i in ins(ls[0], m):
5                 yield i
6     else:
7         yield []
8
9 for m in misterio([1,2,3]):
10    print m
```

| GLOBAL | |
|----------|---|
| misterio | proc |
| m | [1,2,3] [2,1,3] [2,3,1] [1,3,2] [3,1,2] [3,2,1] - |
| pc | 9 10 9 10 9 10 9 10 9 10 9 10 9 |



| | |
|----|----------|
| 10 | misterio |
| 5 | misterio |
| 4 | misterio |
| 3 | misterio |

pc 2 3 4 5 6 7 8 9 10 11



| | |
|---|----------|
| 9 | misterio |
| 8 | m |
| 7 | m |
| 6 | m |

pc 2 3 4 5



| | |
|----|----------|
| 11 | misterio |
| 2 | m |
| 1 | m |
| 0 | m |

pc 4 5 4 3 4 5 4 5 4

Siguiendo con la cadena. El ultimo misterio termina su ejecucion tambien.

Imprime: [1,2,3] , [2,1,3] , [2,3,1],
[1,3,2], [3,1,2], [3,2,1]

```
1 def misterio(ls):
2     if ls:
3         for m in misterio(ls[1:]):
4             for i in ins(ls[0], m):
5                 yield i
6     else:
7         yield []
8
9 for m in misterio([1,2,3]):
10    print m
```

| GLOBAL | |
|----------|---|
| misterio | proc |
| m | [1,2,3] [2,1,3] [2,3,1] [1,3,2] [3,1,2] [3,2,1] - |
| pc | 9 10 9 10 9 10 9 10 9 10 9 10 9 |

| | | |
|----|----------|---|
| 10 | misterio | 9 |
| 5 | m | m |
| 4 | 1 | 1 |
| 3 | 2 | 2 |
| 2 | 3 | 3 |
| 1 | 4 | 4 |
| 0 | 5 | 5 |
| | 6 | 6 |
| | 7 | 7 |
| | 8 | 8 |

| | | |
|---|----------|---|
| 9 | misterio | 8 |
| 8 | m | m |
| 7 | 1 | 1 |
| 6 | 2 | 2 |
| 5 | 3 | 3 |
| 4 | 4 | 4 |
| 3 | 5 | 5 |
| 2 | 6 | 6 |
| 1 | 7 | 7 |
| 0 | 8 | 8 |

| | | |
|----|----------|----|
| 11 | misterio | 11 |
| 2 | m | m |
| 1 | 1 | 1 |
| 0 | 2 | 2 |
| | 3 | 3 |
| | 4 | 4 |
| | 5 | 5 |
| | 6 | 6 |
| | 7 | 7 |
| | 8 | 8 |
| | 9 | 9 |
| | 10 | 10 |
| | 11 | 11 |

Este algoritmo imprimió

```
1 def misterio(ls):
2     if ls:
3         for m in misterio(ls[1:]):
4             for i in ins(ls[0], m):
5                 yield i
6     else:
7         yield []
8
9 for m in misterio([1,2,3]):
10    print m
```

Este algoritmo imprime las permutaciones de una lista.

Imprime:

[1,2,3]

[2,1,3]

[2,3,1]

[1,3,2]

[3,1,2]

[3,2,1]

```
1 def misterio(ls):
2     if ls:
3         for m in misterio(ls[1:]):
4             for i in ins(ls[0], m):
5                 yield i
6     else:
7         yield []
8
9 for m in misterio([1,2,3]):
10    print m
```

Misterio lo que hace es descomponer una lista en sub listas de 1 menos de tamaño. Ejemplo [1,2,3], [2,3], [3] y []. Luego de esto, le pasa a la función ins. El primer elemento de la lista local y de Segundo elemento una lista que resulta de una llamada recursiva misterio. Tomemos [2,3] y [3] como ejemplo. Ins(2, [3]) porque así? Porque la llamada de [2,3] de misterio genera un subconjunto menor que es [3]. Y 2 que es el primer elemento de la lista local y [3] que es el resultado de la llamada recursive.

```
1 def misterio(ls):
2     if ls:
3         for m in misterio(ls[1:]):
4             for i in ins(ls[0], m):
5                 yield i
6     else:
7         yield []
8
9 for m in misterio([1,2,3]):
10    print m
```

La función `ins` como ya sabemos dado 1 elemento no da todos los arreglos en donde el elemento e va ocupar de 1ra a la ncima posición. Ej: $\text{ins}(4,[1,2,3]) = ([4,1,2,3] ; [1,4,2,3]; [1,2,4,3] ; [1,2,3,4])$

Lo que hace `misterio` con esto es aprovecharlo pasando dos sub arreglos del lista inicial para que así vaya poco a poco constuyendo las permutaciones.