

Pasos ejecutados para el ejercicio de filter:

```
λ py filter.py
> c:\tarea\filter.py(10)<module>()
-> def es_primo(n):
(Pdb) next
> c:\tarea\filter.py(33)<module>()
-> lista = [3, 4, 8, 5, 5, 22, 13]
(Pdb) break 27
Breakpoint 1 at c:\tarea\filter.py:27
(Pdb) continue
> c:\tarea\filter.py(27)es_primo()
-> for k in range(2, n):
(Pdb) p n
3
(Pdb) next
> c:\tarea\filter.py(28)es_primo()
-> if n % k == 0:
(Pdb) p k
2
(Pdb) next
> c:\tarea\filter.py(27)es_primo()
-> for k in range(2, n):
(Pdb) p n
3
(Pdb) p k
2
(Pdb) next
> c:\tarea\filter.py(30)es_primo()
-> return True
(Pdb) next
--Return--
> c:\tarea\filter.py(30)es_primo()->True
-> return True
(Pdb) next
--Call--
> c:\tarea\filter.py(10)es_primo()
-> def es_primo(n):
```

Continúa en la siguiente página.

```

> for k in range(2, n):
(Pdb) next
> c:\tarea\filter.py(28)es_primo()
-> if n % k == 0:
(Pdb) next
> c:\tarea\filter.py(27)es_primo()
-> for k in range(2, n):
(Pdb) next
> c:\tarea\filter.py(28)es_primo()
-> if n % k == 0:
(Pdb) next
> c:\tarea\filter.py(27)es_primo()
-> for k in range(2, n):
(Pdb) next
> c:\tarea\filter.py(28)es_primo()
-> if n % k == 0:
(Pdb) next
> c:\tarea\filter.py(27)es_primo()
-> for k in range(2, n):
(Pdb) next
> c:\tarea\filter.py(28)es_primo()
-> if n % k == 0:
(Pdb) next
> c:\tarea\filter.py(27)es_primo()
-> for k in range(2, n):
(Pdb) next
> c:\tarea\filter.py(30)es_primo()
-> return True
(Pdb) p n
13
(Pdb) next
--Return--
> c:\tarea\filter.py(30)es_primo()->True
-> return True
(Pdb) next
[3, 5, 5, 13]
--Return--
> c:\tarea\filter.py(35)<module>()->None
-> print(list(filter(es_primo, lista)))
(Pdb) continue

```

Conclusión, se recorre perfectamente la función es_primo con los valores que se le van pasando, definitivamente me gusta más depurar con vscode.