

# Examen Final parte 1

Item 1/30

✓ Correct

¿Cuál es el resultado del siguiente fragmento de código?

```
lst = [1, 2]
for v in range(2):
    lst.insert(-1, lst[v])
print(lst)
```

- ☐ [1, 2, 2, 2]
- ☐ [2, 1, 1, 2]
- ☒ [1, 1, 1, 2]
- ☐ [1, 2, 1, 2]

Item 2/30

✓ Correct

El significado de un *argumento posicional* está determinado por:

- ☐ el nombre del argumento especificado junto con su valor
- ☐ su conexión con variables existentes
- ☒ su posición dentro de la lista de argumentos
- ☐ su valor

Item 3/30

✓ Correct

¿Cuál de los siguientes enunciados es verdadero?

```
nums = [1, 2, 3]
vals = nums
```

- ☒ `nums` y `vals` son diferentes nombres de la misma lista
- ☐ `vals` es más largo que `nums`
- ☐ `nums` es más largo que `vals`
- ☐ `nums` y `vals` son listas diferentes

Item 4/30

✓ Correct

Un operador capaz de verificar si dos valores no son iguales es:

- ☐ `not ==`
- ☐ `!=`
- ☐ `<>`
- ☒ `!=`

Item 5/30

✓ Correct

El siguiente fragmento de código:

```
def func1(a):
    return None

def func2(a):
    return func1(a) * func1(a)

print(func2(2))
```

- ☐ dará como salida `4`
- ☐ dará como salida `16`
- ☒ provocará un error de tiempo de ejecución
- ☐ dará como salida `2`

## Item 6/30

✔ Correct

El resultado de la siguiente división:

`1 // 2`☐ es igual a `0.0`☐ es igual a `0.5`☒ es igual a `0`☐ no se puede predecir

## Item 7/30

✔ Correct

El siguiente fragmento de código:

```
def func(a, b):  
    return b ** a  
  
print(func(b=2, 2))
```

☒ es erróneo☐ dará como salida `4`☐ dará como salida `None`☐ dará como salida `2`

## Item 8/30

✔ Correct

¿Qué valor se asignará a la variable `x`?

```
z = 0  
y = 10  
x = y < z and z > y or y > z and z < y
```

☒ `True`☐ `False`☐ `0`☐ `1`

## Item 9/30

✔ Correct

Uno de los siguientes nombres de variables es ilegal - ¿cuál?☐ `in`☐ `In`☒ `in`☐ `IN`

## Item 10/30

✔ Correct

¿Cuál es el resultado del siguiente fragmento de código?

```
list = [x * x for x in range(5)]  
def fun(list):  
    del list[list[2]]  
    return list  
  
print(fun(list))
```

☐ `[1, 4, 9, 16]`☐ `[0, 1, 9, 16]`☒ `[0, 1, 4, 9]`☐ `[0, 1, 4, 16]`

Item 11/30

✔ Correct

¿Cuál es el resultado del siguiente código?

```
x = 1
y = 2
x, y, z = x, x, y
z, y, z = x, y, z
print(x, y, z)
```

☐ 1 2 1

☒ 1 1 2

☐ 2 1 2

☐ 1 2 2

Item 12/30

✔ Correct

¿Cuál será la salida del siguiente fragmento de código?

```
a = 1
b = 0
a = a ^ b
b = a ^ b
a = a ^ b
print(a, b)
```

☐ 1 0

☒ 0 1

☐ 0 0

☐ 1 1

Item 13/30

✔ Correct

¿Cuál es el resultado del siguiente fragmento de código?

```
def fun(x):
    if x % 2 == 0:
        return 1
    else:
        return 2
print(fun(fun(2)))
```

☐ el código provocará un error de tiempo de ejecución

☐ 2None

☐ 1

☒ 2

Item 14/30

✔ Correct

Observa el fragmento de código y elige la sentencia verdadera:

```
numa = [1, 2, 3]
vals = numa
del vals[1]
```

☐ el código provocará un error de tiempo de ejecución

☐ vals es más largo que numa

☐ numa y vals son diferentes listas

☒ numa y vals son diferentes nombres de la misma lista

☐ numa es más largo que vals

Item 15/30

✔ Correct

¿Cuál es la salida del siguiente código si el usuario ingresa dos líneas que contienen 3 y 2 respectivamente?

```
x = int(input())
y = int(input())
x = x % y
x = x % y
y = y % x
print(y)
```

☐ 1

☐ 2

☐ 3

☒ 0

## Item 16/30

✔ Correct

¿Cuál es la salida del siguiente código si el usuario ingresa dos líneas que contienen 3 y 6 respectivamente?

```
y = input()
x = input()
print(x + y)
```

- ☐ 36
- ☐ 3
- ☐ 6
- ☒ 63

## Item 17/30

✔ Correct

¿Cuál es el resultado del siguiente código?

```
print("a", "b", "c", sep="sep")
```

- ☐ abc
- ☐ a b c
- ☒ asepbsepc
- ☐ asepbsepcsepc

## Item 18/30

✔ Correct

¿Cuál es el resultado del siguiente código?

```
x = 1 // 5 + 1 / 5
print(x)
```

- ☒ 0.2
- ☐ 0
- ☐ 0.4
- ☐ 0.0

## Item 19/30

✔ Correct

Suponiendo que `tuple` es una tupla creada correctamente, el hecho de que las tuplas sean inmutables significa que la siguiente instrucción:

```
tuple[1] = tuple[1] + tuple[0]
```

- ☐ es completamente correcto
- ☐ puede ser ilegal si la tupla contiene cadenas
- ☐ se puede ejecutar si y solo si la tupla contiene al menos dos elementos
- ☒ es ilegal

## Item 20/30

✔ Correct

¿Cuál es la salida del siguiente código si el usuario ingresa dos líneas que contienen 2 y 4 respectivamente?

```
x = float(input())
y = float(input())
print(y ** (1 / x))
```

- ☒ 2.0
- ☐ 0.0
- ☐ 4.0
- ☐ 1.0

Item 21/30

✔ Correct

¿Cuál es el resultado del siguiente fragmento de código?

```
dot = { 'uno':'dos', 'tres':'uno', 'dos':'tres' }  
v = dot['tres']  
  
for k in range(len(dot)):  
    v = dot[v]  
  
print(v)
```

☐ dos☐ ('uno', 'dos', 'tres')☐ tres☒ uno

Item 22/30

✔ Correct

¿Cuántos elementos contiene la lista `lst`?

```
lst = [i for i in range(-1, -2)]
```

☐ uno☒ cero☐ tres☐ dos

Item 23/30

✔ Correct

¿Cuál de las siguientes líneas incorrectamente invoca la función definida como:

```
def fun(a, b, c=0):
```

☐ `fun(a=1, b=0, c=0)`☒ `fun(b=1)`☐ `fun(0, 1, 2)`☐ `fun(a=0, b=0)`

Item 24/30

✔ Correct

¿Cuál es el resultado del siguiente fragmento de código?

```
def fun(x, y):  
    if x == y:  
        return x  
    else:  
        return fun(x, y-1)  
  
print(fun(0, 3))
```

☐ 1☐ 2☒ 0☐ el fragmento de código provocará un error de tiempo de ejecución

Item 25/30

✔ Correct

¿Cuántos asteriscos ( `*` ) imprimirá el siguiente fragmento de código en la consola?

```
i = 0  
while i < i + 2 :  
    i += 1  
    print("***")  
else:  
    print("***")
```

☐ cero☐ uno☐ dos☒ el fragmento entrará en un bucle infinito, imprimiendo un asterisco por línea

## Item 26/30

✔ Correct

¿Cuál es el resultado del siguiente fragmento de código?

```
tup = (1, 2, 4, 8)
tup = tup[-2:-1]
tup = tup[-1]
print(tup)
```

☐ {4}☐ {4,}☒ 4☐ 44

## Item 27/30

✔ Correct

¿Cuál es el resultado del siguiente fragmento de código?

```
dd = { "1":"0", "0":"1" }
for x in dd.values():
    print(x, end="")
```

☐ 1 0☐ 0 0☐ 0 1☒ el código es erróneo (el objeto `dict` no contiene el método `vals()`)

## Item 28/30

✔ Correct

¿Cuál es el resultado del siguiente fragmento de código?

```
dot = {}
dot['1'] = (1, 2)
dot['2'] = (2, 1)

for x in dot.keys():
    print(dot[x][1],end="")
```

☐ (2,1)☒ 21☐ (1,2)☐ 12

## Item 29/30

✔ Correct

¿Cuál es el resultado del siguiente fragmento de código?

```
def fun(inp=2, out=3):
    return inp * out
print(fun(out=2))
```

☐ 2☐ 6☐ el fragmento de código es erróneo☒ 4

## Item 30/30

✔ Correct

¿Cuántos ( # ) imprimirá el siguiente fragmento de código en la consola?

```
lst = [[x for x in range(3)] for y in range(3)]

for x in range(3):
    for o in range(3):
        if lst[x][o] % 2 != 0:
            print("#")
```

☐ cero☐ nueve☐ seis☒ tres

