**ALGEBRA ABSTRACTA**

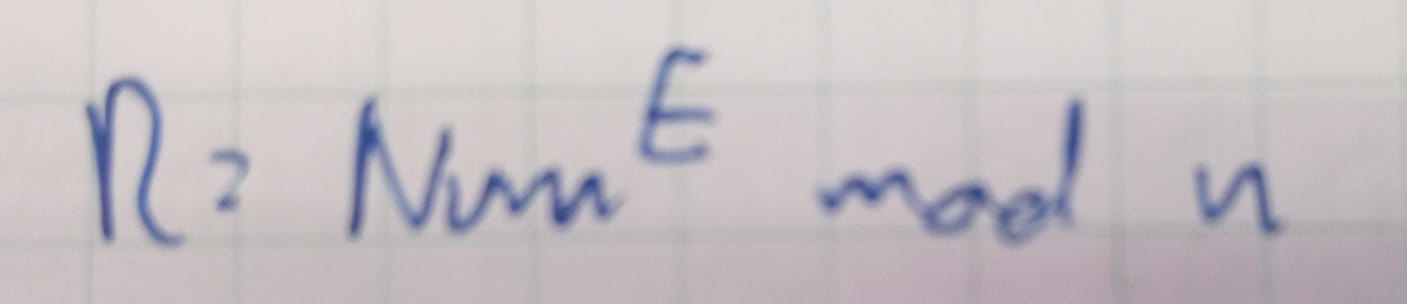
**TERCER CONTROL**

1. Escribir un programa que halle el residuo R de dividir un número Num elevado un exponente E módulo n

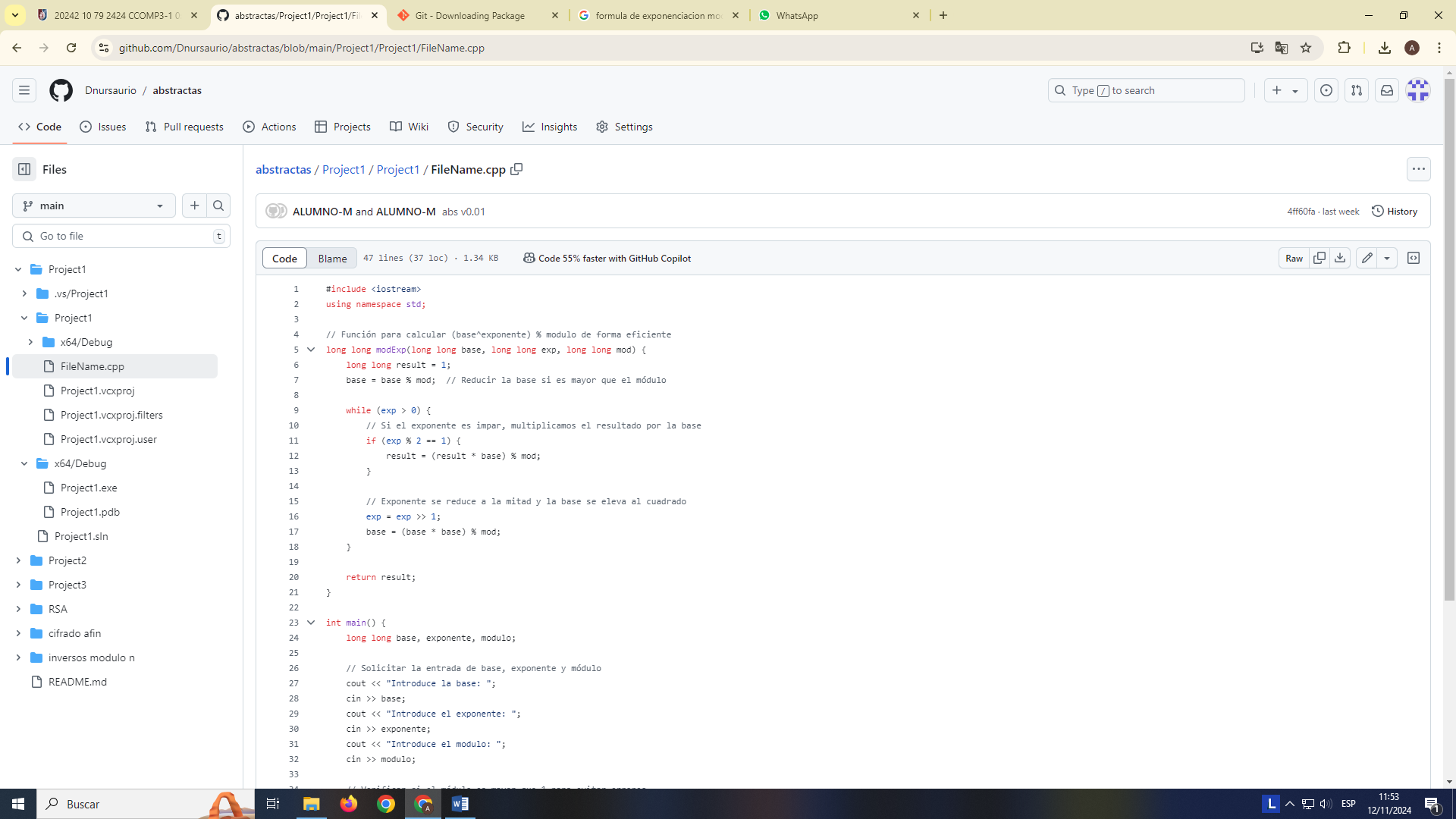
R = PotenciaModulo(Num,E,n))

Inserte el código del programa y en modo de comentario diga:

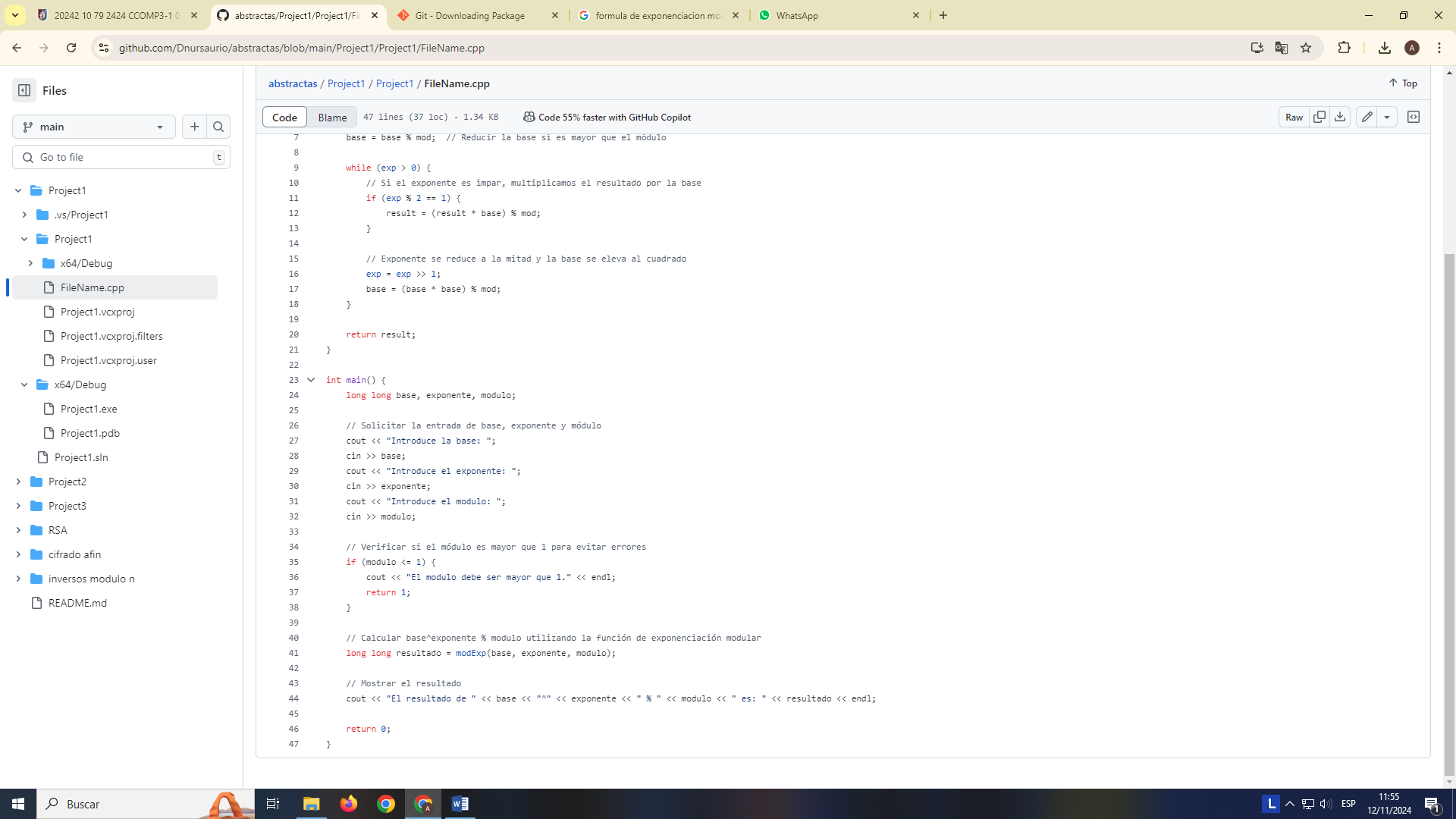
* 1. (1 punto) Pasos relevantes que usa para hallar este residuo
     + Basados en la siguiente fórmula:



* + - Con esto sacamos el exponente modulo n

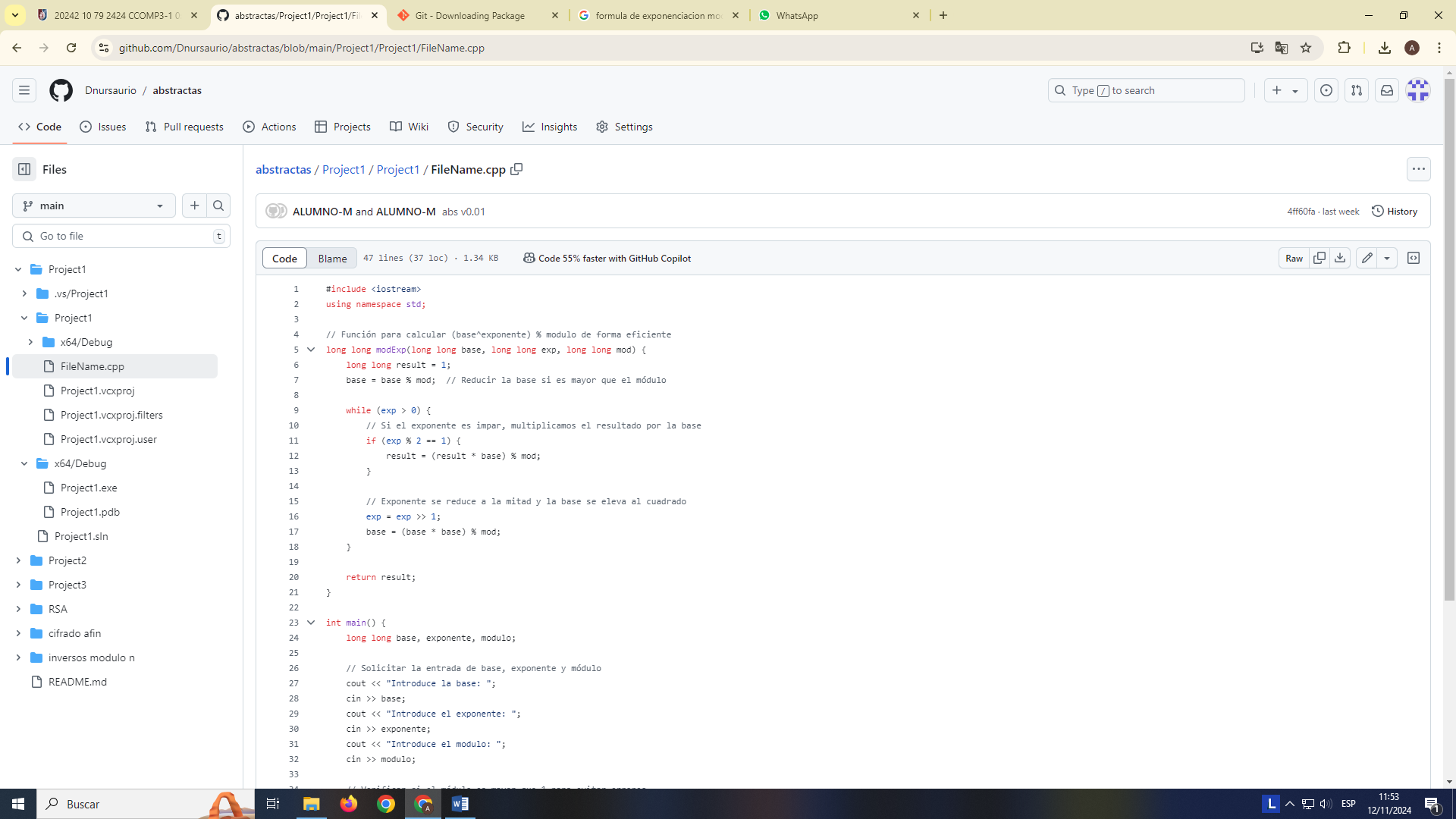


* + - Luego llamamos a la función principal

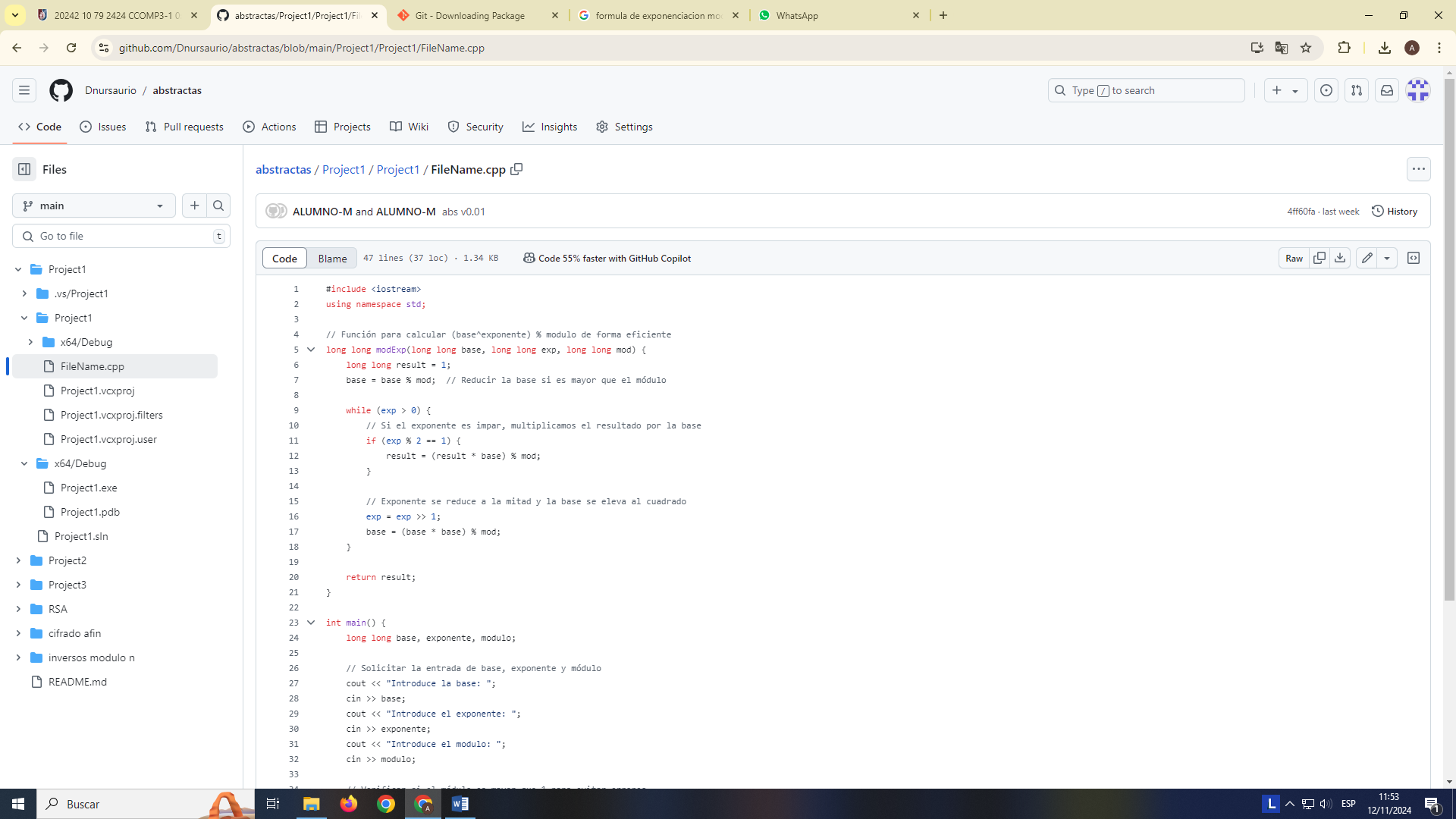


Donde solicitamos una base un exponente y un numero de módulo, para luego hacer uso de la fórmula y obtener el resultado

* 1. (1 punto) Finalidad de cada una de las variables locales utilizadas
     + En la función modExp, donde nos encargamos de todos recibe tres parámetros uno que es la base, otro que es el exponente y finalmente el número que recibe el modulo



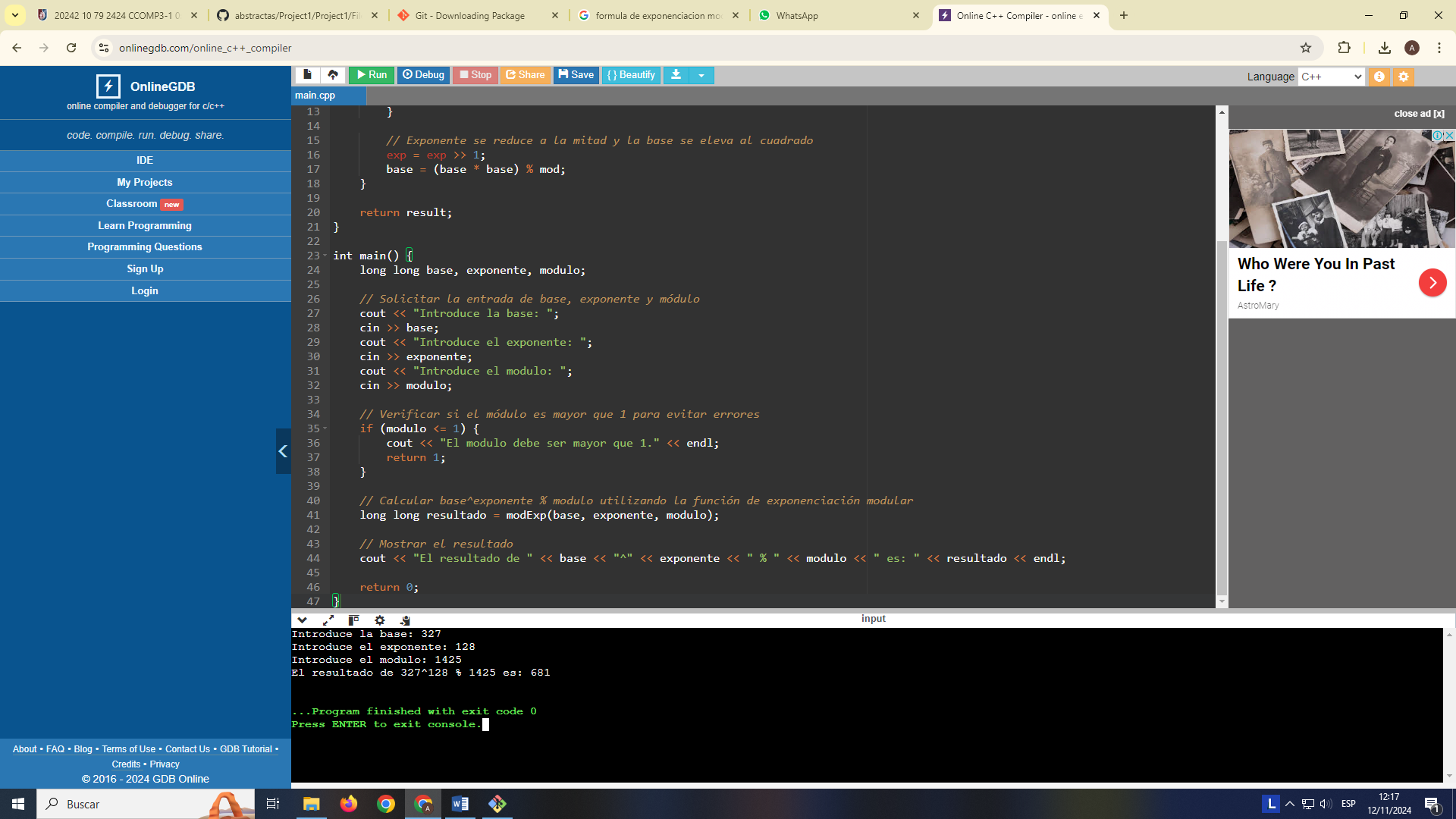
Luego evaluamos si el exponente es mayor que 0 paa operar con ello de sr caso contrario ósea con un exponente igual el resultado será 1



Después evaluamos si el exponente es par para multiplicar el resultado por la base y finalmente recudir el exponente y hallar el modulo de la bese al cuadrado devolviéndonos el resultado.

* 1. (1 punto) Finalidad de cada una de las funciones invocadas en el proceso
     + Aquí usamos únicamente dos funciones modExp y la función principal:

1. En modExp solicitamos los componentes de la formula y luego se encarga de calular el resultado con lo antes mecionado en el punto anterior.
2. Finalmente usamos la función principal como formato de salida para la ejecución del programa
   1. (2 puntos) Ilustre el cálculo del residuo con Num=327, E=128, n=1425



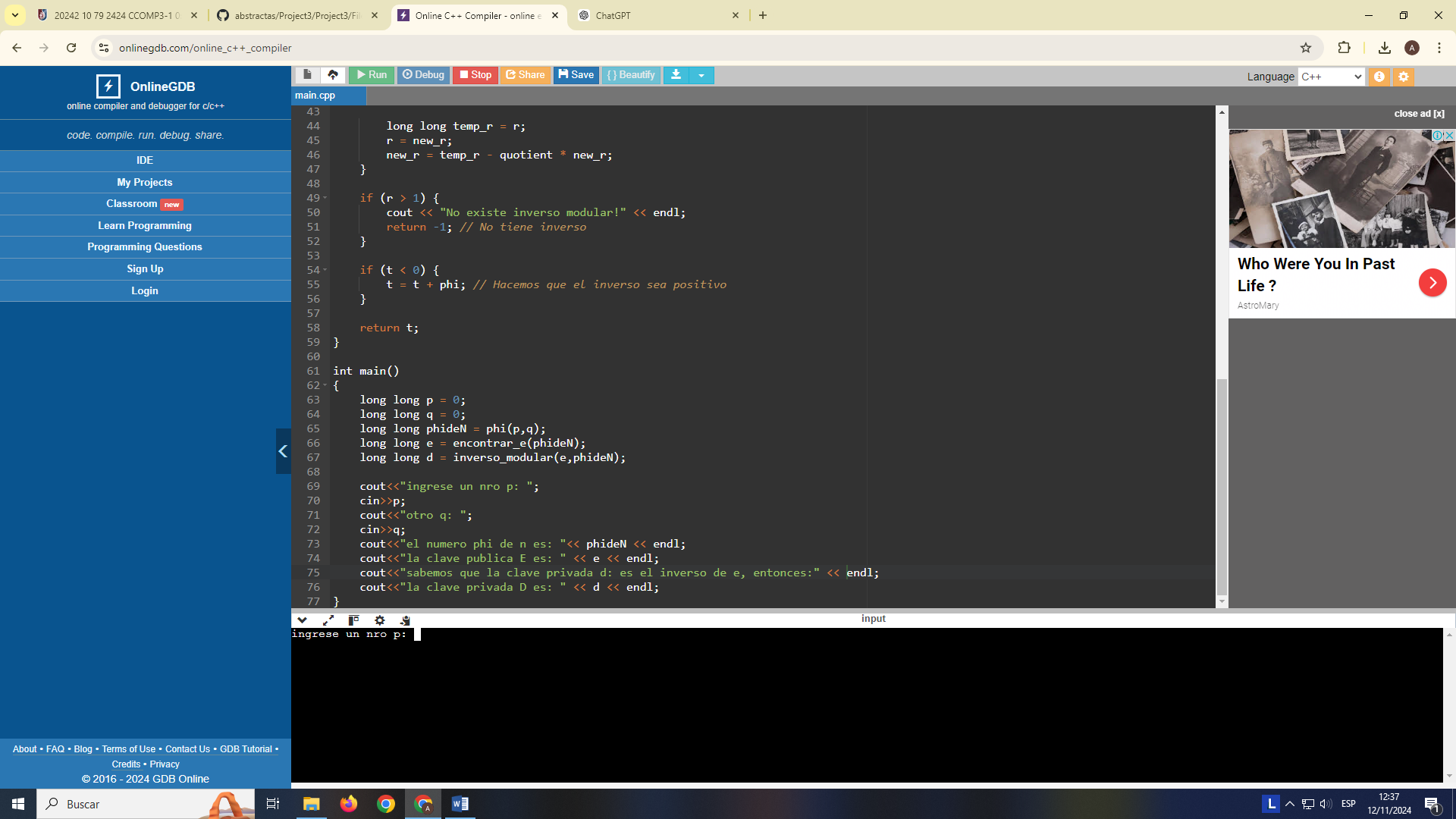
El resultado es 681.

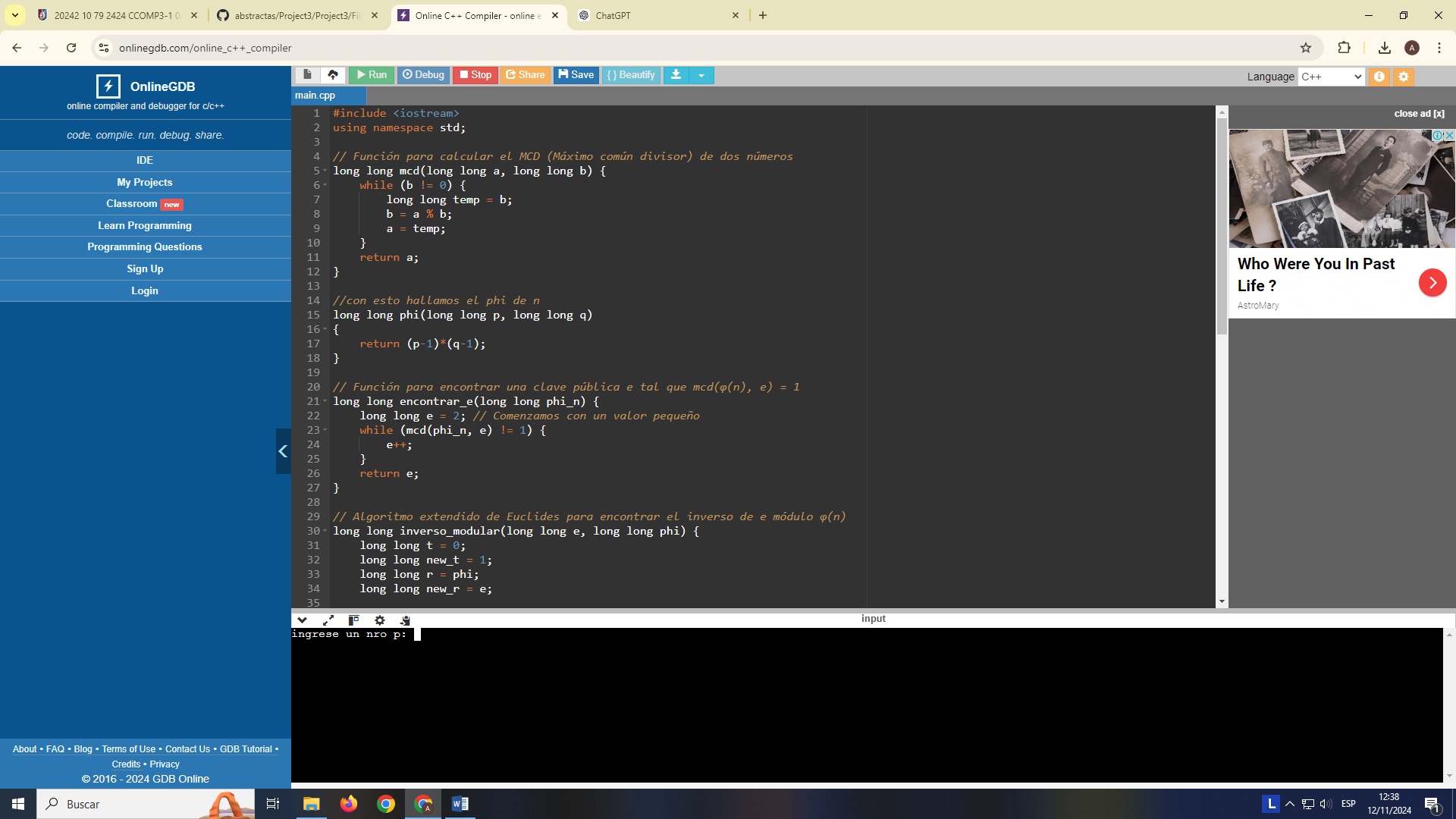
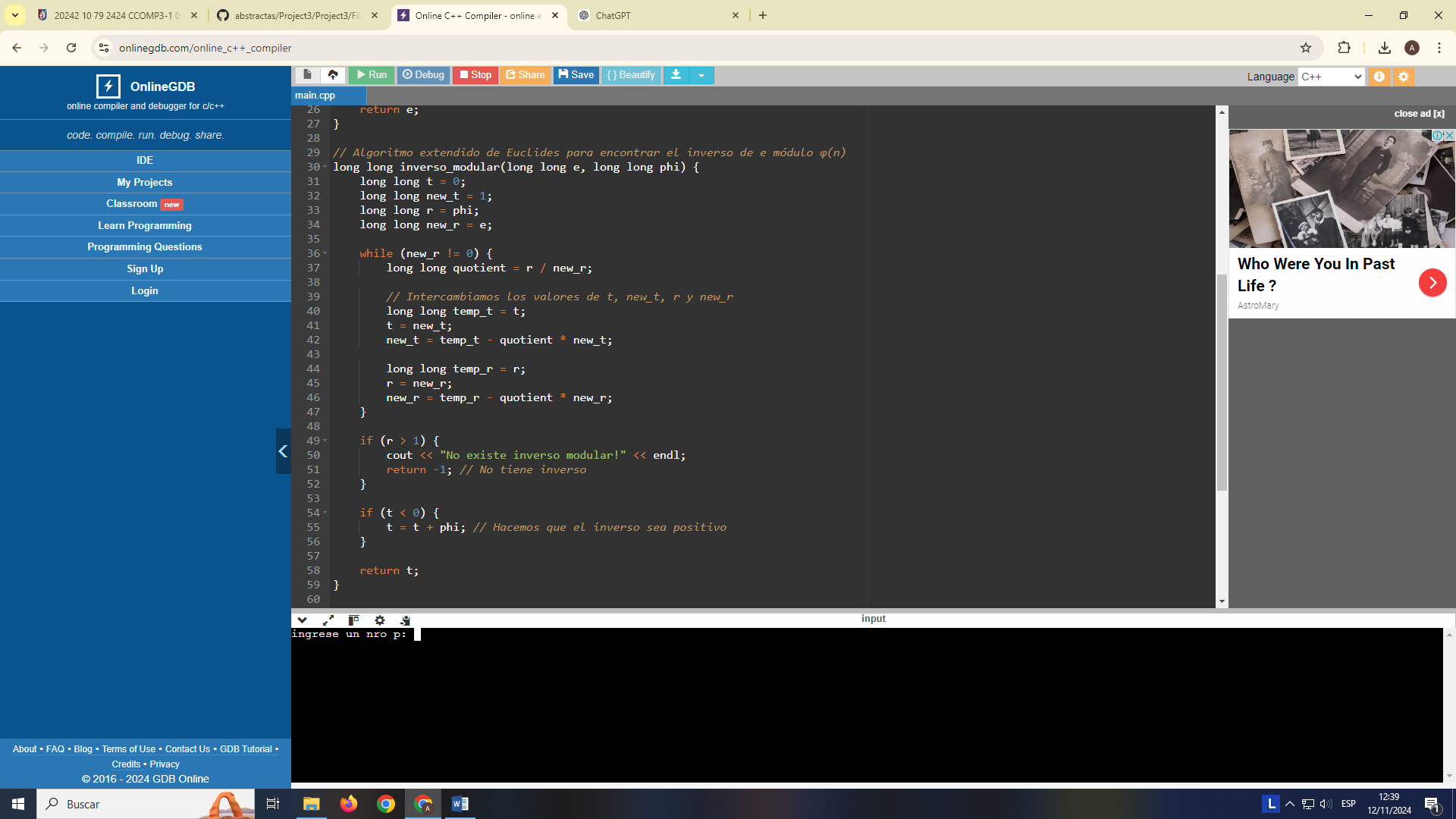
1. Escribir un programa que genere una clave pública e y otra privada d a partir de dos números primos p, q

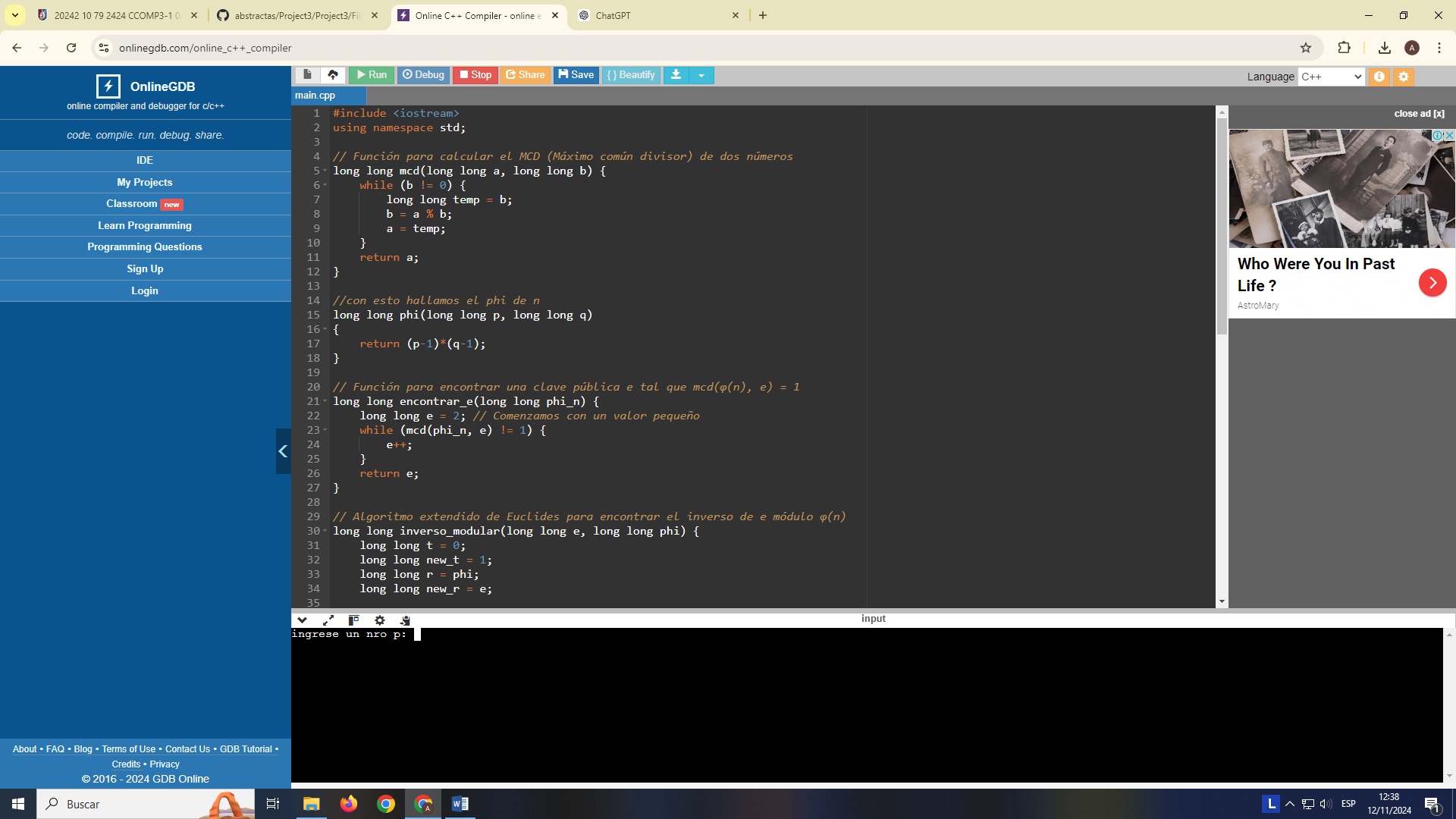
[e, d] = GeneraClave(p, q)

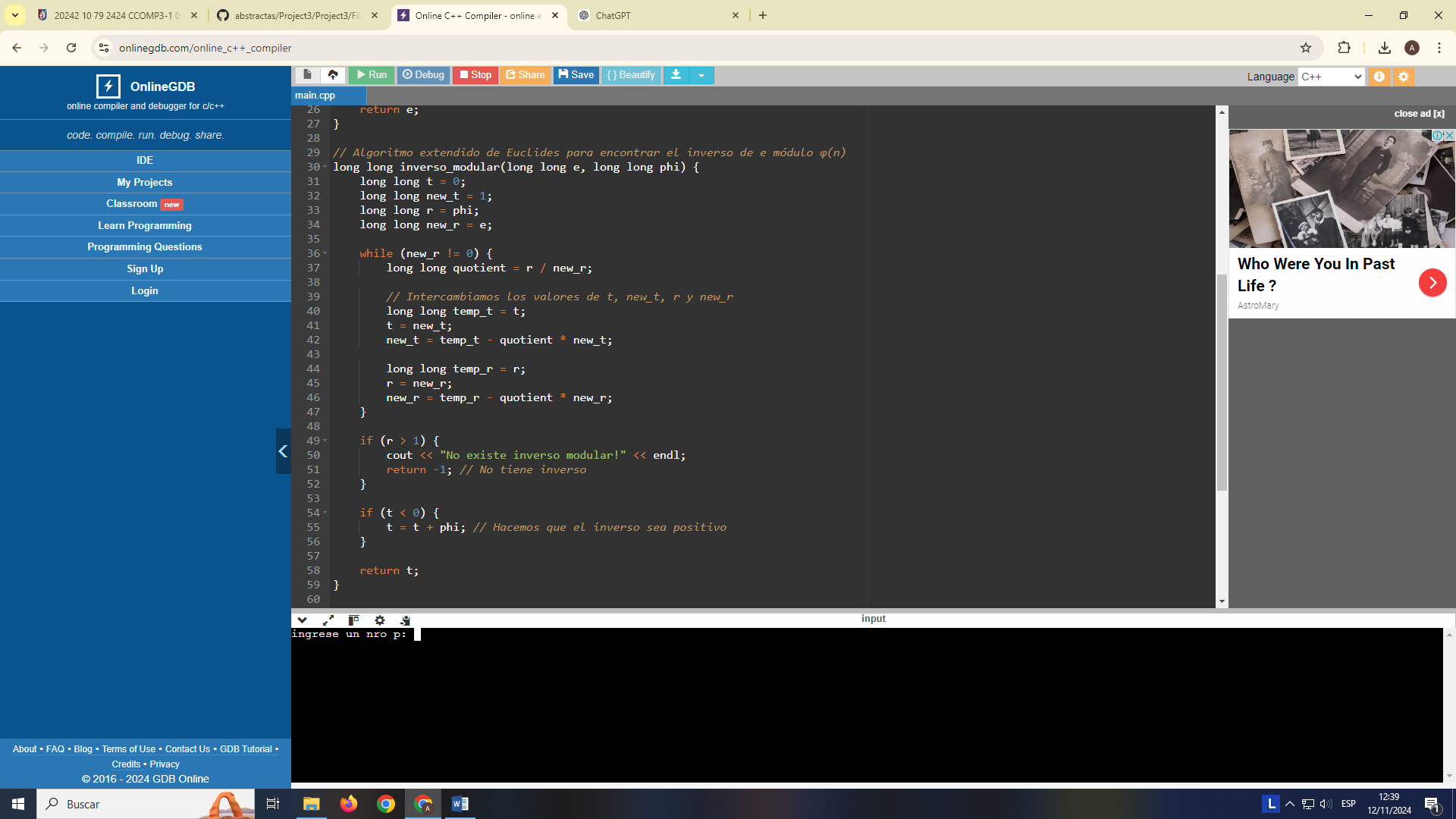
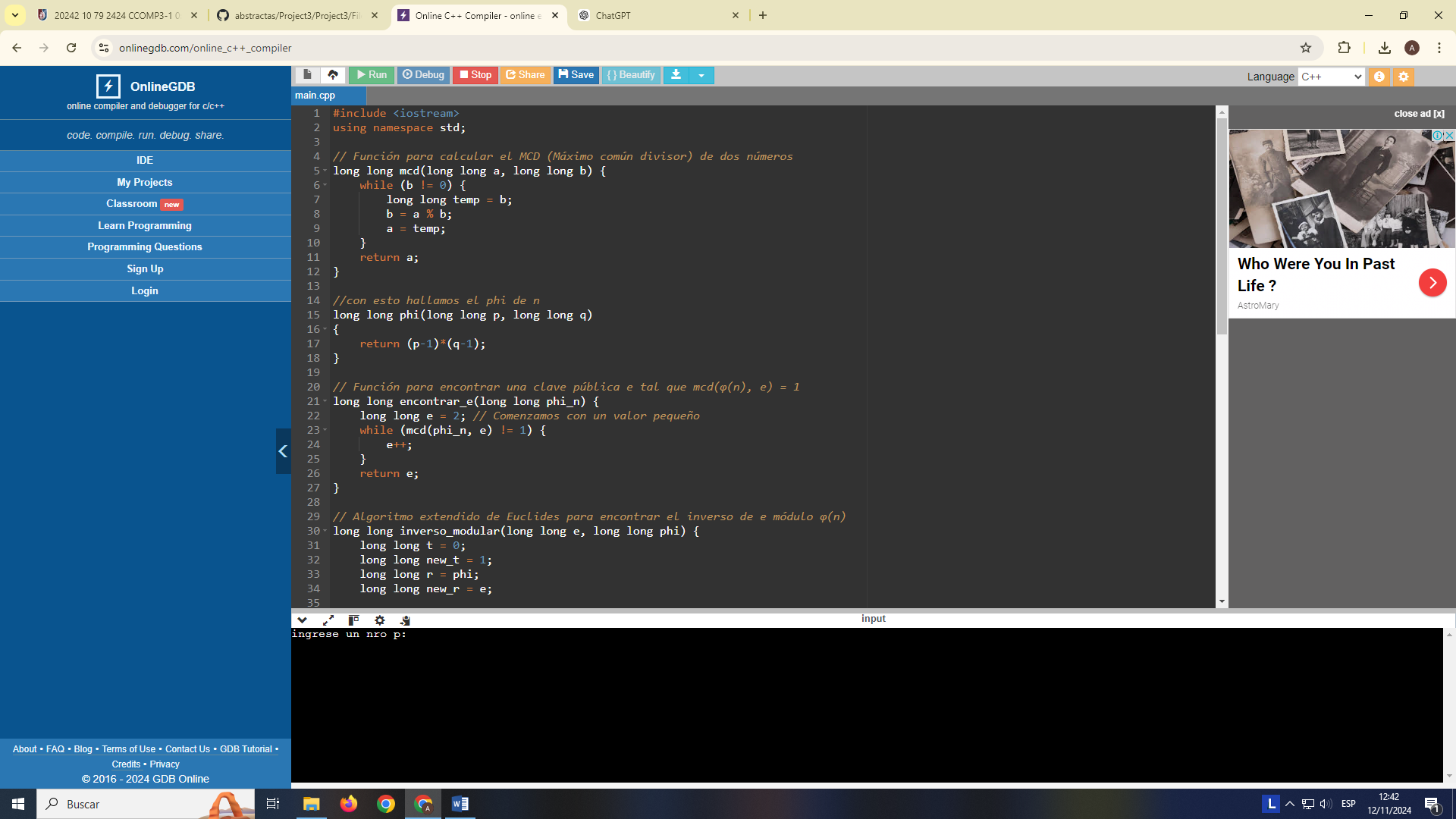
Inserte el código del programa y en modo de comentario diga:

* 1. (1 punto) Pasos relevantes para hallar estas claves.
     + Primero debeos solicitar los numero p y q



* + - Luego debemos de hallar e que es el resultado de que en el mcd de phi de n y e salga 1.
    - 
    - Luego hallamos el inverso modular de e para hallar la clave privada d
    - 
  1. (1 punto) Finalidad de cada una de las variables locales utilizadas
     + En el caso del mcd almacenamos un a y un b que nos ayudaran a hacer las divisiones contantes hasta que ambos sean 1.



* + - En el caso del inverso modular usamos los números e y phi de n para luego usar las ecuaciones de residuos y sacar el inverso
    - 
    - -luego en phi de n usamos nos numero p y q para multiplicarlos a cada uno reduciéndole una unidad.
    - 
  1. (1 punto) Finalidad de cada una de las funciones invocadas en el proceso
     + Como ya se mencionó, la función del inverso modular se encargar de almacenar e y phi den para hallar el d que sería la calve privada.
     + En el mcd hallamos un máximo común divisor normal, pero en este paso lo usaremos para hallar la clave publica e
     + En la phi de n nos encargamos de hallar el valor phi de n para las otras dos funciones.
  2. (2 puntos) Ilustre el cálculo del residuo con p=19, q=51



E = 7 y D = 643.

1. Escribir un programa que **cifre** un mensaje utilizando el algoritmo RSA

C=CifradoRsa(Mensaje, ClavePública)

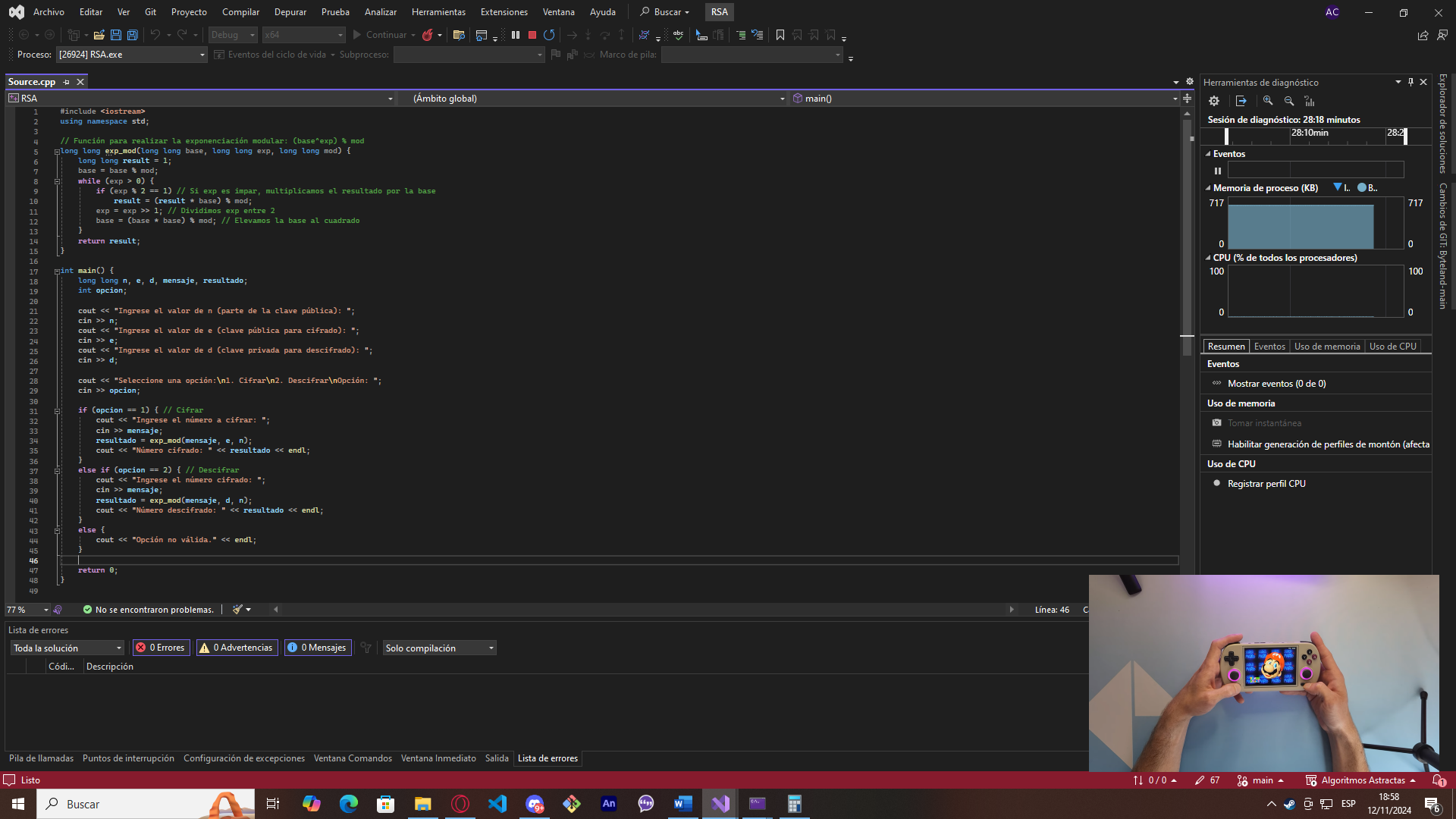
Donde:

* C es el mensaje cifrado.
* ClavePublica = (n, e);
* n es el producto de dos números primos
* e es el número coprimo con (n)

Considere que, en el cifrado, cada carácter es un bloque.

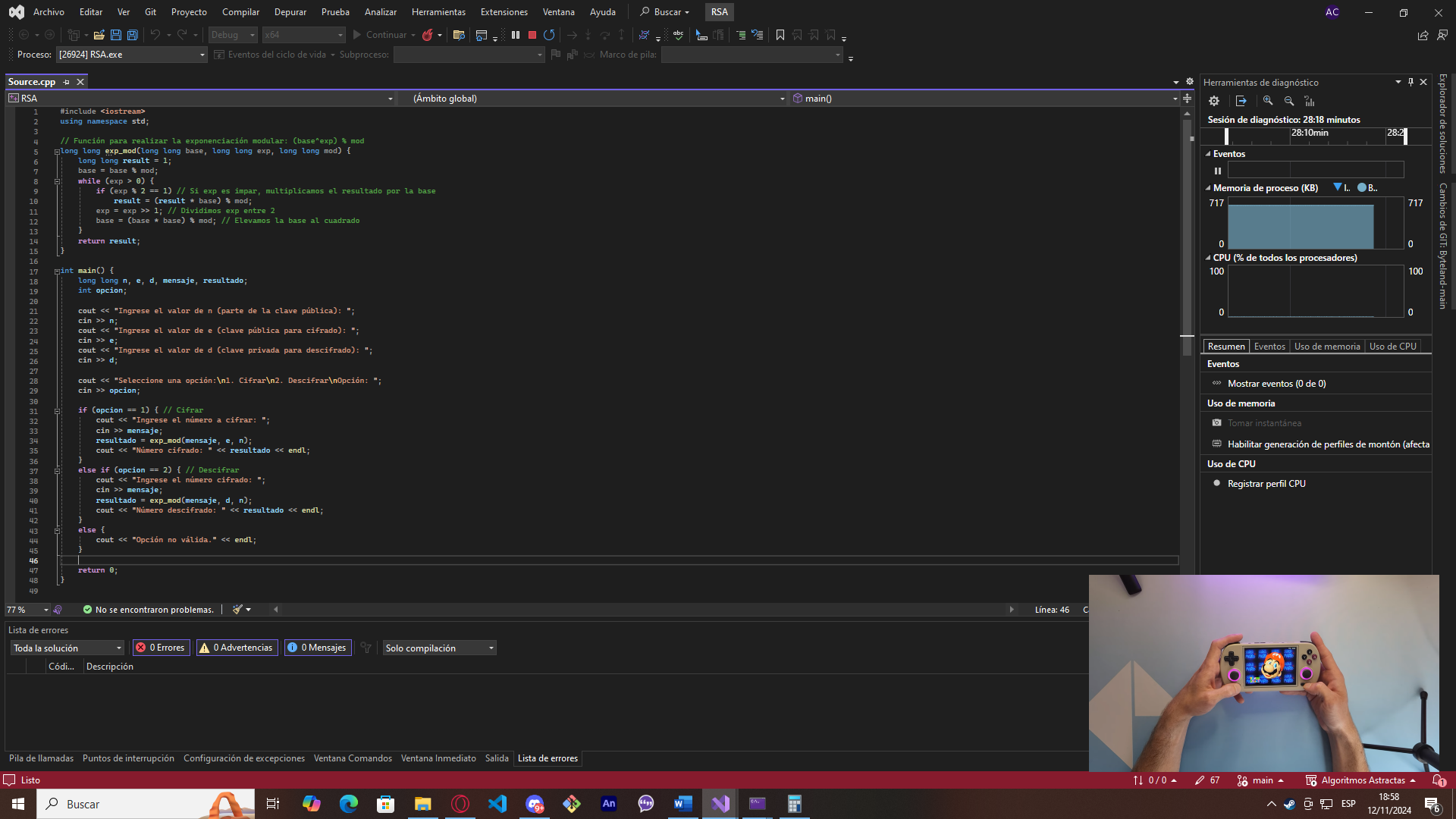
Inserte el código del programa y en modo de comentario diga:

* 1. (1 punto) Pasos relevantes para realizar el cifrado
     + Pues aquí seguimos la formula donde:
     + C = (mensaje ^ clave publica) mod n
     + Donde N es el numero producto de sus factores primos p y q.



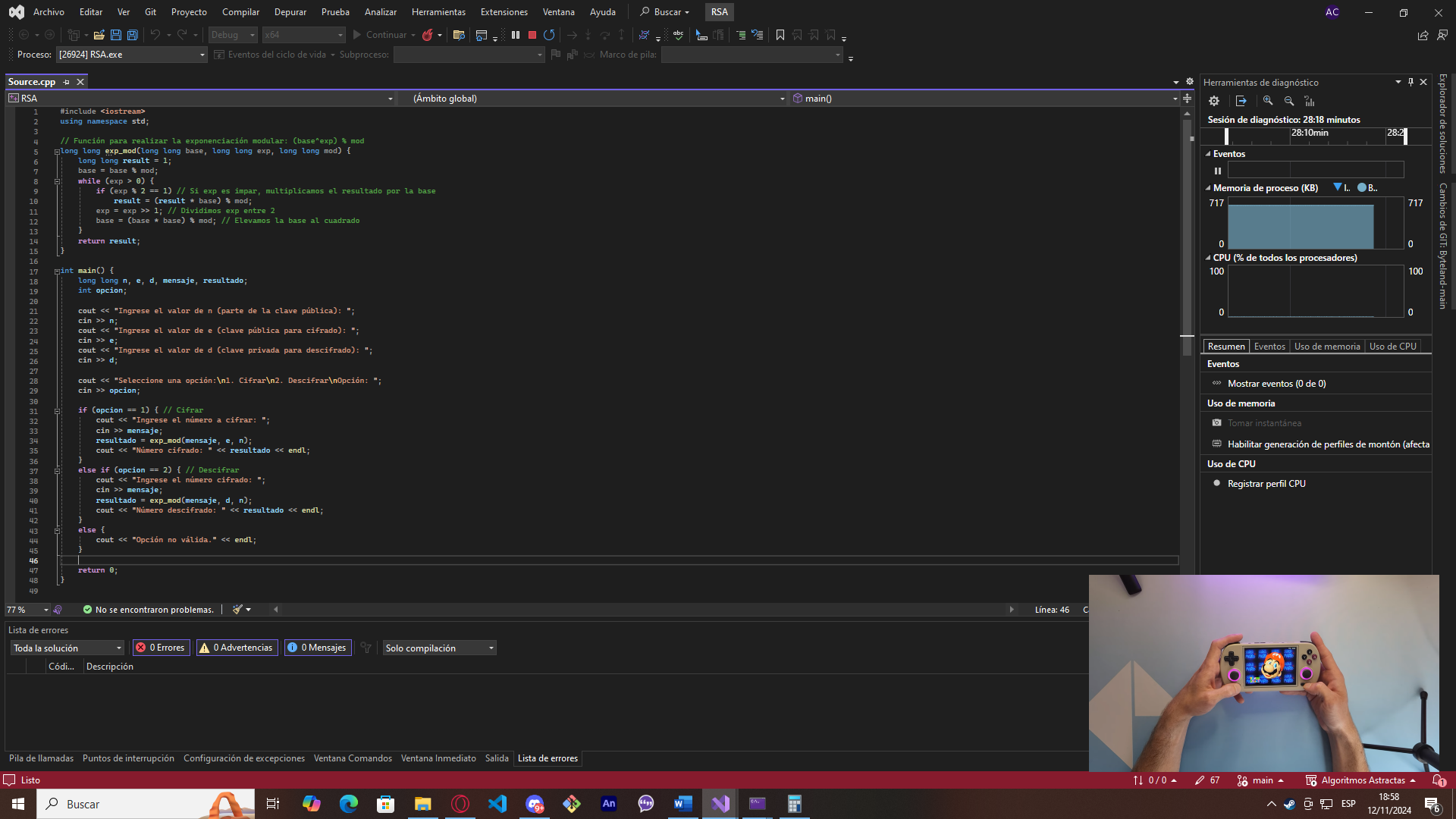
* 1. (1 punto) Finalidad de cada una de las variables locales utilizadas

Pues en esta parte usamos el mismo algoritmo que la exponenciación modular ya que la fórmula es la misma salvo que usa únicamente, así que usamos las variables de una base, un exponente y un módulo. Eso lo único que hace esta parte ya que el desciframiento es lo mismo, la única diferencia son los datos de la formula.



* 1. (1 punto) Finalidad de cada una de las funciones invocadas en el proceso

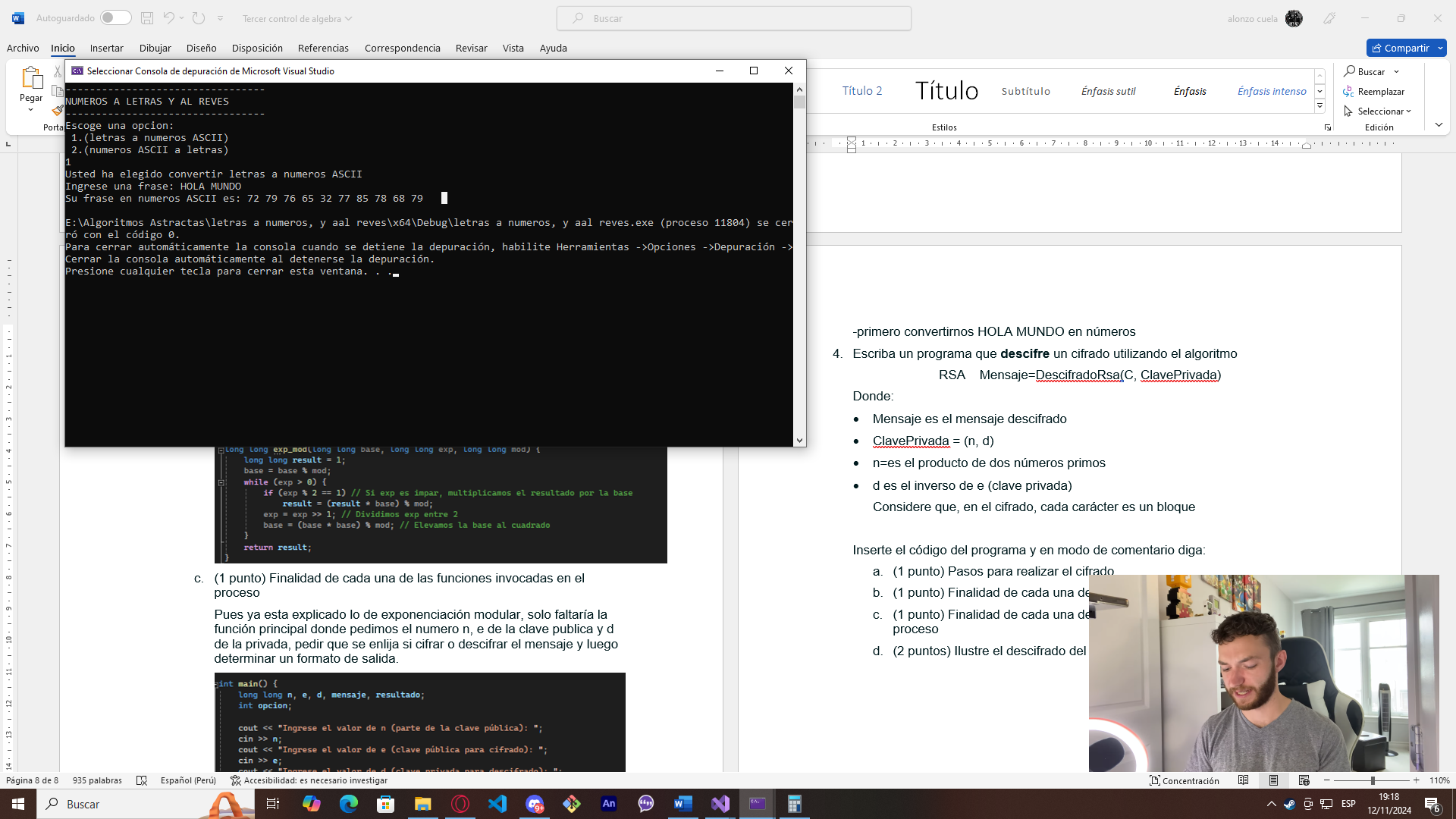
Pues ya esta explicado lo de exponenciación modular, solo faltaría la función principal donde pedimos el numero n, e de la clave publica y d de la privada, pedir que se enlija si cifrar o descifrar el mensaje y luego determinar un formato de salida.



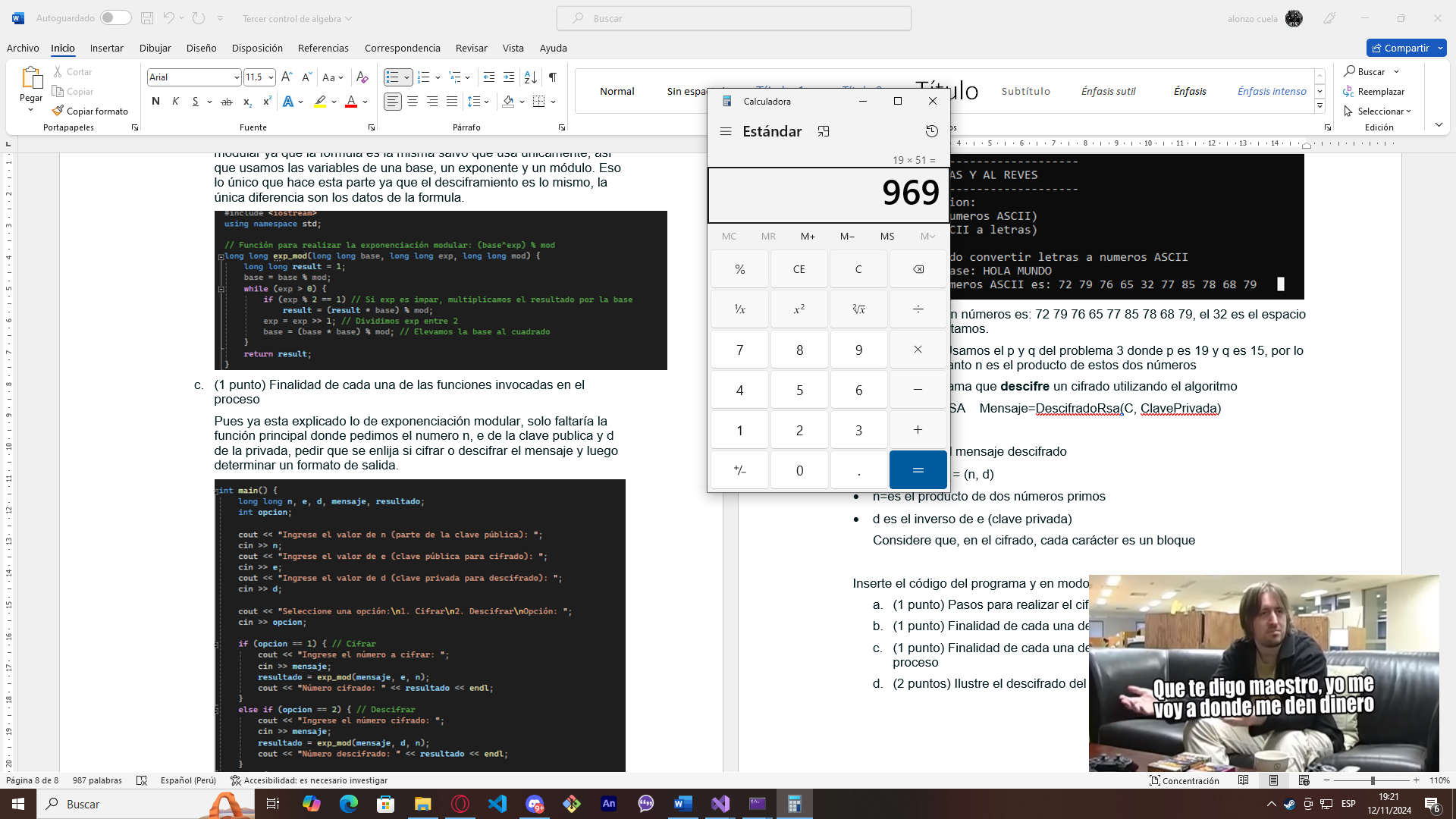
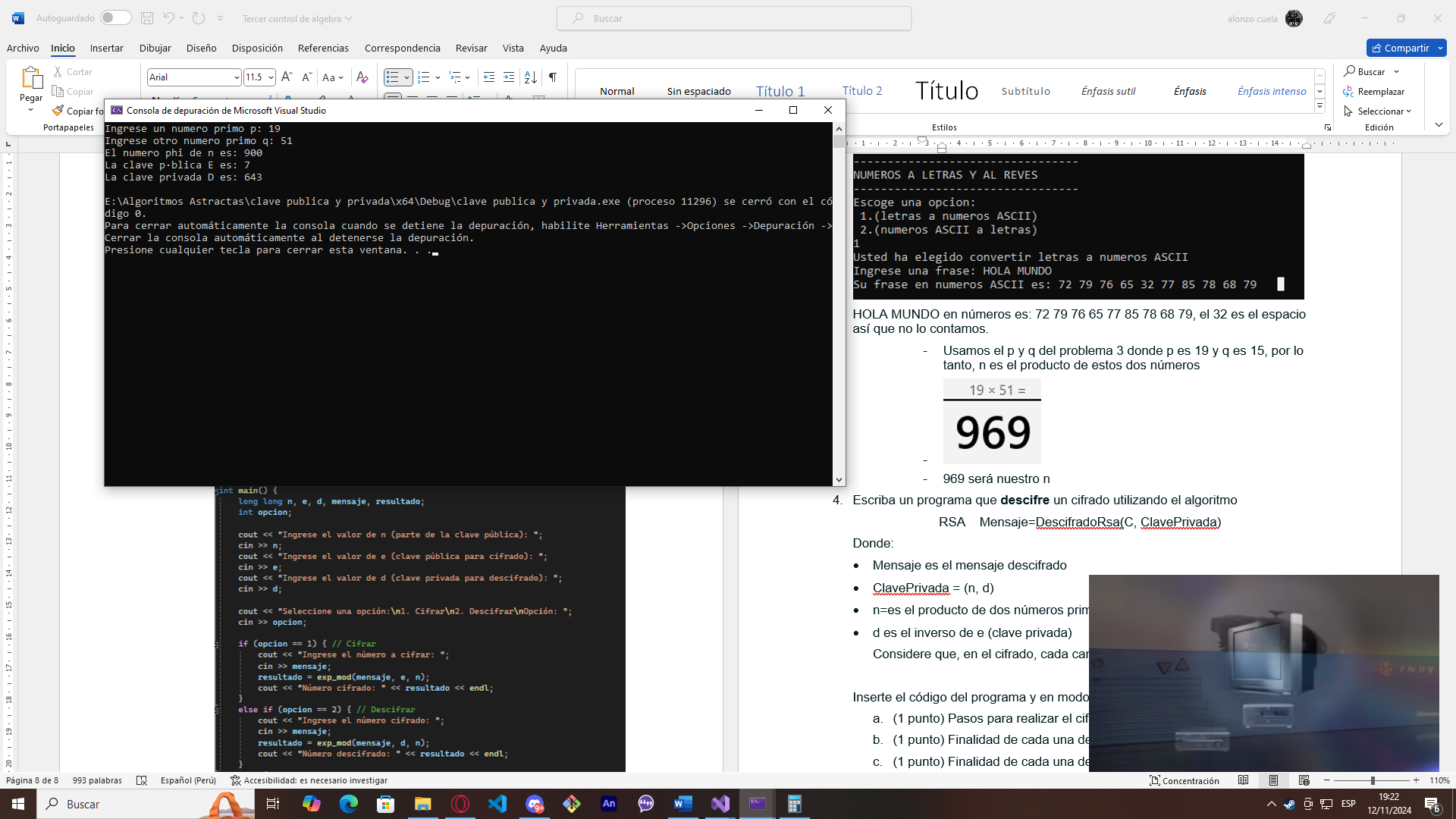
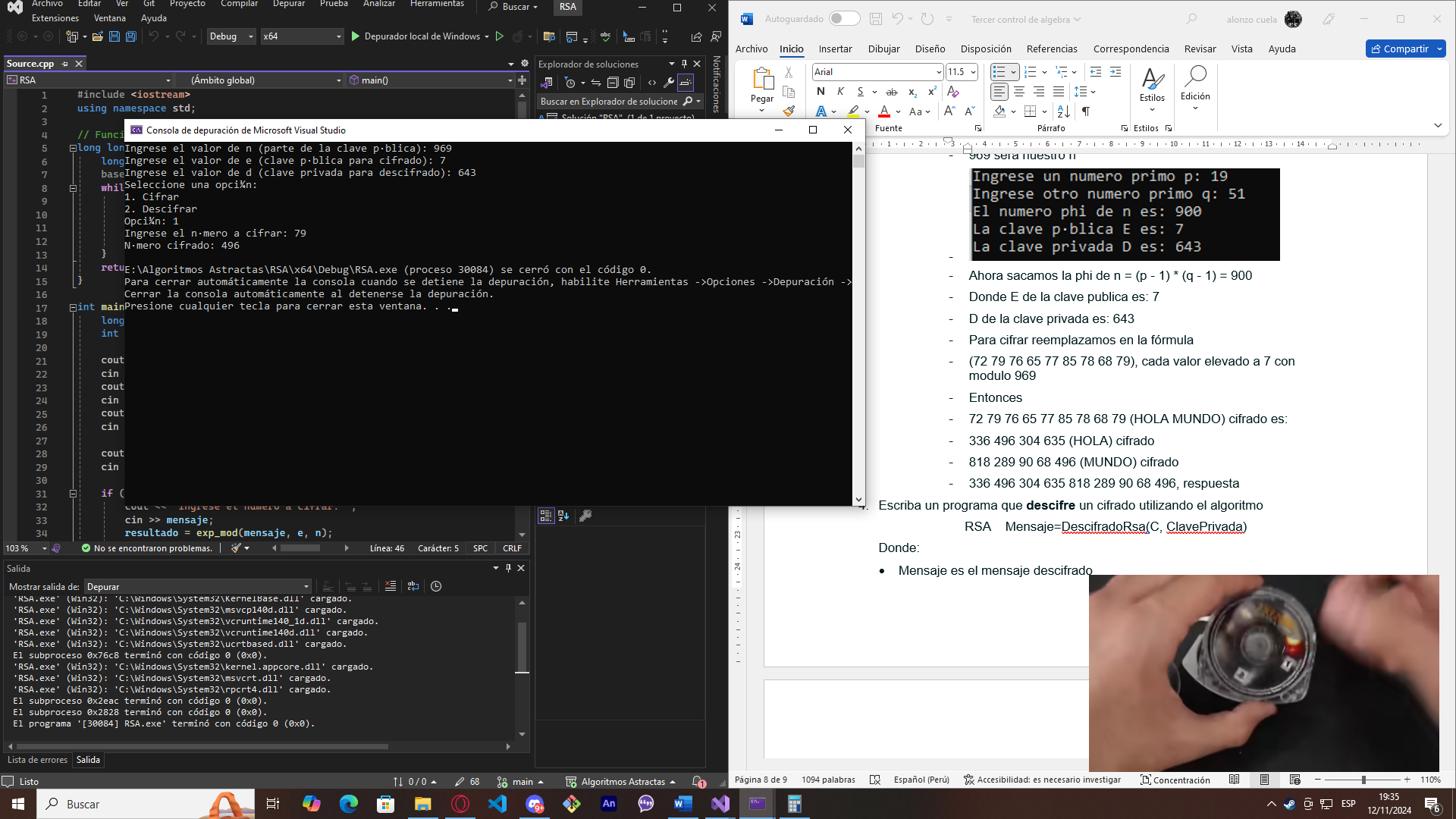
* 1. (2 puntos) Ilustre el cifrado del mensaje “HOLA MUNDO”

Tengo tres programas uno para las claves privadas, otro que se encarga del RSA, y el mago del ejercicio en encargado de convertir números a letra en ASCII

-primero convertirnos HOLA MUNDO en números



HOLA MUNDO en números es: 72 79 76 65 77 85 78 68 79, el 32 es el espacio así que no lo contamos.

* + - Usamos el p y q del problema 3 donde p es 19 y q es 51, por lo tanto, n es el producto de estos dos números
    - 
    - 969 será nuestro n
    - 
    - Ahora sacamos la phi de n = (p - 1) \* (q - 1) = 900
    - Donde E de la clave publica es: 7
    - D de la clave privada es: 643
    - Para cifrar reemplazamos en la fórmula
    - (72 79 76 65 77 85 78 68 79), cada valor elevado a 7 con modulo 969
    - Entonces
    - 72 79 76 65 77 85 78 68 79 (HOLA MUNDO) cifrado es:
    - 336 496 304 635 (HOLA) cifrado
    - 818 289 90 68 496 (MUNDO) cifrado
    - 336 496 304 635 818 289 90 68 496, respuesta
    - Prueba de cifrado de RSA por unidades, es decir letra por letra
    - 
    - Conversión de la letra O = 79 en ASCII, cifrado con RSA = 496

1. Escriba un programa que **descifre** un cifrado utilizando el algoritmo

RSA    Mensaje=DescifradoRsa(C, ClavePrivada)

Donde:

* Mensaje es el mensaje descifrado
* ClavePrivada = (n, d)
* n=es el producto de dos números primos
* d es el inverso de e (clave privada)

Considere que, en el cifrado, cada carácter es un bloque

Inserte el código del programa y en modo de comentario diga:

* 1. (1 punto) Pasos para realizar el cifrado
     + Igualmente seguimos una formula por exponenciación modular:
     + Mensaje = (cifrado ^ clave privada) modulo n
     + Entonces usamos lo mismo
  2. (1 punto) Finalidad de cada una de las variables locales utilizadas
     + Pues realmente es igual al ejercicio anterior con las mismas variables, si que la función es la misma
  3. (1 punto) Finalidad de cada una de las funciones invocadas en el proceso
     + Igual que en el ejercicio 4.
  4. (2 puntos) Ilustre el descifrado del mensaje “24 58 125 130 254”
     + Tenemos (24 58 125 130 254) mensaje cifrado, elevamos cada elemento con la clave privada ( p= 11 y q=13), por lo que n = 143, e=7 y d=103
     + Reemplazando tenemos:
     + Mensaje descifrado = (24 58 125 130 254) ^ 1 modulo 6
     + 41 137 31 91 45 que seria el mensaje decifrado, usando los programas no tiene equivalente en letras
     + 