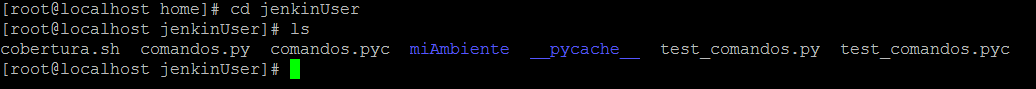
**Pruebas con Pytest:**

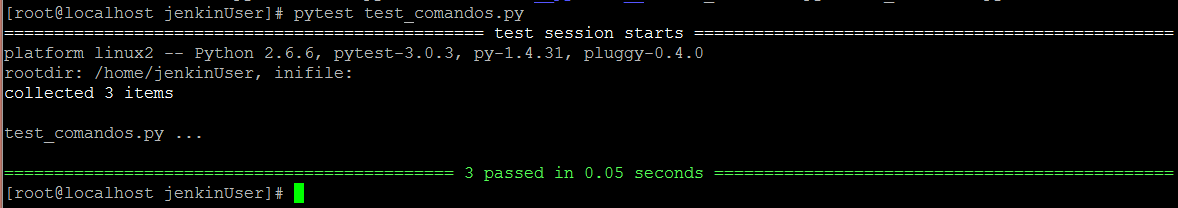
Las estrategias, para analizar la eficacia de cada algoritmo, están descritas en el archivo **README.pdf**

**Prueba 1: test\_Agregar ().**

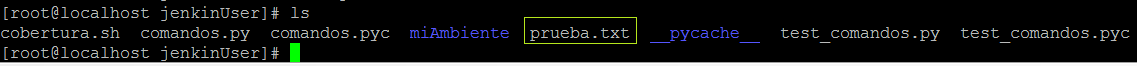
**Escenario I:** No existe un archivo llamado **“prueba.txt”** dentro de la carpeta.



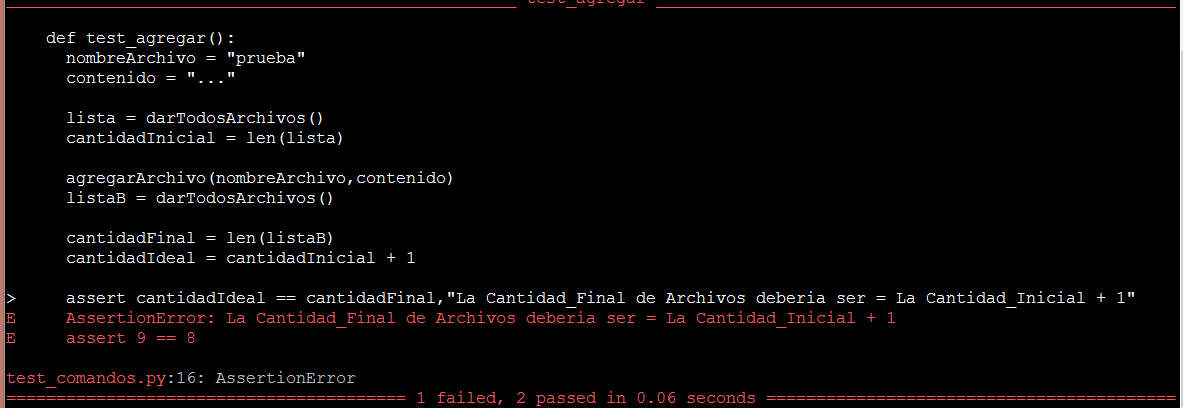
Las 3 pruebas, incluyendo el test\_agregar () sale exitoso



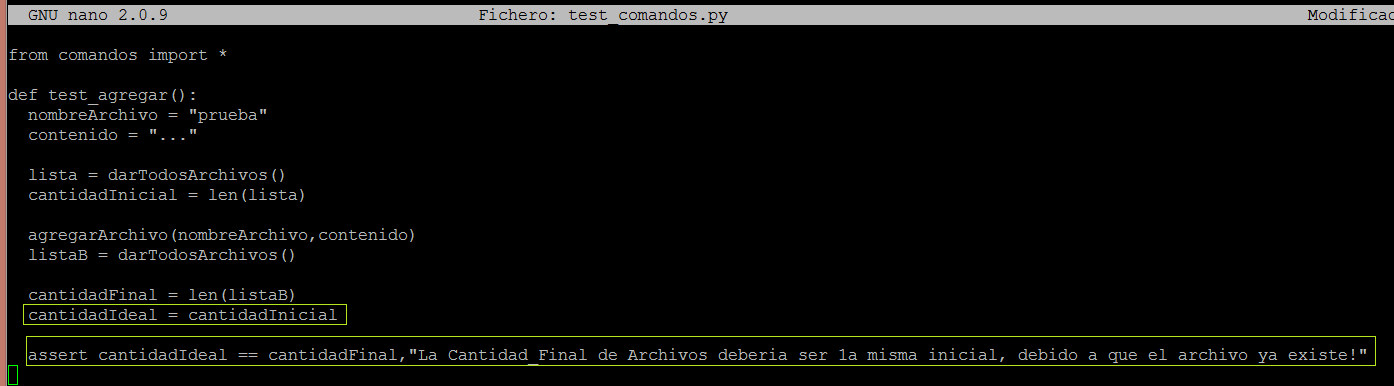
**Escenario II:** Existe un archivo llamado **“prueba.txt”**, dentro de la carpeta.



En este caso la prueba falla porque el algoritmo de test\_Agregar () solo tiene en cuenta el caso cuando **“prueba.txt”** no existe, para simplificar el código.

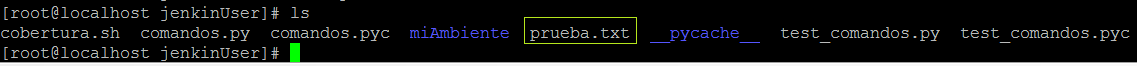


Para tener en cuenta este 2do caso, es necesario modificar el código de la sgte manera:

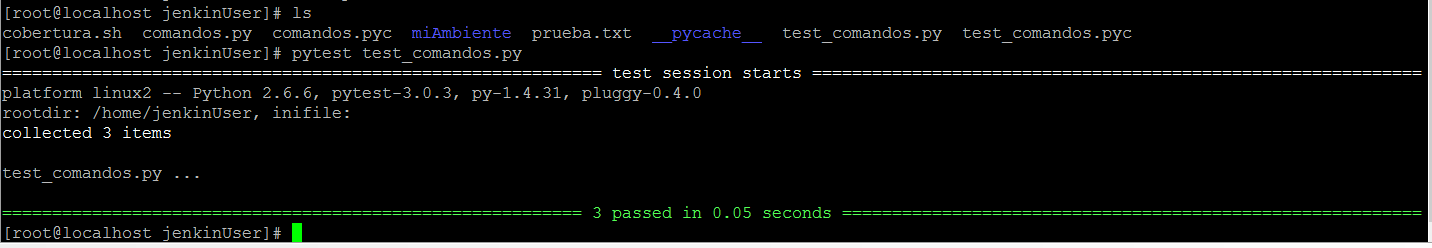


Aquí la variable, que cuenta la cantidad de Archivos, ya no se incrementa, porque no se

creó un archivo nuevo, debido a que ya existía.

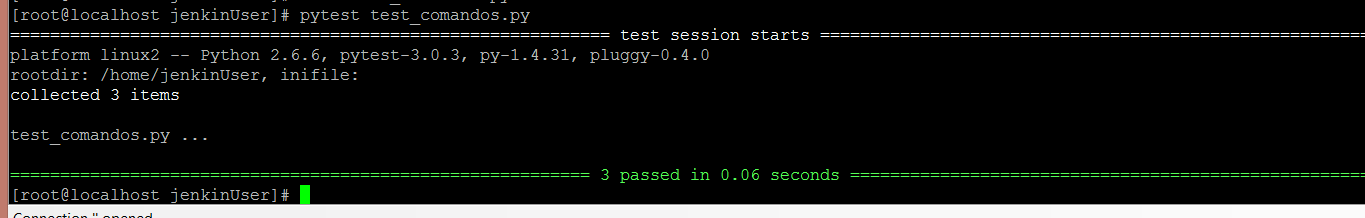


Ahora la prueba si sale exitosa, para el escenario 2



**Prueba 2: test\_Borrar ()**

**Escenario** I: Existe un archivo llamado **“prueba.txt.”,** dentro de la carpeta.



Tengamos en cuenta que los algoritmos de prueba se ejecutan en el sgte orden:

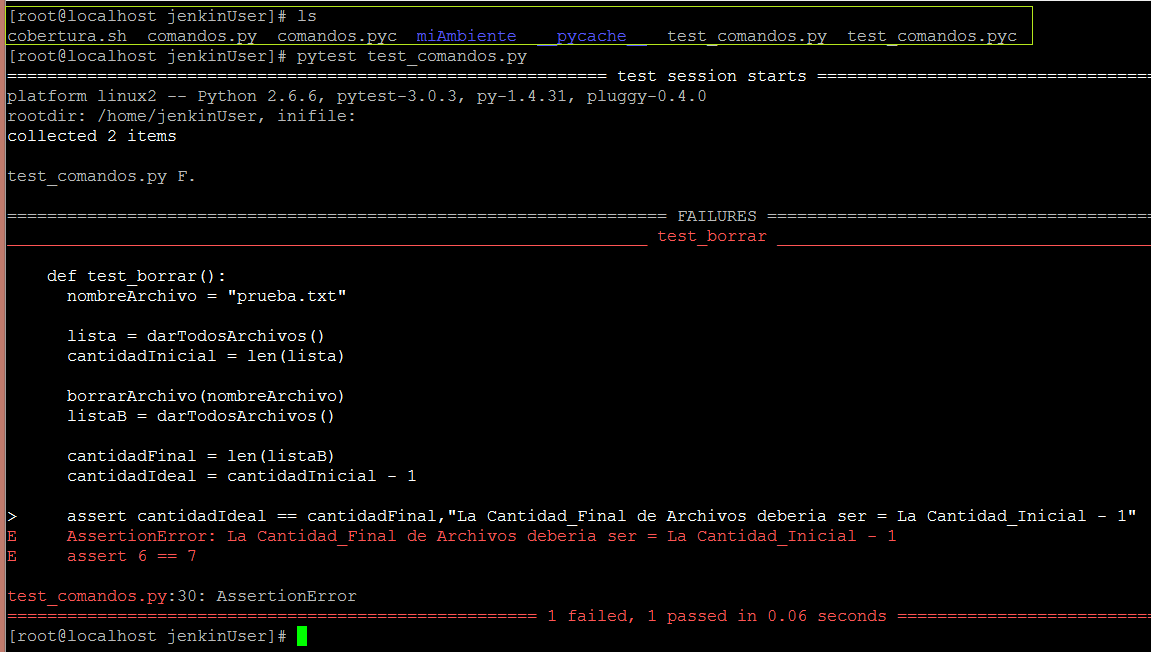
1. test\_agregar ()

2. test\_borrar ()

3.test\_darTodos ().

Este algoritmo (borrar) pasa exitosamente la prueba, debido a que test\_agregar () genera primero el archivo (por el orden de ejecución, descrito arriba) y, cuando el algoritmo test\_borrar se va a efectuar, este archivo ya está dentro de la carpeta, dando lugar al escenario I.

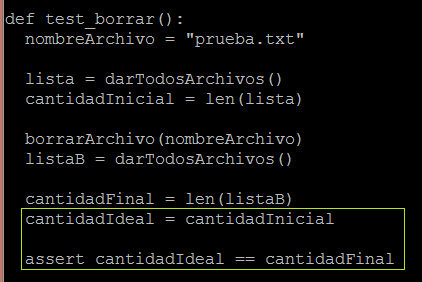
**Escenario II:** No existe un archivo llamado “prueba.txt.” dentro de la carpeta. Se hizo una modificación al test\_comandos.py para que el algoritmo test\_agregar () no generara el archivo “prueba.txt”.



Para este caso, es normal que la prueba falle, porque la estrategia asume que el archivo ya existe y que fue eliminado con éxito de la carpeta (por eso la variable contadora disminuye en 1). Como simplificación de código de prueba, no se incluye el escenario donde el archivo no existe.

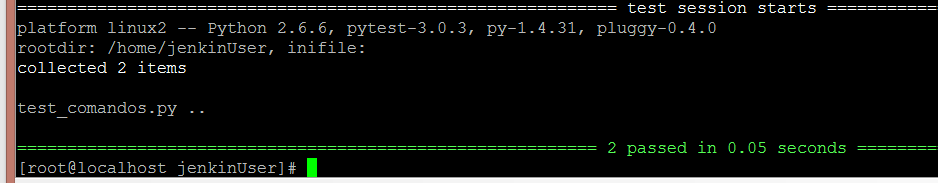
Para que el algoritmo apruebe este escenario, se hace algo análogo al caso del test\_agregar (): La variable contadora ya no se disminuye en 1, sino que queda igual, debido a que no se borró nada (y por ende la cantidad de archivos en la carpeta quedó exacta).

De lo anterior resulta:



La imagen anterior muestra el algoritmo test\_borrar () modificado, para que acepte este 2do escenario.

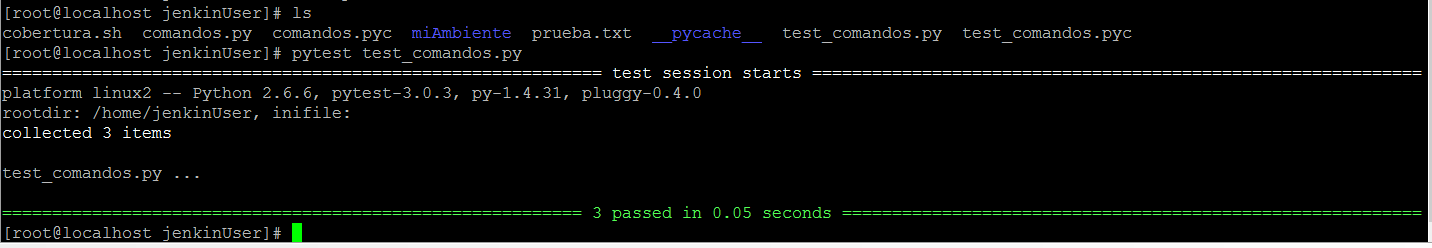
La imagen sgte muestra que la prueba fue exitosa (recordemos que solo se ejecuta el test\_borrar () y el test\_darTodos () porque el test\_agregar () no se ejecutó en este caso, para no crear el archivo **“pruebas.txt”**)



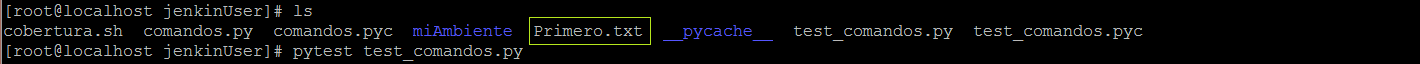
**Prueba 3: test\_darTodos()**

**Escenario I:** Ninguno de los sgtes archivos existe, dentro de la carpeta: “primero.txt”, “segundo.txt”, “Tercero.txt”, “Cuarto.txt”.





La prueba es exitosa, sin ningún problema

**Escenario II:** Alguno de los archivos anteriores existe. 

Ocurre lo mismo que los dos algoritmos anteriores (agregar y borrar). Si alguno ya existe dentro de la carpeta, la prueba falla porque no tiene en cuenta el caso donde alguno de los archivos exista.

