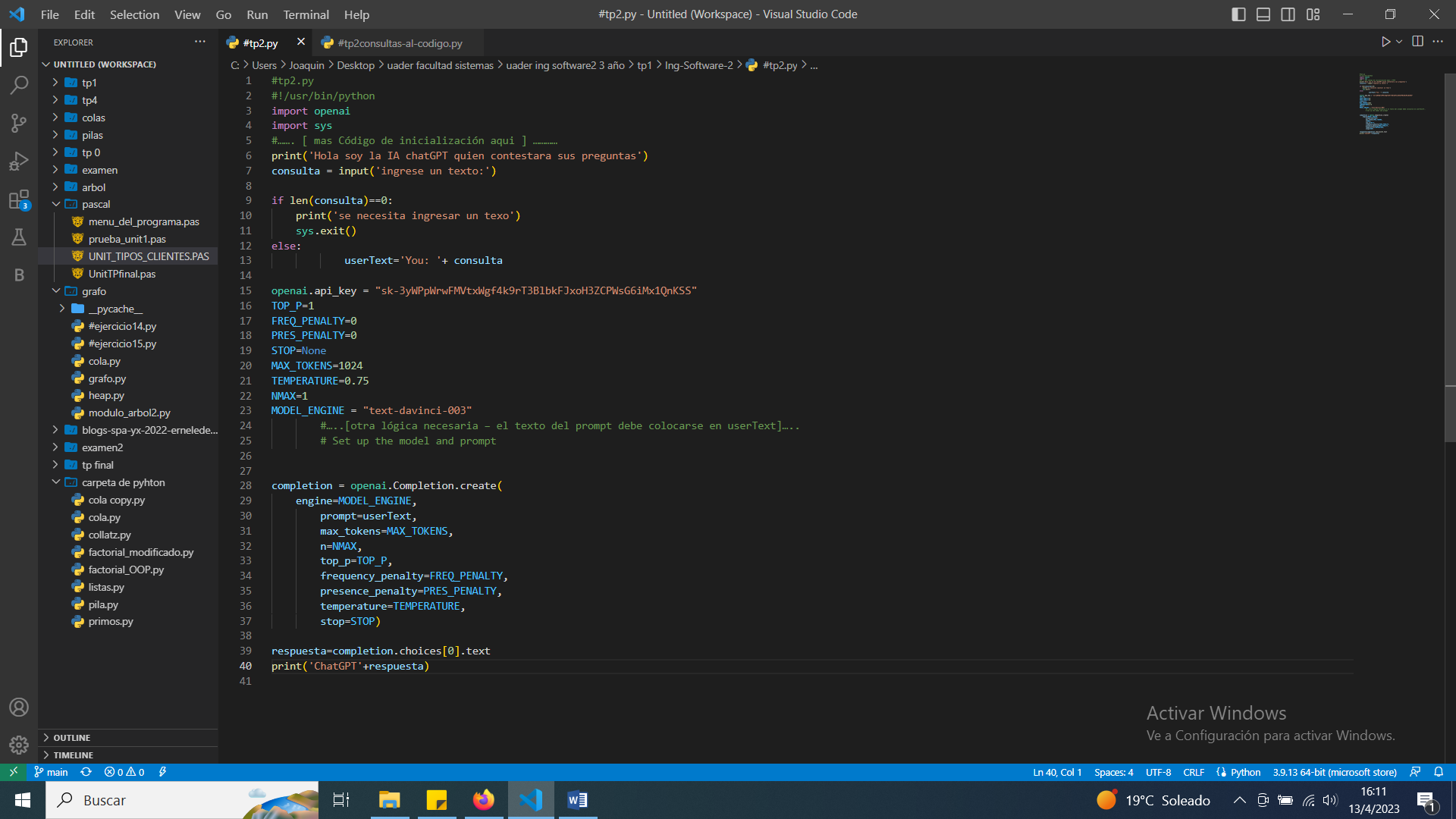
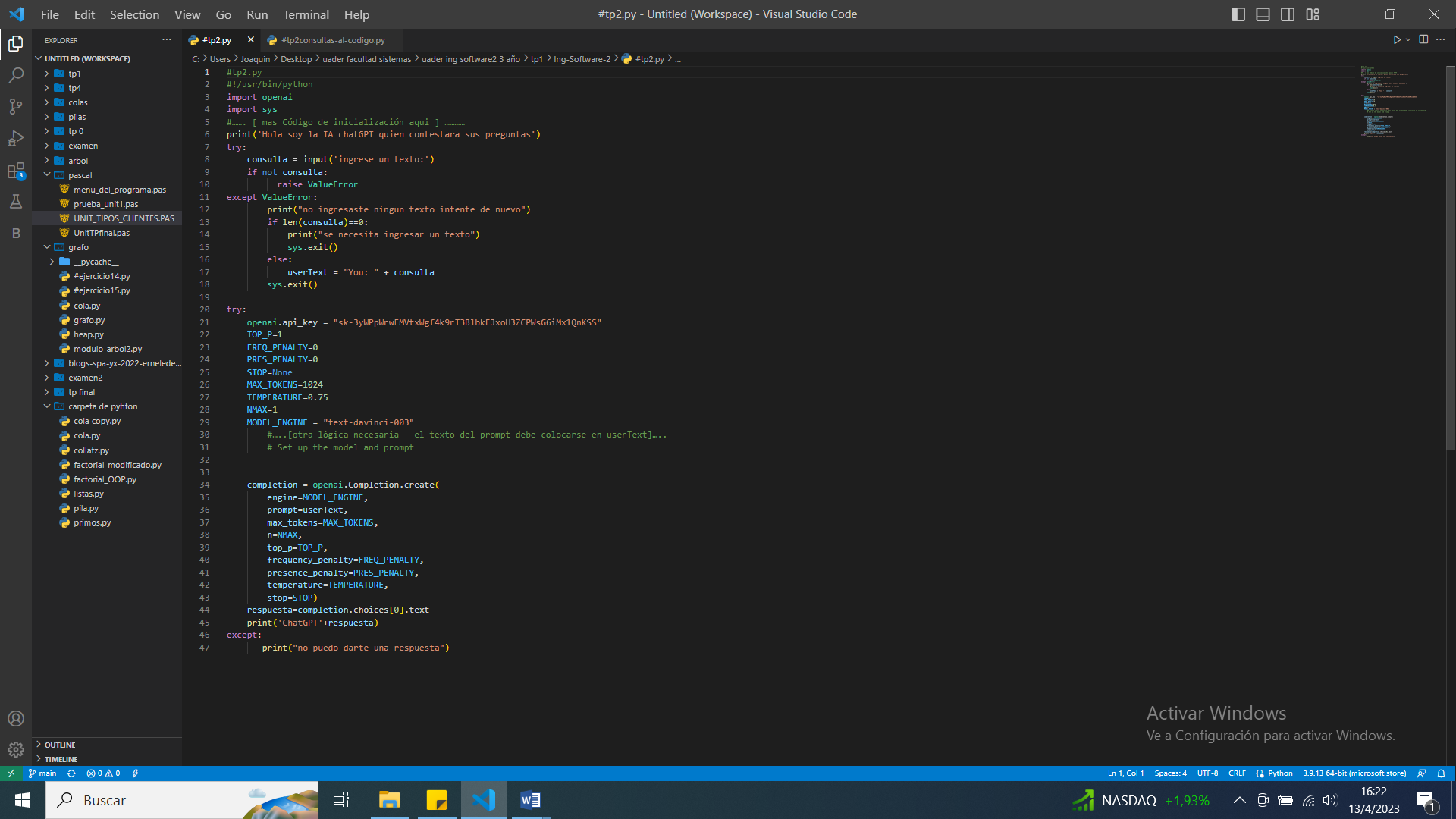
Punto 1

Escriba un programa en lenguaje Python que basándose en el esqueleto previamente proporcionado acepte una consulta del usuario, verifique si la misma tiene texto, imprima su contenido, invoque el API de chatGPT con esa consulta e imprima en pantalla el resultado que se obtenga como respuesta. El contenido de la consulta debe agregársele “*You:*” antes de imprimirlo y de enviarlo. La respuesta de chatGPT deberá agregársele “*chatGPT:* “antes de imprimirse.



Punto 2

Agregue al programa anterior estructuras *Try:/Except*: para gestionar problemas en la ejecución, coloque un nido para la aceptación de consulta desde el usuario, otro para su tratamiento y un tercero para la invocación.



Punto 3

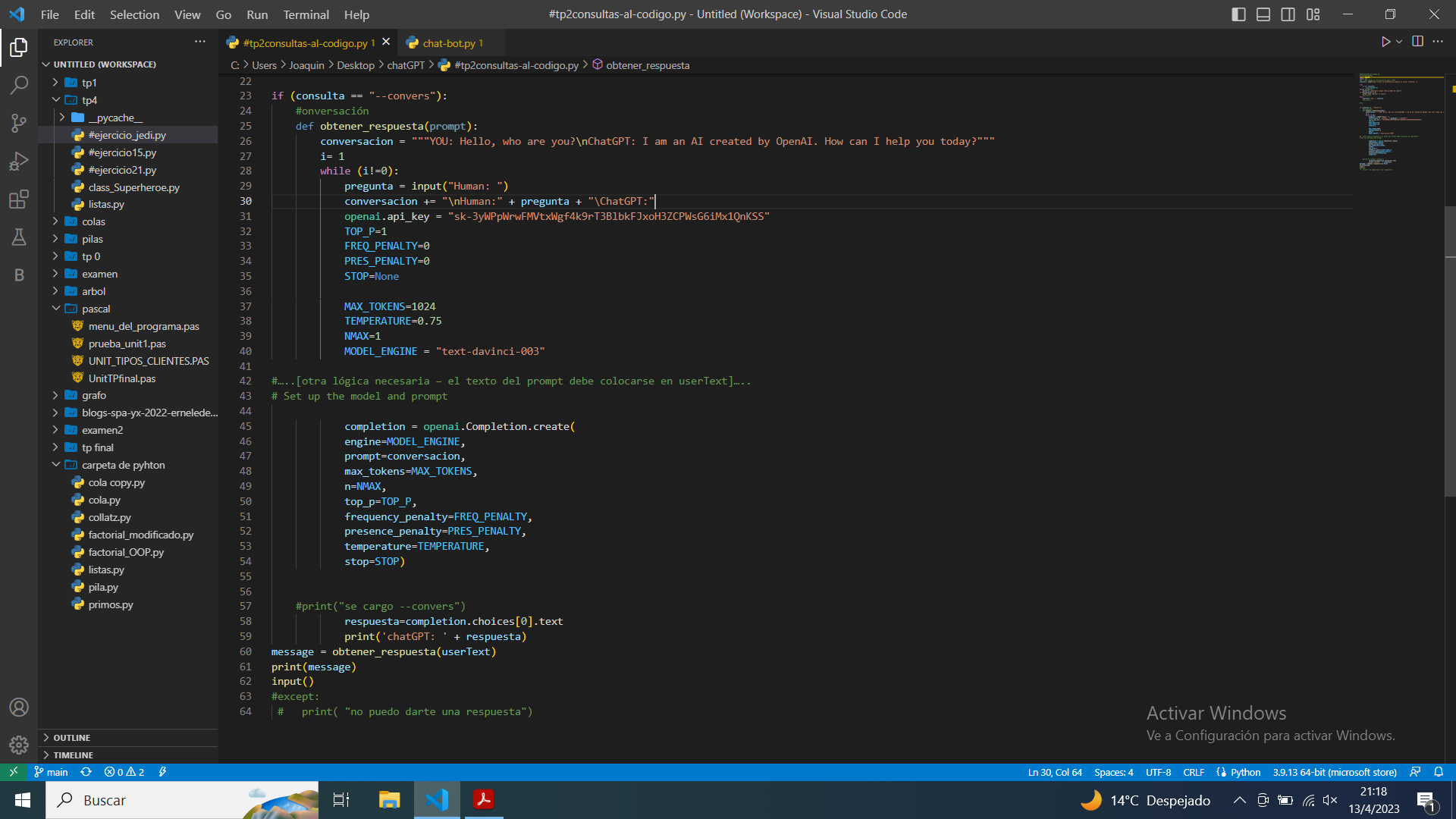
Agregue la posibilidad de recuperar la última consulta realizada para poder editarla y volver a enviar a chatGPT, esto debe hacerse con la tecla “cursor Up”.

Este mismo ya fue resuelto por el mismo Windows lo cual nos comentó el profesor Hernan que ya estaba resuelto ya que dicha función venia incorporada en la pc.

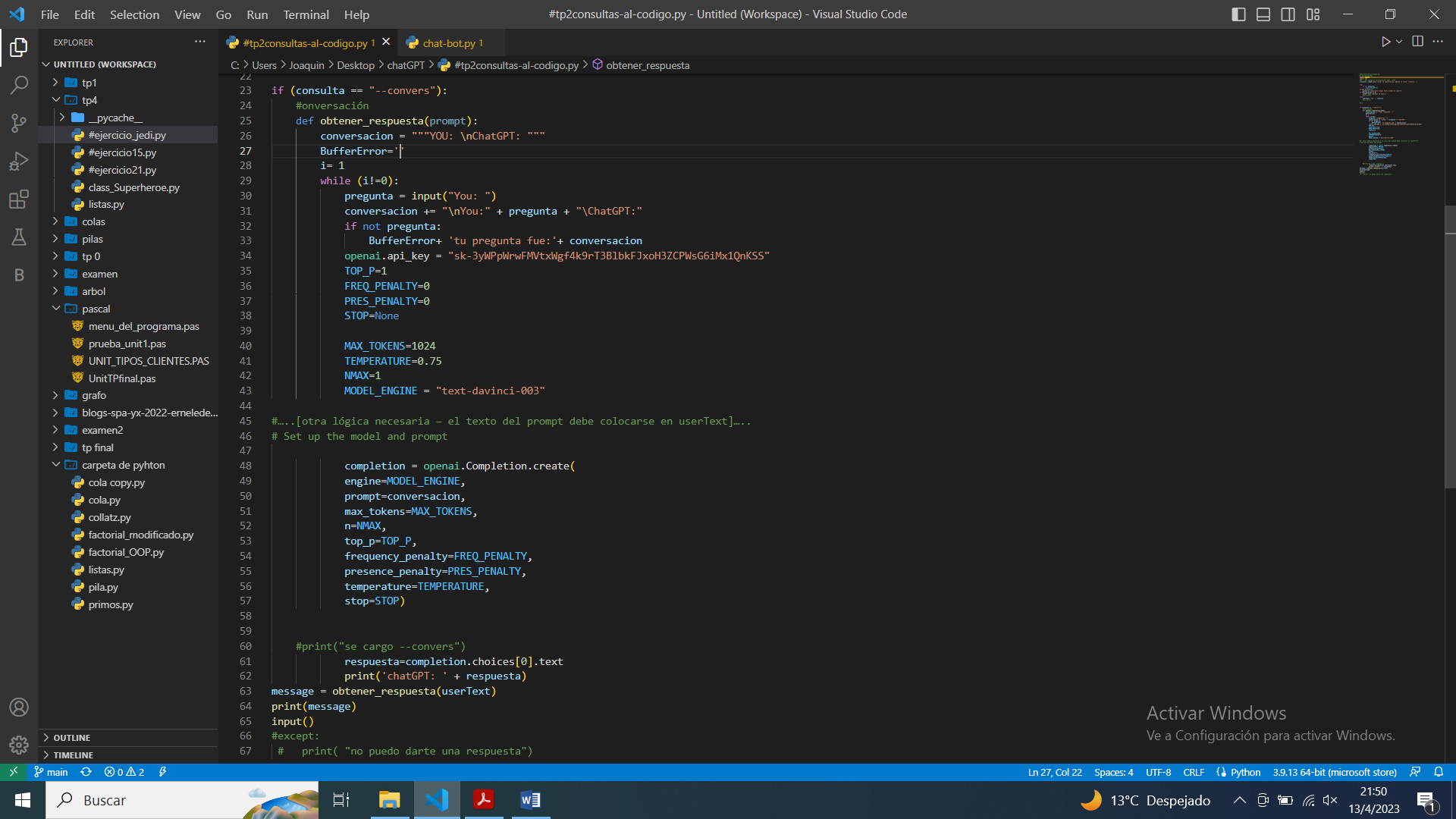
Punto 4

Acepte un argumento de llamada “*—convers*” y en caso de existir genere un modo de “conversación” caracterizado por:

a. Asegúrese de revisar si el argumento existe y es el esperado, caso contrario debe ser ignorado (consejo: revise primero si se ha indicado algún argumento).



b. Cada vez que se acepta una consulta no nula se agrega a un buffer.



c. Cada interacción con *chatGPT* utilizará la última consulta realizada agregada a todas las anteriores.

d. La respuesta de *chatGPT* se agregará al buffer para ser también re-enviada en las próximas consultas.

e. Al actualizar el contenido de la consulta o la respuesta al buffer asegúrese que se incluyen los prefijos “*You:* “ y “*chatGPT:* “, en las llamadas de API realizadas bajo ésta modalidad debe utilizarse un contenido del parámetro STOP con lo siguiente:

STOP=[“You:”,”chatGPT:”]

<https://github.com/JoaquinMAGimenez/UADER_IS2_GIMENEZJOAQUIN>