**“ecuacion\_segundo\_grado”**

**“ecuacion\_segundo\_grado\_complejos”**

**INTRODUCCIÓN A LAS VARIABLES COMPLEJAS Y A LA SENTENCIA “IF”**

Yago Pego Martínez ([yago.pego.martinez@alumnos.upm.es](mailto:yago.pego.martinez@alumnos.upm.es)) Evaristo De Vega Galindo ([evaristo.devega.galindo@alumnos.upm.es](mailto:evaristo.devega.galindo@alumnos.upm.es))

**ESPECIFICACIONES**

**Enunciado:** escribir, compilar y ejecutar un programa que calcule las raíces de una expresión algebraica de segundo grado en todas sus variantes: cuando solo existe una solución, cuando uno de los coeficientes es nulo, cuando las soluciones son complejas…

Deberán acompañarse todas estas demostraciones con los resultados hallados al ejecutar en la consola el programa.

**Objetivo:** comprender la utilización de variables complejas y saber trabajar con ellas, a partir de la ecuación de segundo grado y el uso de la sentencia “if”.

**INTRODUCCIÓN TEÓRICA**

*¿De dónde “sale” la expresión ?*

La forma original es , siendo “a”, “b” y “c” los coeficientes que acompañan a la variable “x”. A continuación, desarrollamos el proceso por pasos para llegar a la expresión del enunciado:

→ Pasamos la “c” al otro lado de la expresión: .

→ Dividimos ambos lados de la ecuación entre “a”: .

→ Deseamos buscar un producto notable de manera que solamente exista una “x” en la expresión. Sumamos en ambos lados de la expresión. Queda .

→ La parte de la derecha es equivalente al producto notable: . Raíz cuadrada en ambos lados de la ecuación y queda: .

→ Se suma en ambos lados y se llega a la tan conocida ecuación de segundo grado: . Se añade antes de la raíz el símbolo ± para explicitar que, al ser una ecuación de segundo grado, existen dos raíces.

**BIBLIOTECA DE VARIABLES**

Comunes para ambos programas:

“a”, “b”, “c” y “D” son variables reales que pueden tomar cualquier valor real (siempre que sea cuantitavamente perteneciente al intervalo numérico ]±10-7, ±1038[.

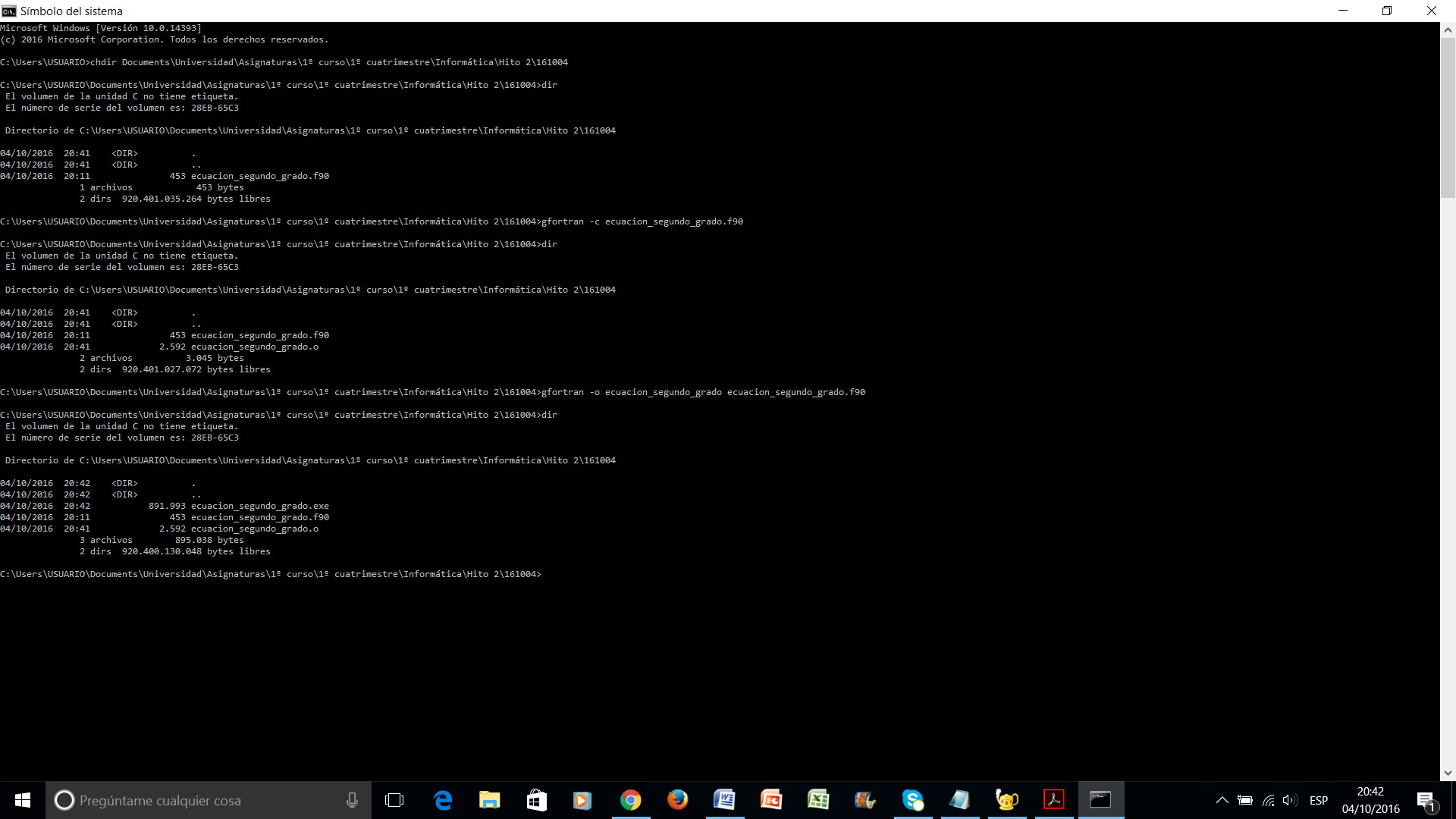
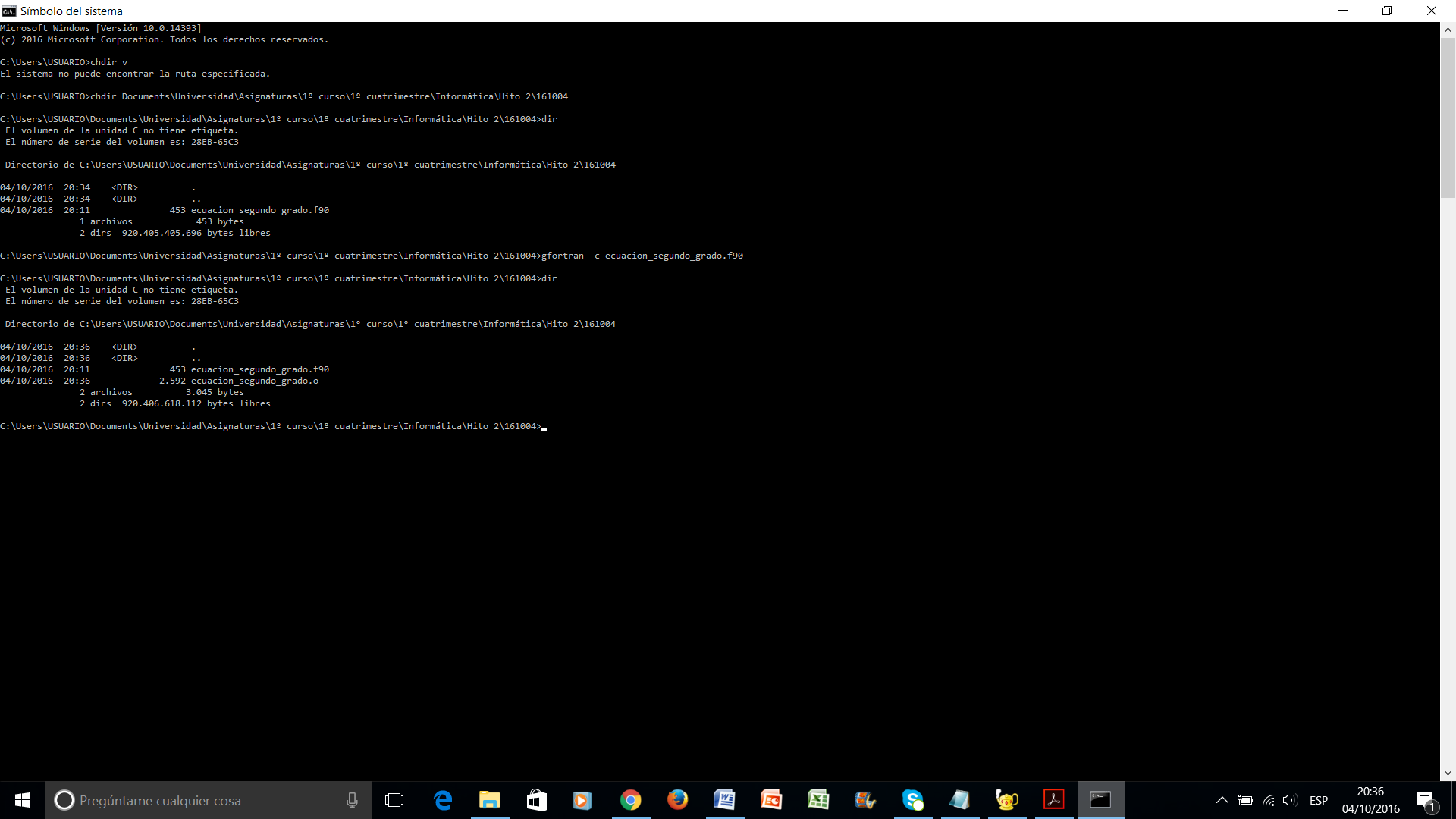
Según el programa:

En “ecuacion\_segundo\_grado”, “x1” y “x2” tomarán siempre valores reales, pues para este primer programa el discriminante de la ecuación de segundo grado será siempre a mayor o igual a cero. Si es mayor a cero, “x1” y “x2” serán variables reales diferentes. Si el discriminante es igual a cero, sólo habrá una solución para la ecuación.

En “ecuacion\_segundo\_grado\_complejos” se busca que el discriminante sea negativo, de modo que aparezca una raíz negativa. Las soluciones “z1” y “z2” son imaginarias, por tanto, variables complejas y su valor se ha de expresar dentro de un paréntesis, separando la parte real e imaginaria: (a, ±b) ≡ a ± bi.

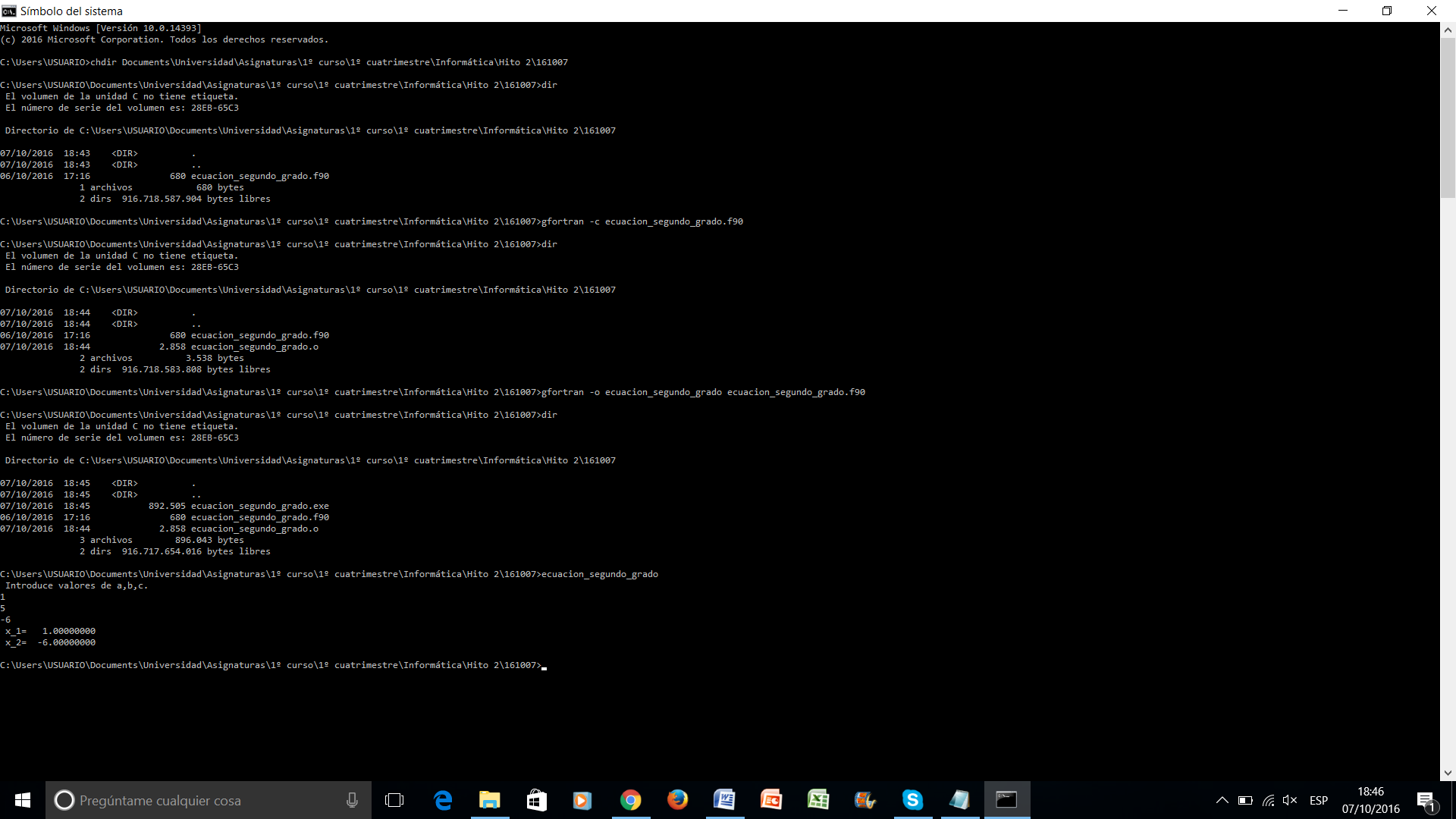
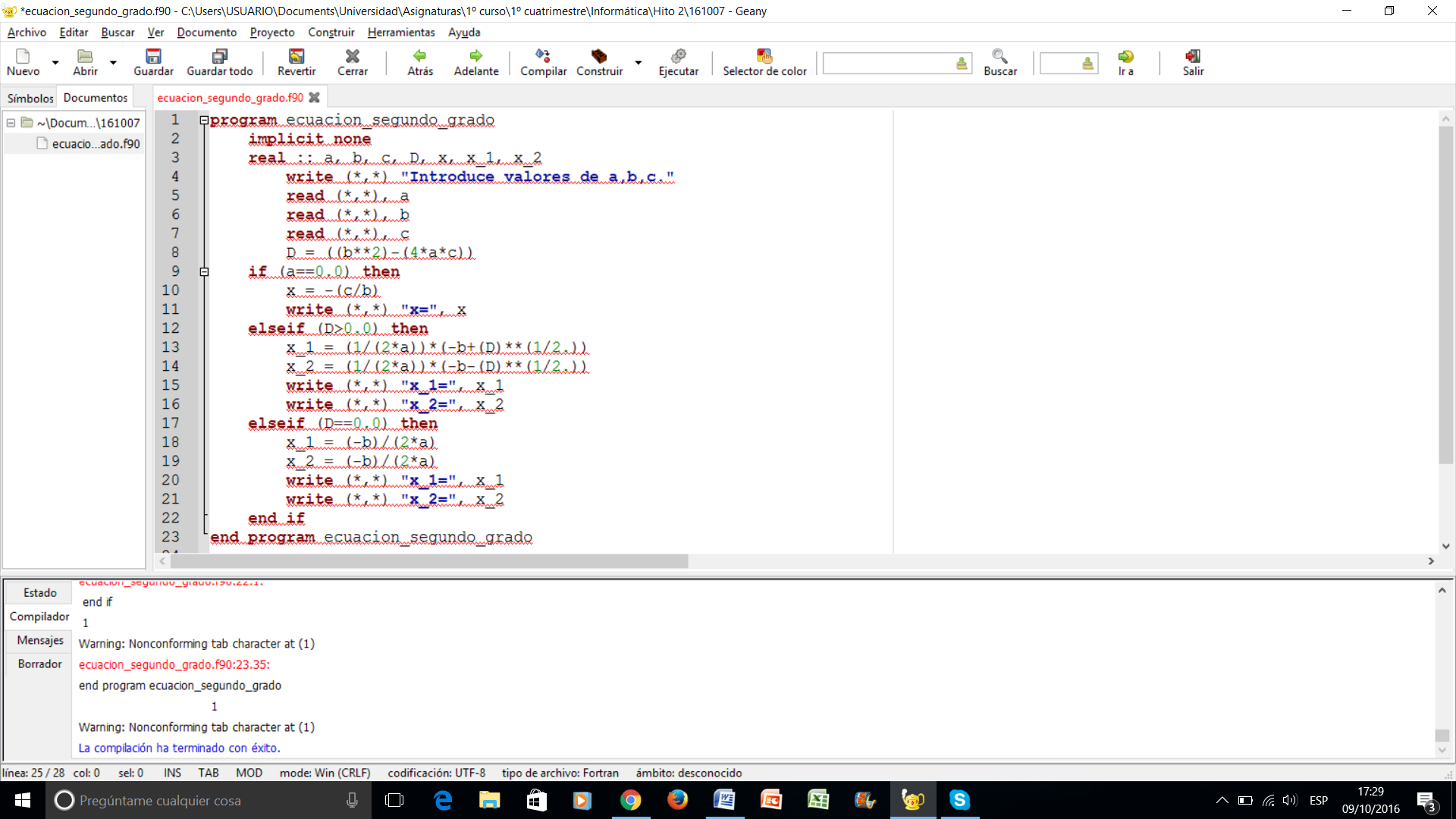
**RESULTADOS**

**Primer programa**: “ecuacion\_segundo\_grado”



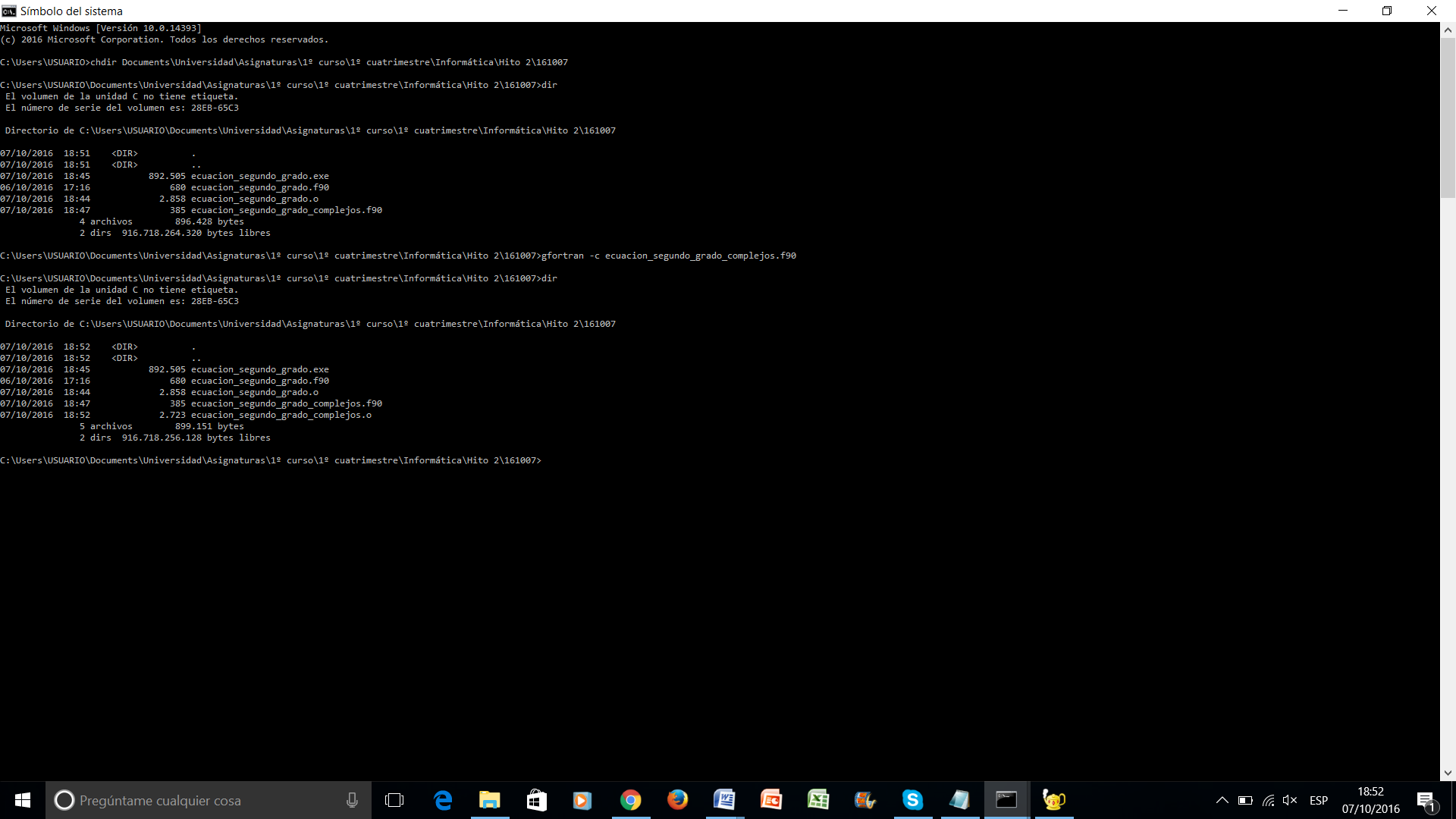
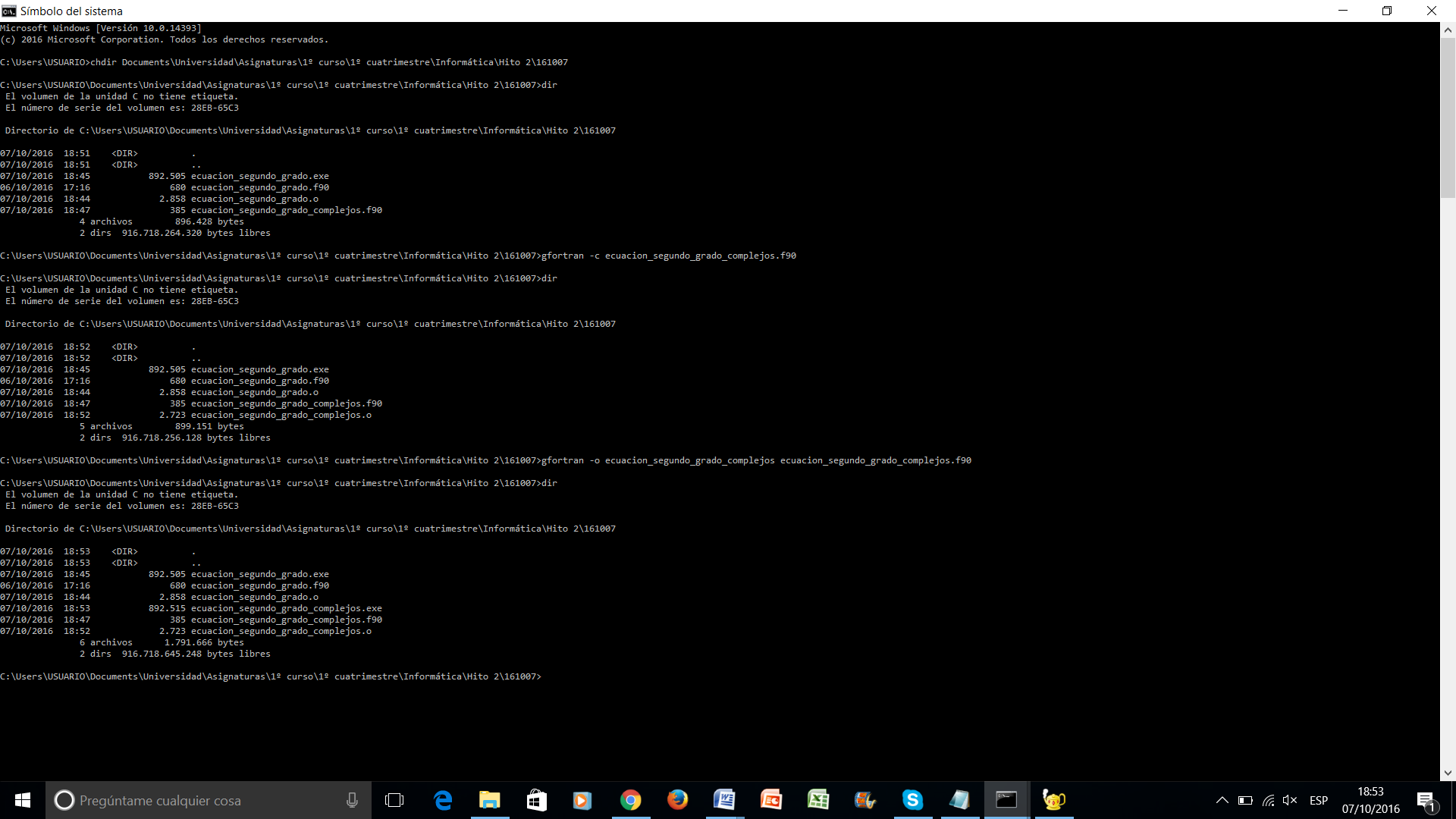
*Compilación del programa “ecuacion\_segundo\_grado” en ‘Símbolo del sistema’*

*Ejecución del programa “ecuación\_segundo\_grado” en ‘Símbolo del sistema’*



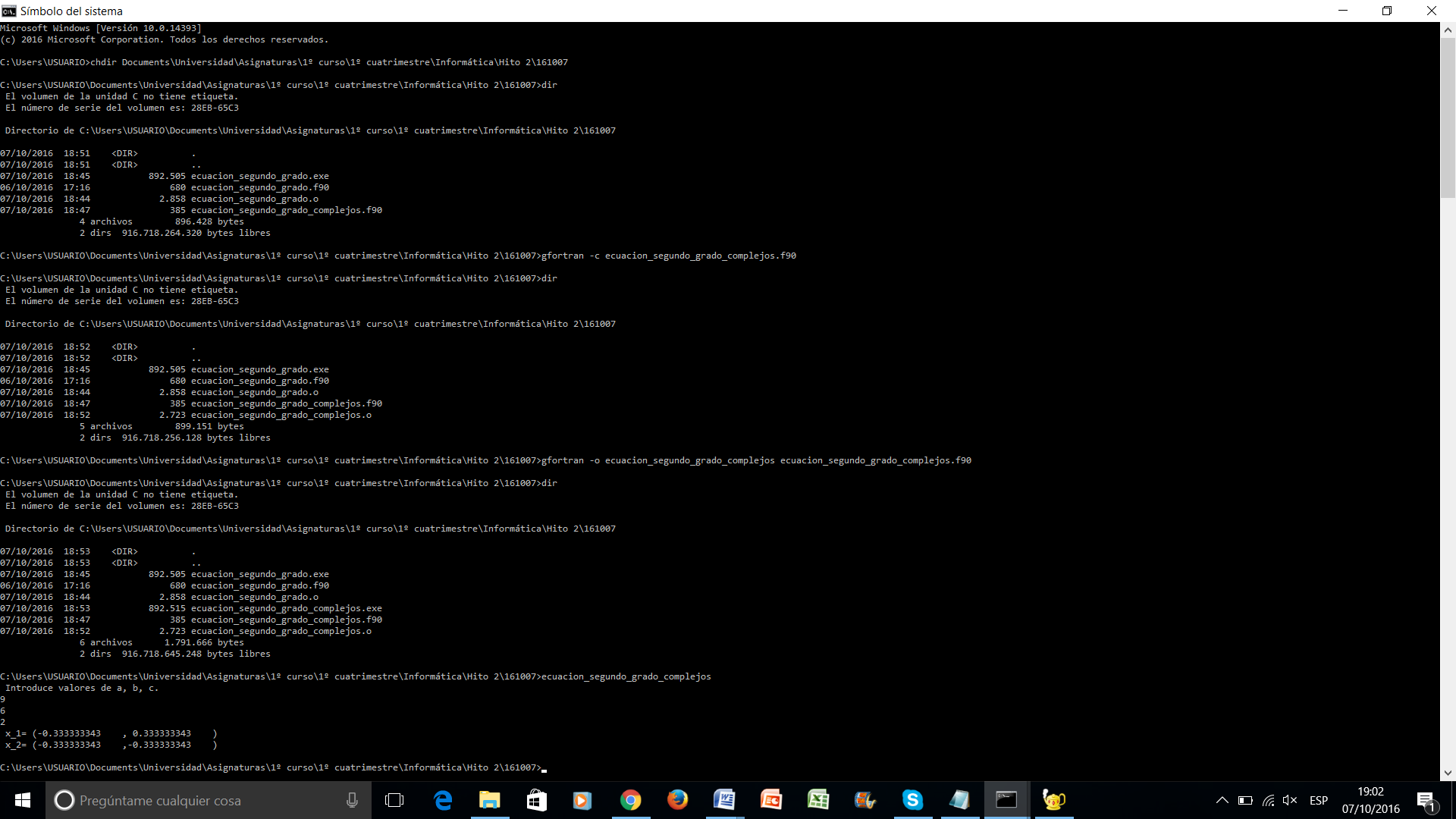
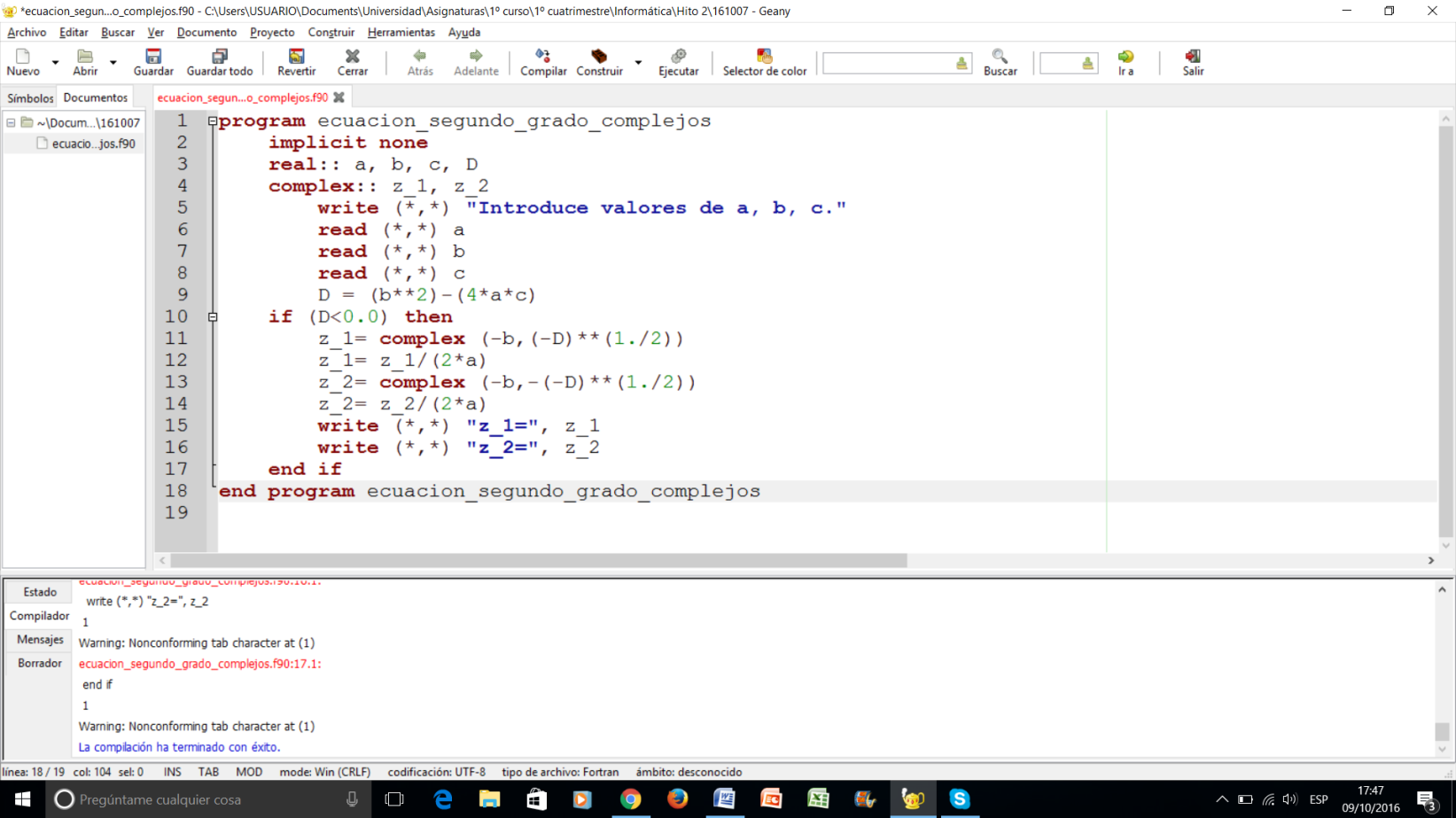
*Escritura del programa “ecuacion\_segundo\_grado” en el editor Geany*

*Prueba (satisfactoria) del programa “ecuación\_segundo\_grado” en ‘Símbolo del sistema’*

**Segundo programa**: “ecuación\_segundo\_grado\_complejos”

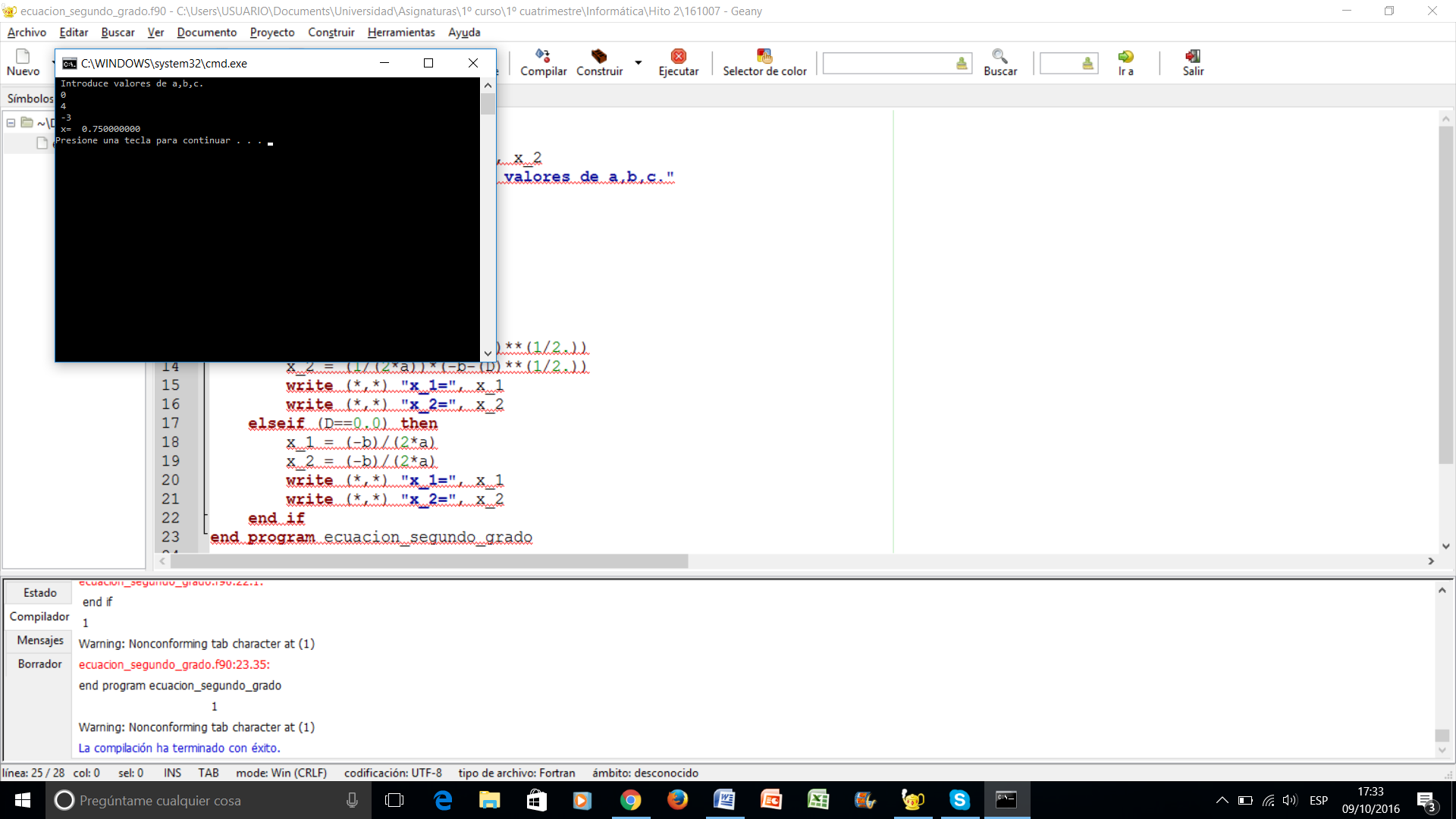
*Compilación del programa “ecuacion\_segundo\_grado-complejos” en ‘Símbolo del sistema’*

*Ejecución del programa “ecuación\_segundo\_grado\_complejos” en ‘Símbolo del sistema’*



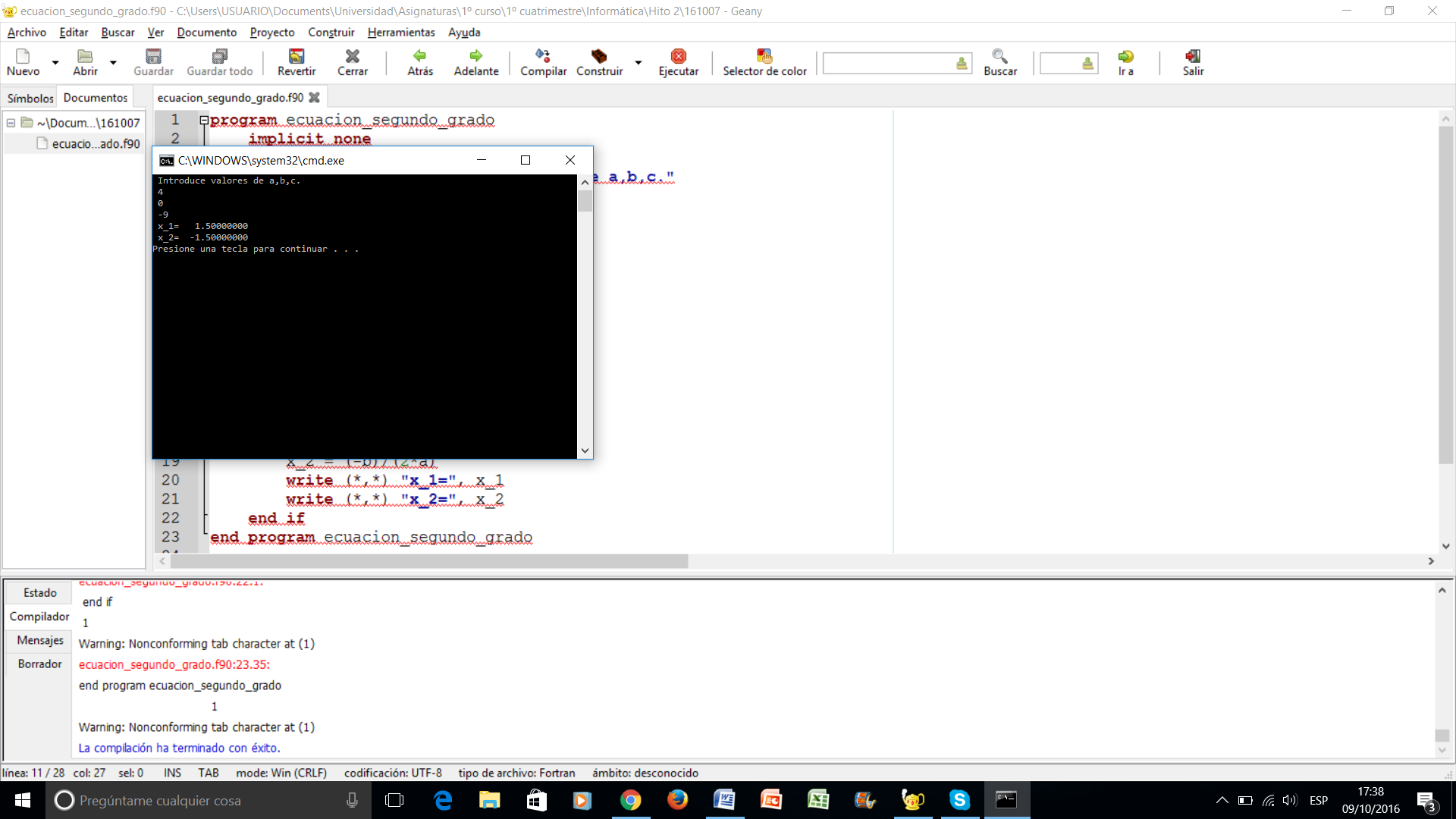
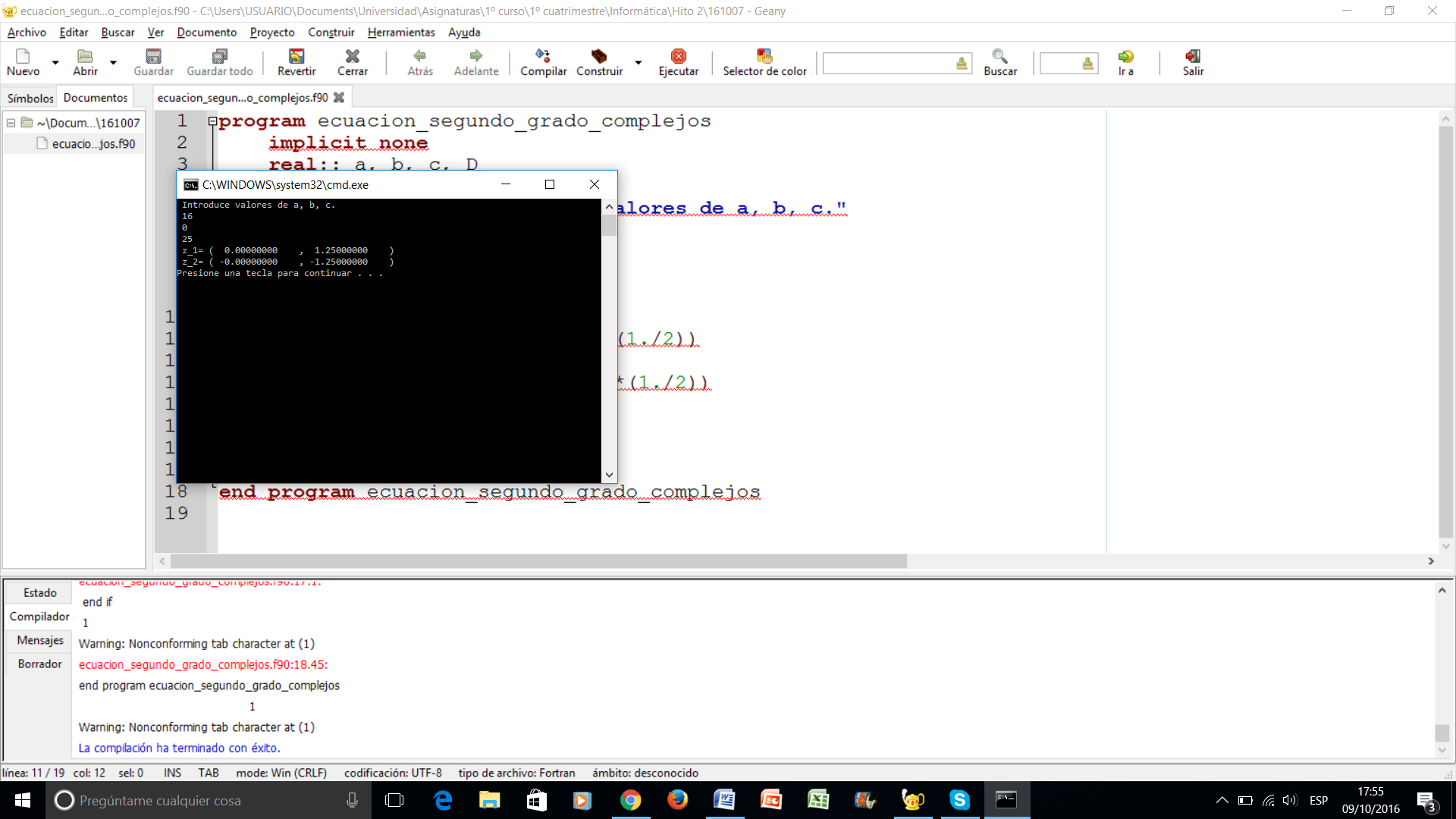
*Escritura del programa “ecuacion\_segundo\_grado\_complejos” en el editor Geany*

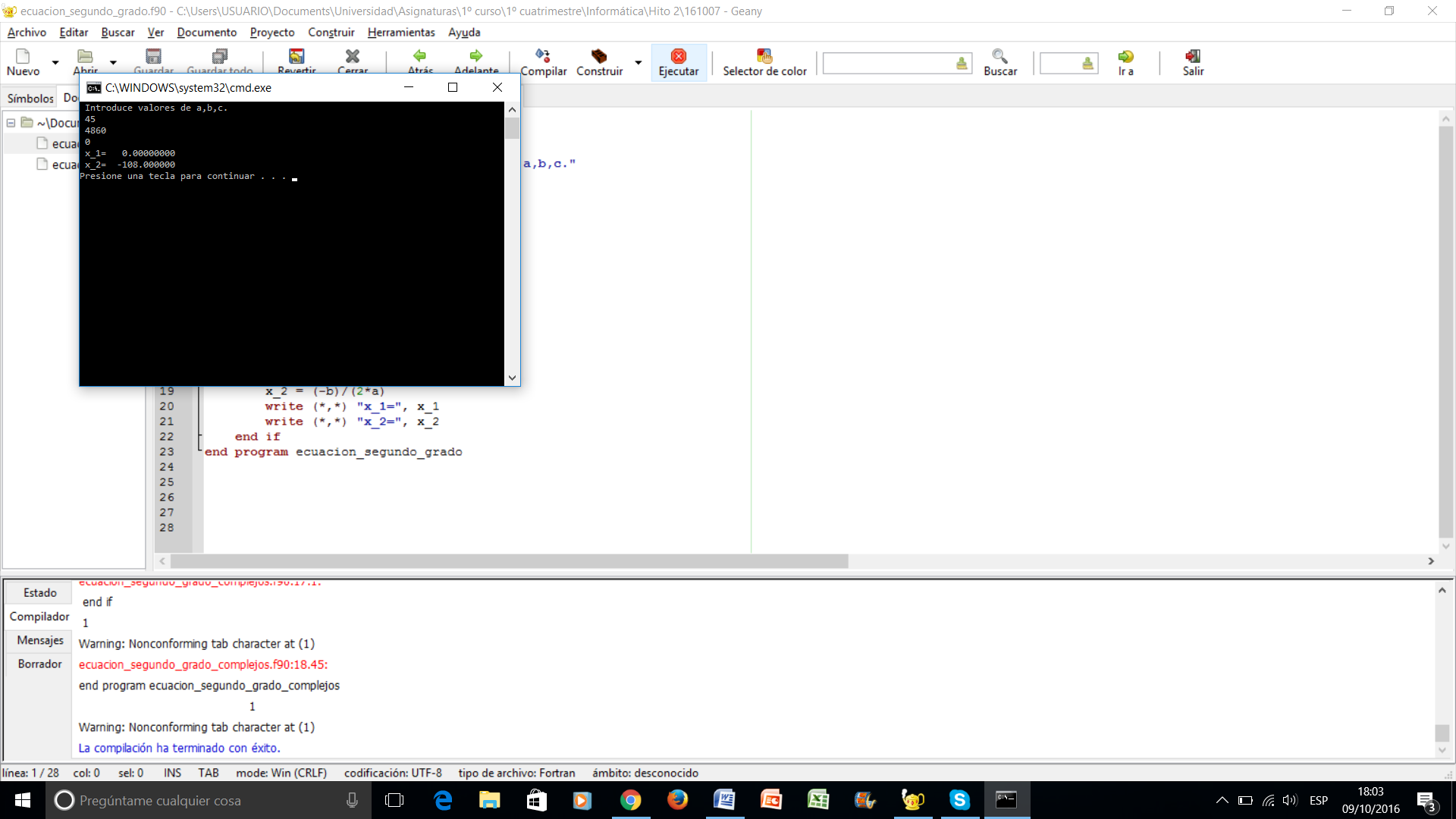
*Prueba (satisfactoria) del programa “ecuación\_segundo\_grado\_complejos” en ‘Símbolo del sistema’*

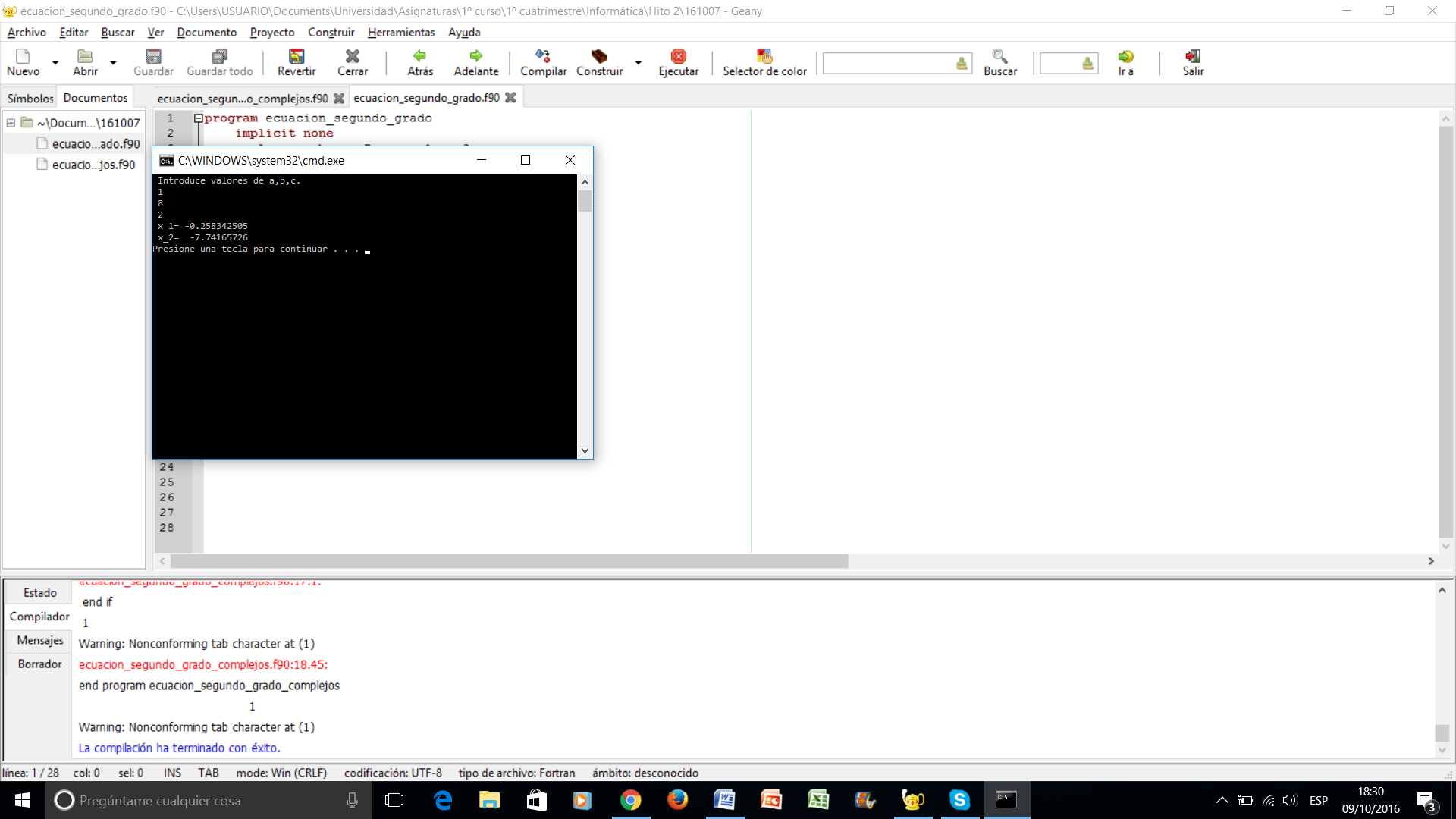
**POSIBILIDADES**

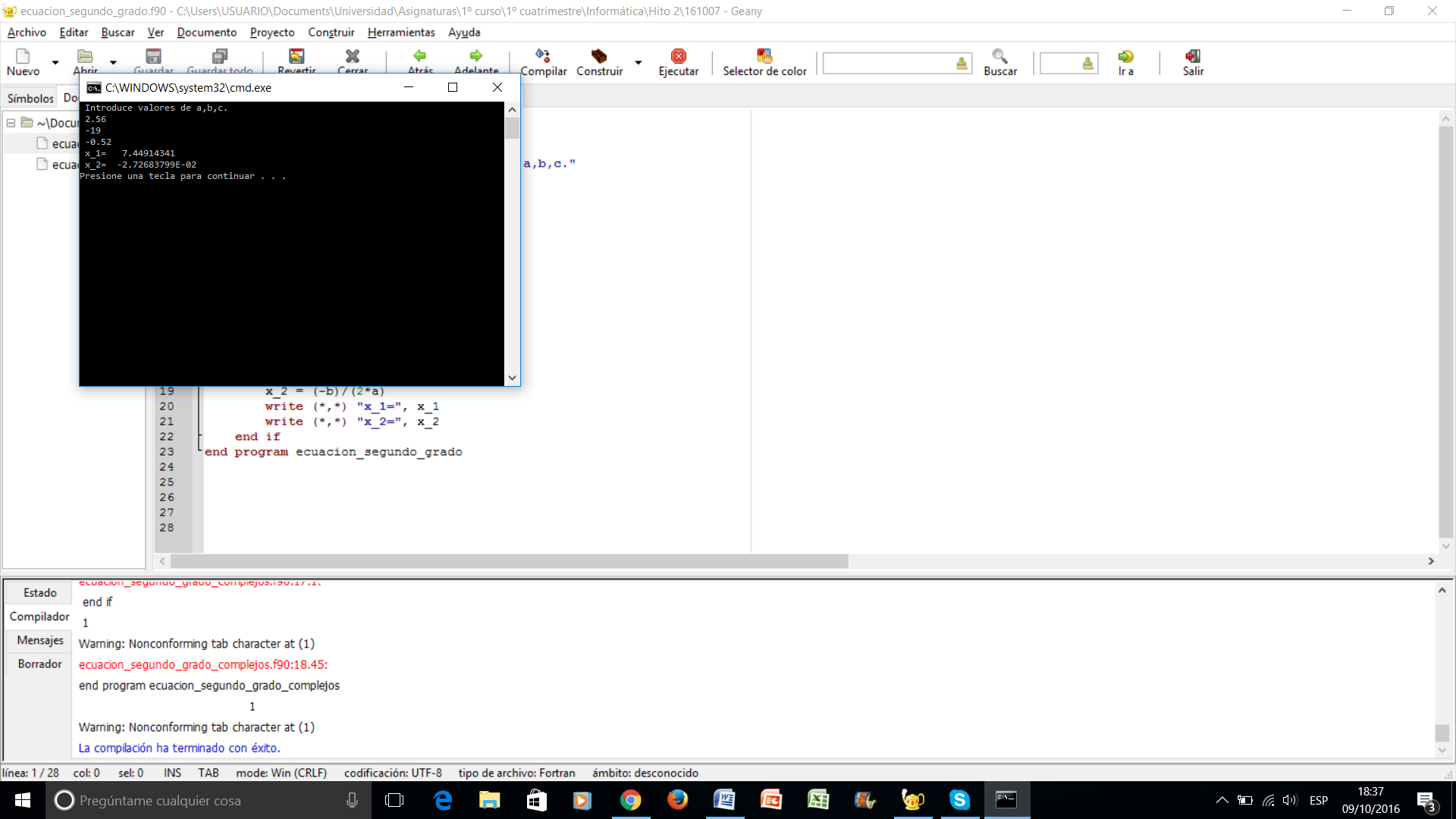


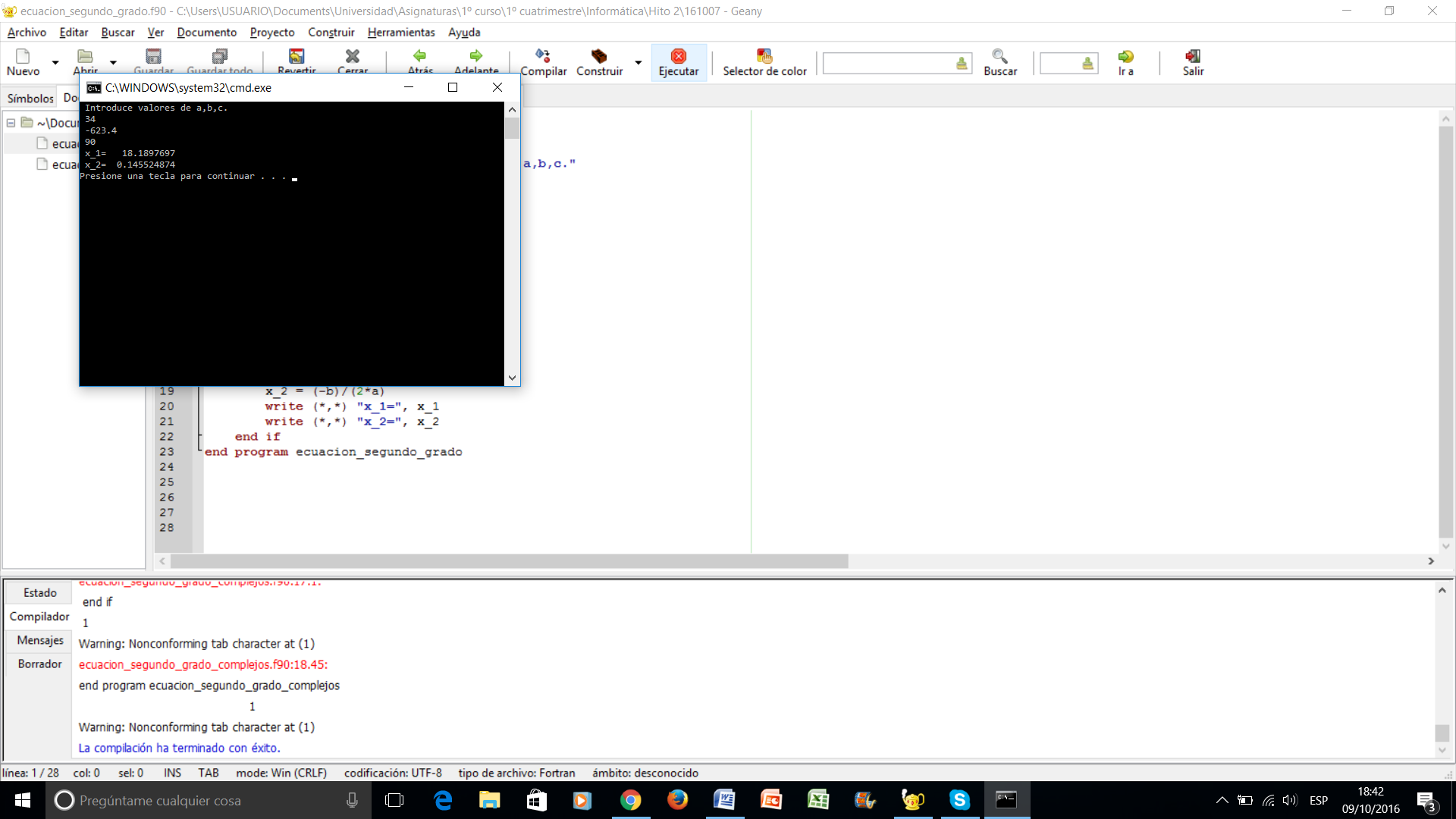
→

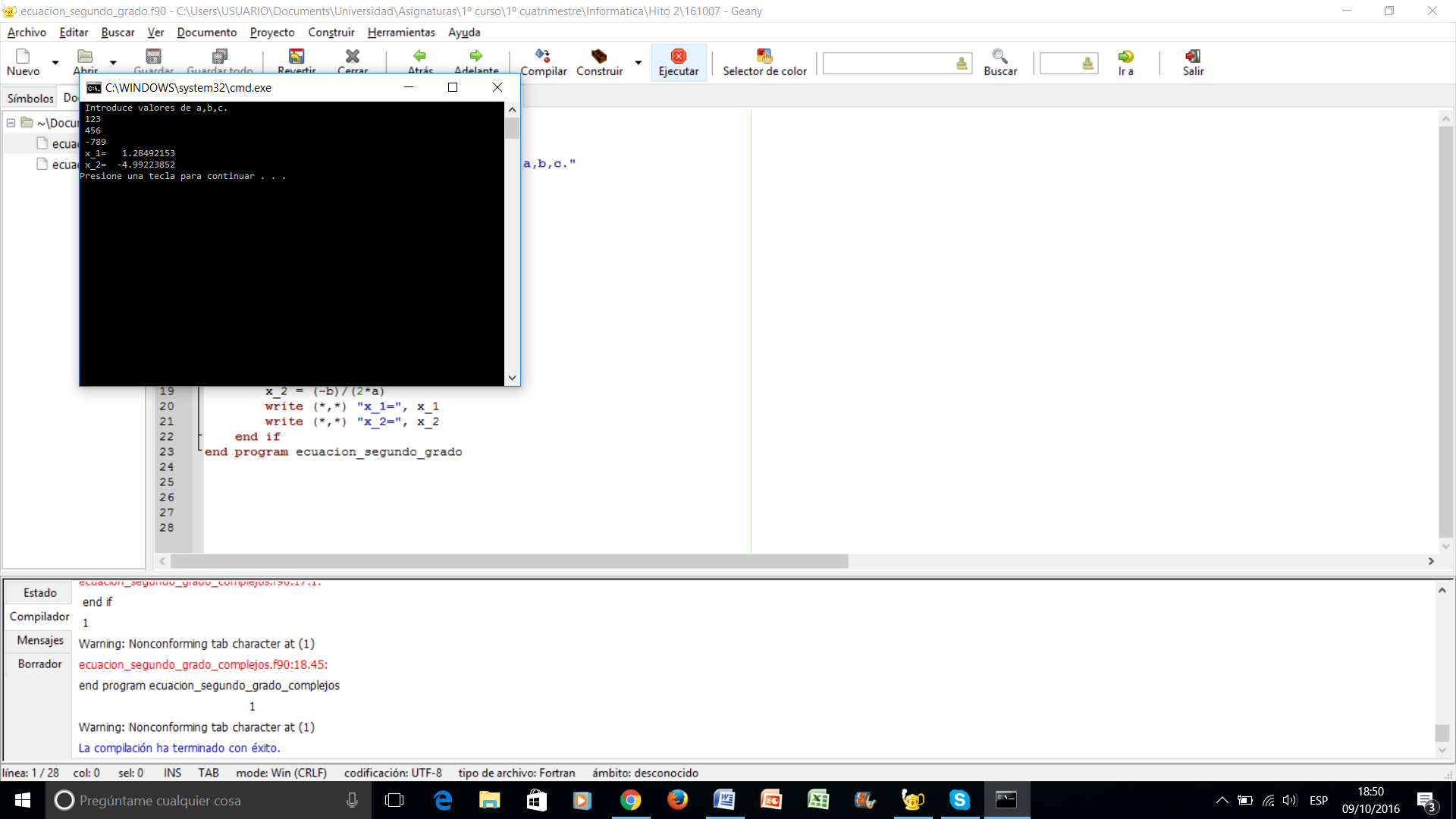
**Soluciones reales (x) si: Soluciones imaginarias (z) si:**



*  (mismo caso para coeficientes negativos).
* tal que .



*  siempre que .
* tal que .



**CONCLUSIÓN**

Con este segundo hito seguimos progresando en nuestras capacidades programadoras (de forma… ¿logarítmica?). Trabajar con variables complejas y mediante el uso de la sentencia “if” es ya un gran paso respecto al anterior trabajo, en que apenas usábamos una “character”. Aunque al principio costó, luego logramos escribir el programa en *Geany* correctamente.

Hemos de confesar que, ahora, ya manejamos bastante mejor la consola, pues, al principio, andábamos un poco perdidillos.

Por otro lado, la introducción teórica (“¿de dónde sale la expresión ?”) no resultó especialmente difícil, ya que uno de los integrantes del grupo ya conocía todo el procedimiento.

*Evaristo de Vega Galindo Yago Pego Martínez*