Memoria sobre la primera fase de HackingForce

Grupo 4 – PL4, FCR GIITIN01 (20-21)

UO283319 – Juan Francisco Mier Montoto  
UO282574 - Miguel del Riego Lázaro  
UO281892 - Ignacio Valdés Azorín

# Índice

* [Fase 1.1 – Ceremonia de iniciación](#_Fase_1.1_–)
  + [Explicación del código](#_Explicación_del_código)
  + [Ejemplos de entradas válidas e inválidas](#_Ejemplos_de_entradas)
* [Fase 1.2 – Supervivencia en código máquina](#_Fase_1.2_–)
  + Introducción general
  + [Pregunta 1](#_Pregunta_1.)
  + [Pregunta 2](#_Pregunta_2.)
  + [Pregunta 3](#_Pregunta_3.)
  + [Pregunta 4](#_Pregunta_4.)
* Reparto de trabajo

# Fase 1.1 – Ceremonia de iniciación

## Explicación del código

El código está compuesto de tres partes fundamentales:

* El receptor de errores, que recibe la cadena de error correspondiente después de que se haya incumplido una de las condiciones, la muestra por pantalla y sale del programa con ERRORLEVEL establecido a 1.
* Los métodos principales del programa, que desarrollan las tareas especificadas, incluyendo las funciones de assembly.
* El main, que se encarga de dirigir el programa y mostrar el mensaje de confirmación en caso de que no se salga antes del problema.

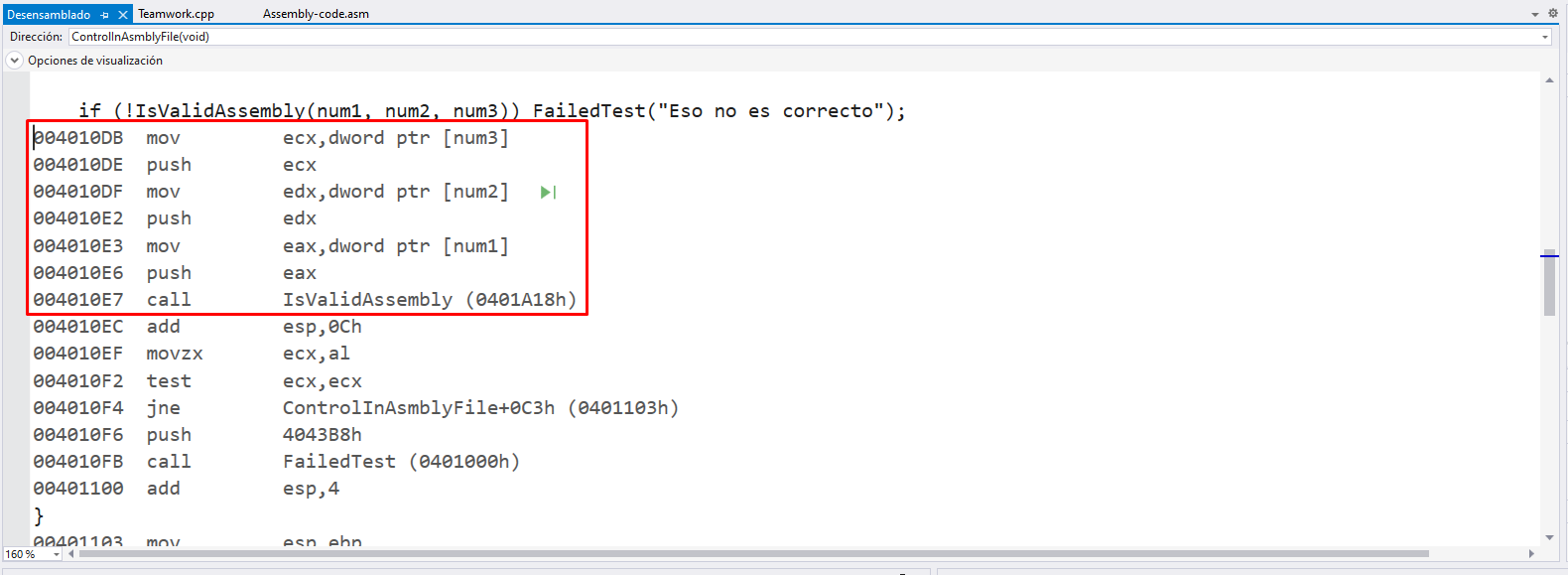
## Ejemplos de entradas válidas e inválidas

Para cada método, hay una combinación infinita de entradas válidas e inválidas. Estos son algunos ejemplos de ambos casos:

* **ControlString():**
  + En el caso de la primera cadena, solo es válida la entrada “otiPqhXcD”. Cualquier otra entrada será inválida y devolverá la cadena “Sayonara, baby” al receptor de errores.
  + En el caso de la segunda cadena, para que sea válida debe tener una longitud de al menos 6 y que el carácter de índice 3 sea igual al de índice 5, como por ejemplo “BBBaBaBB” o “f8ds0sa2d02”. Cualquier otra entrada que no cumpla estas condiciones será inválida, como por ejemplo “abcd” o “123456789”.
* **BitControl():**
  + En la primera condición, los bits entre los índices 2 y 4 (ambos incluidos) del segundo número introducido deben de ser iguales a la representación en binario de 1, es decir, 001. Algunos ejemplos de entradas válidas son “740” o “4”.
  + Para la segunda etapa, se aplican máscaras a ambos números y se combinan los resultados, de manera que el nuevo número tiene los 23 bits de mayor peso del primer número y los 9 bits restantes de menor peso del segundo. Además, el bit 6 del primer número debe de ser 1. Utilizando respectivamente los ejemplos de entradas válidas anteriores, para 740 una entrada válida sería “22080” y para 4 una entrada válida sería “3648”.
* **ControlInAsmblyFile():**
  + Para el primer número, el valor de los 8 bits de menor peso interpretados en binario debe de ser 238, como por ejemplo 240 o 61695.
  + Para los otros dos números, sus bits de la posición 8 deben coincidir, como por ejemplo 256 y 16777472 o 768 y 260.
* **ControlWithInlineAssembly():**
  + Los 16 bits de mayor peso del número introducido deben de ser iguales que los 16 bits restantes, como por ejemplo el 0 o el 143198345.

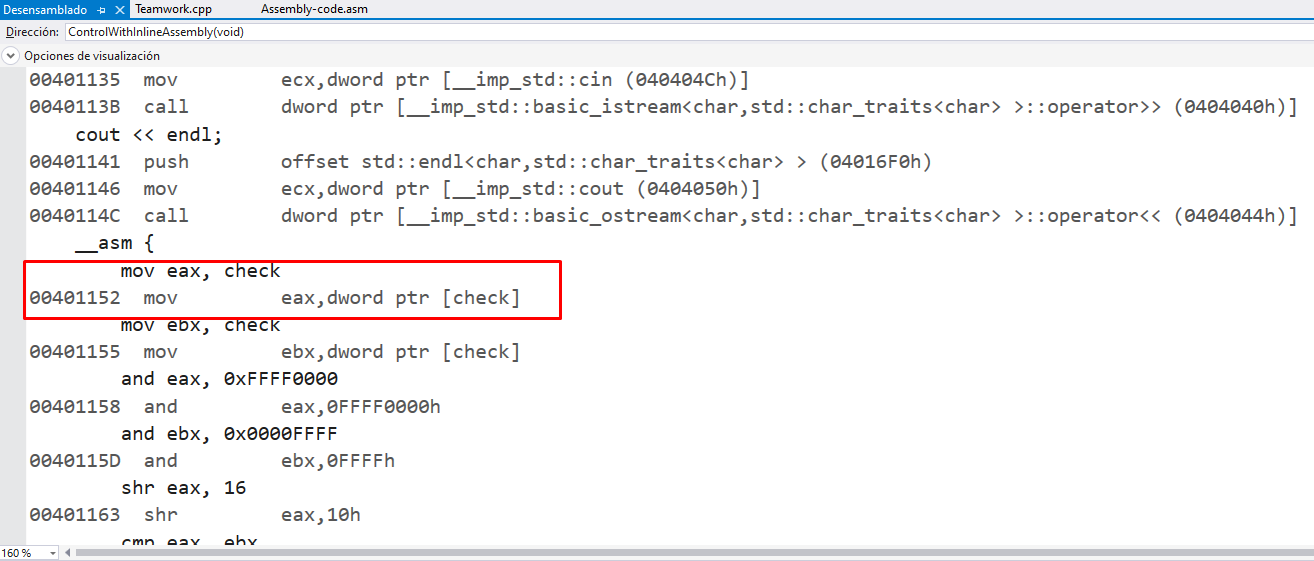
# Fase 1.2 – Salvando al mundo

## Pregunta 1.



## Pregunta 2.

## Pregunta 3.



## Pregunta 4.