INFORME EJ1 - Área de polígonos

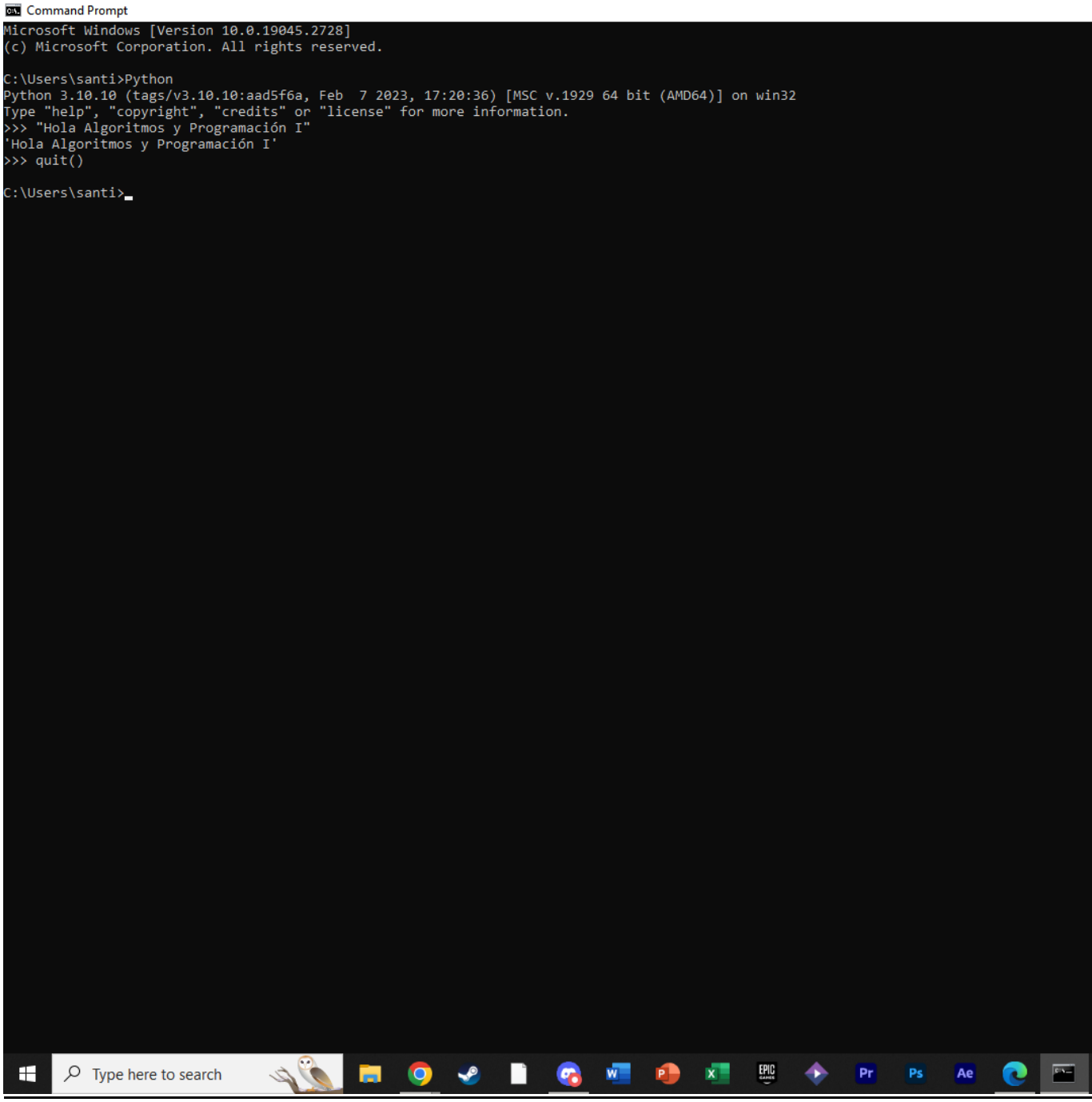
ALGORITMOS Y PROGRAMACIÓN I

CURSO 04-ESSAYA, PRÁCTICA GRACE

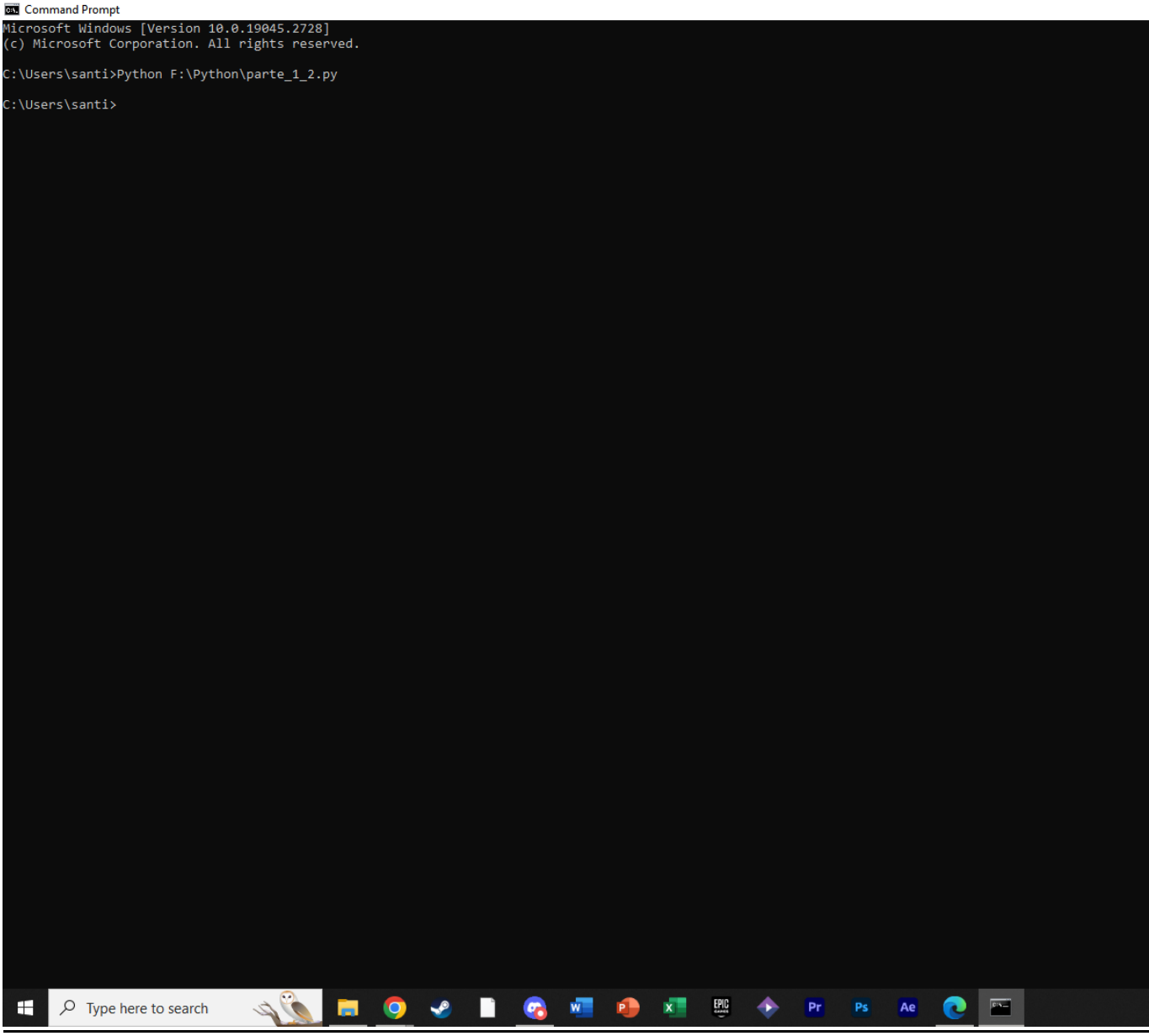
ALUMNO: ROJO SANTIAGO 110700

AYUDANTE A CARGO: IVÁN BOTOSHANSKY

PARTE 1.1

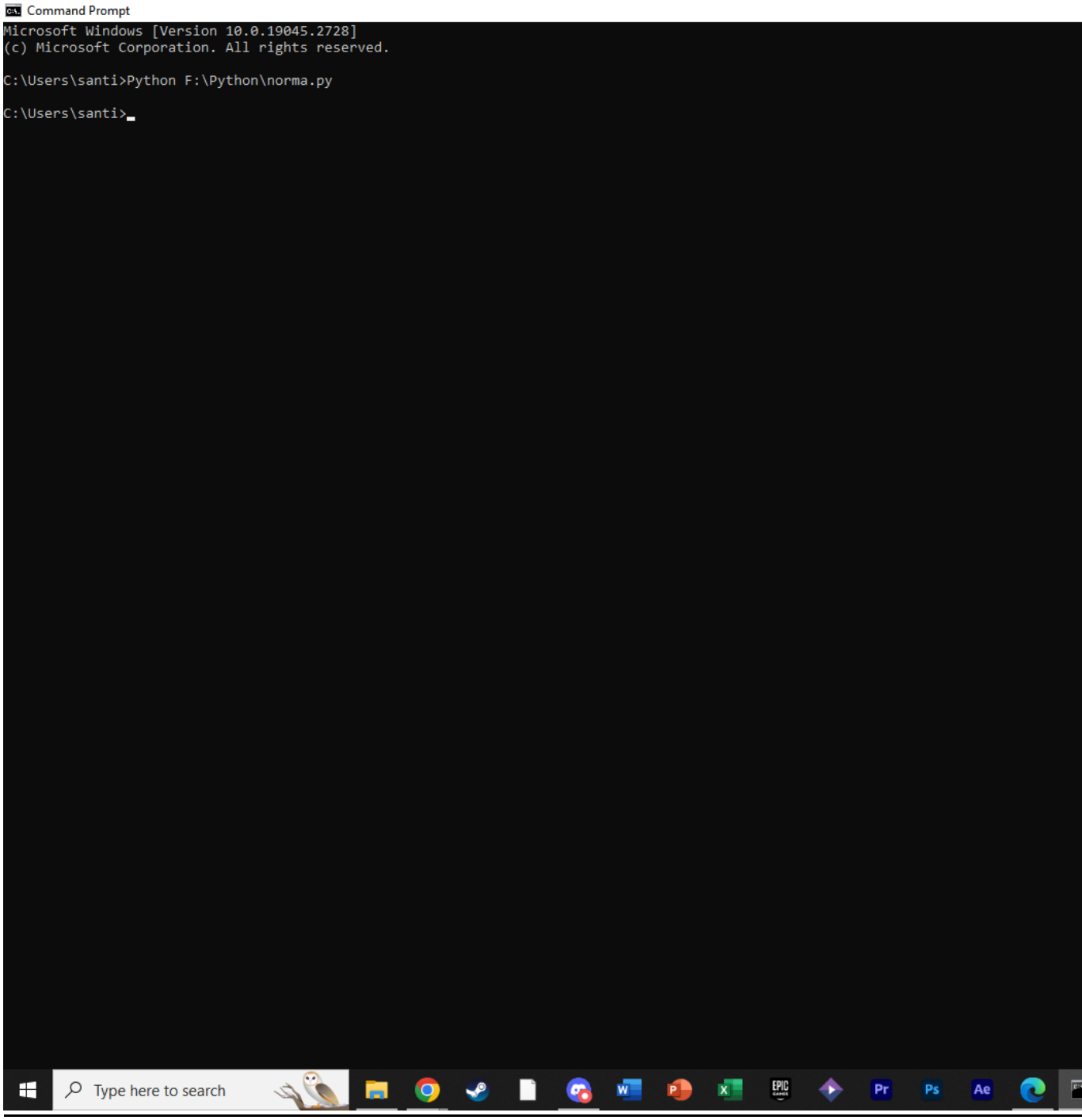

Command Prompt
Microsoft Windows [Version 10.0.19045.2728]
(c) Microsoft Corporation. All rights reserved.
C:\Users\santi>Python
Python 3.10.10 (tags/v3.10.10:aad5f6a, Feb 7 2023, 17:20:36) [MSC v.1929 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license" for more information.
>>> "Hola Algoritmos y Programación I"
'Hola Algoritmos y Programación I'
>>> quit()
C:\Users\santi>
Type here to search

PARTE 1.2

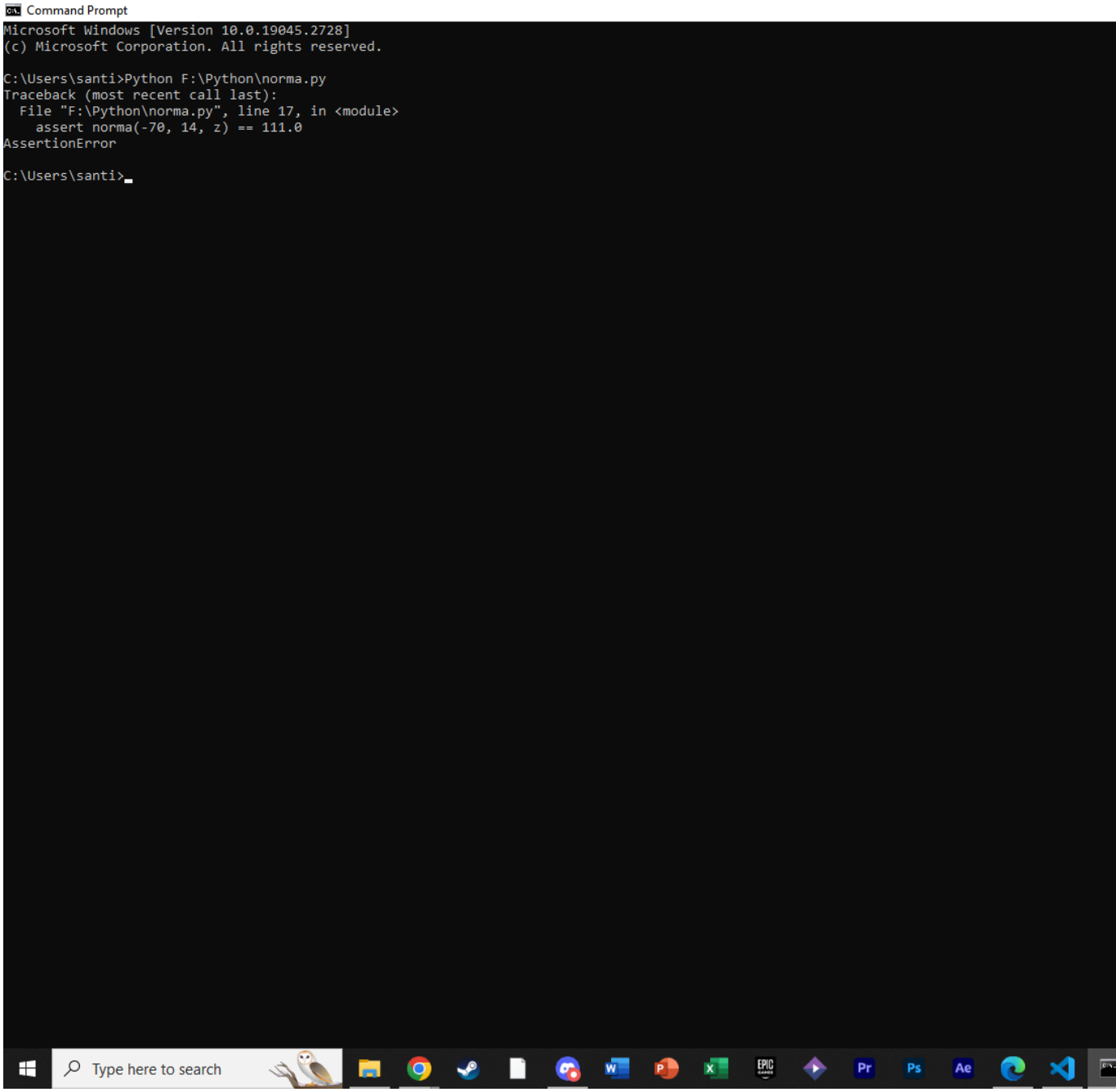


Para conseguir el mismo resultado que en la parte 1.1, deberíamos usar la función print. En la parte 1.1 no es necesario, ya que estamos poniendo "Hola Algoritmos y Programación I" en el intérprete directamente, en vez de ejecutar un programa.

PARTE 2

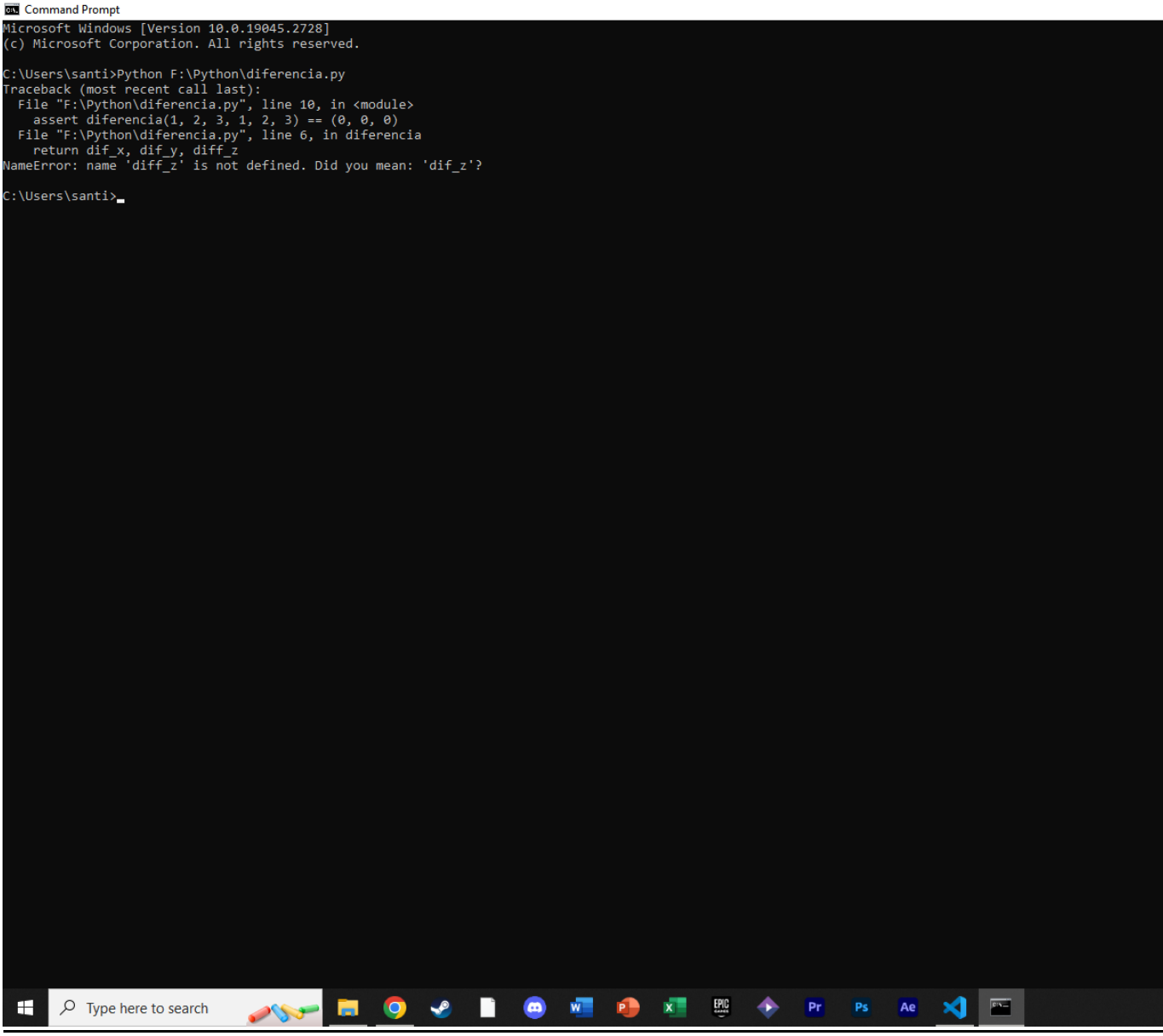

Command Prompt
Microsoft Windows [Version 10.0.19045.2728]
(c) Microsoft Corporation. All rights reserved.
C:\Users\santi>Python F:\Python\norma.py
C:\Users\santi>
Type here to search

Al descomentar las dos últimas líneas, recibimos el siguiente error:



El programa está teniendo un “AssertionError” en la línea 17, como nos indica el intérprete de Python. La instrucción “Assert” permite verificar si cierta condición se cumple o no, en este caso verificando los valores de diferentes vectores. Si hay un AssertionError, es porque la condición no se cumple. Si cambiamos el valor de Z de -80 a +-85, entonces ya no recibiremos ningún error.

PARTE 3



Vemos que se detectó un “NameError” en la línea 6. Esto surge porque se trata de utilizar una variable que no está definida. Si cambiamos la variable de “diff_z” a “dif_z”, el error deja de aparecer.

PARTE 4

El programa muestra un “AssertionError” en la línea 10. Sin embargo, si probamos comentando la línea 10 temporalmente (usando el caracter #) para verificar que el resto del programa funciona bien, vemos que el error aparece nuevamente en la siguiente línea. Este error aparece en todas las líneas desde la 10 hasta la 19. Ahora, si probamos ver cuanto debería dar alguna de estas líneas que esta dando error, vamos a ver que en la coordenada “y” el número es absurdamente grande. Esto sucede porque, a pesar de que estamos recibiendo un AssertionError en las líneas 10 a 19, el problema está en la línea 4. Hay un error de tipeo, y en vez de hacer z1*x2, estamos haciendo z1**x2. Es decir, estamos elevando la coordenada z1 a x2. Una vez corregido ese error de tipeo, el resto de líneas dejan de tener algún error.

PARTE 4.6

Es importante reutilizar funciones porque se ahorra bastante tiempo.