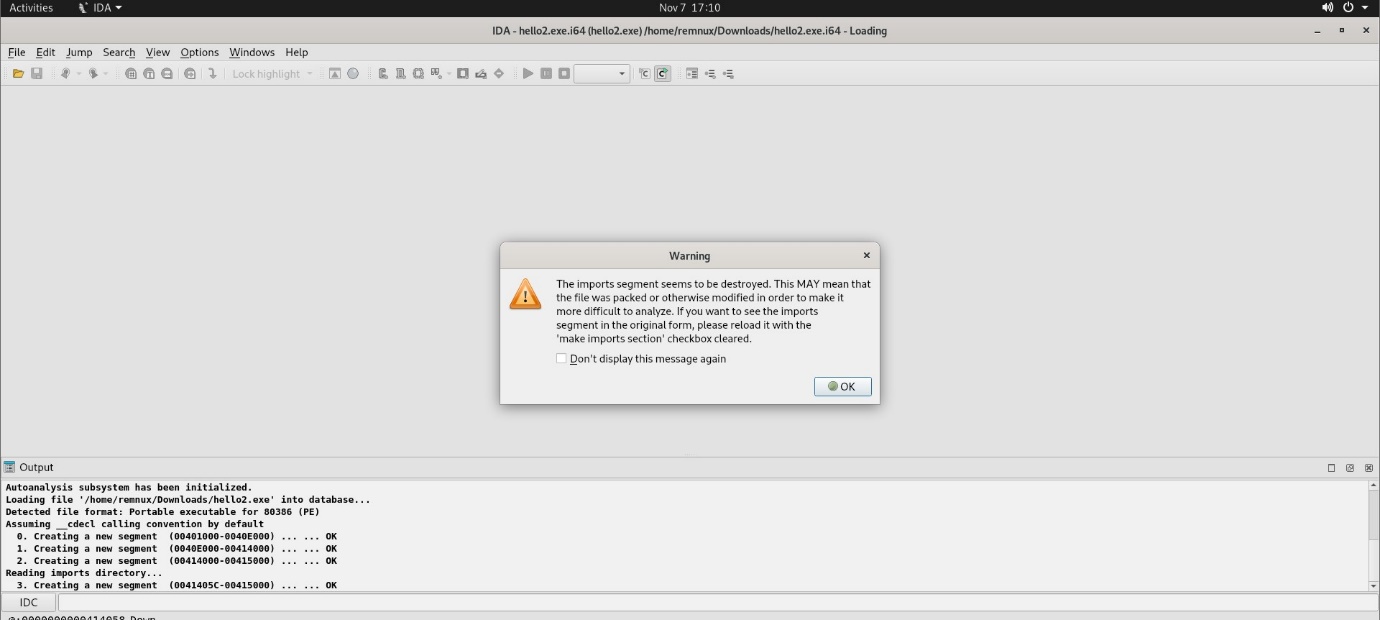
Name: Rithik Sarvesh Bharathiraja

UID: 120395246 (gbr\_2001)

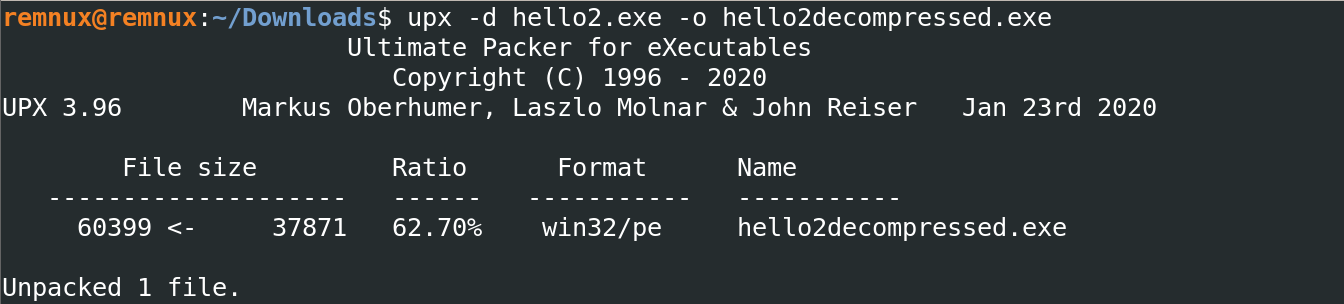
Task: Assignment-3

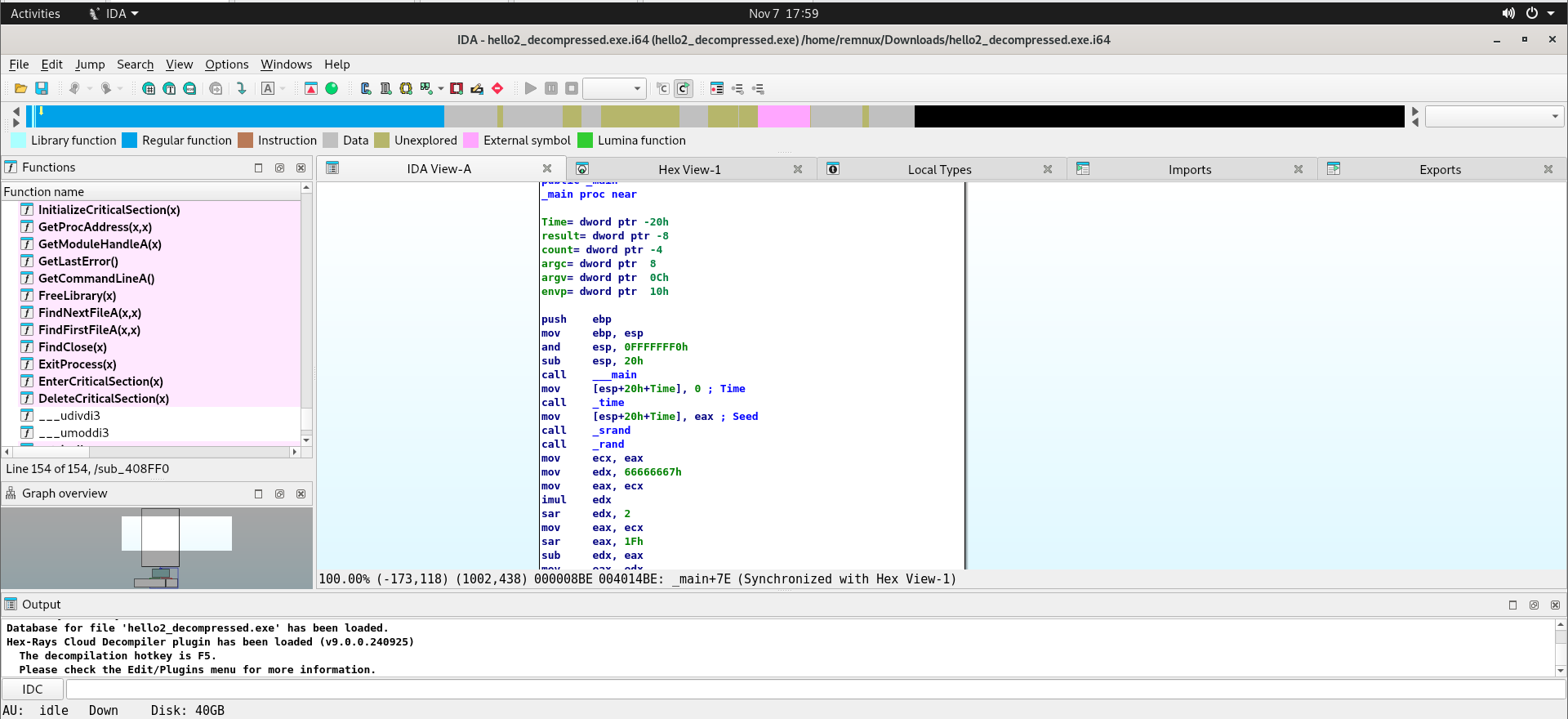
Course: ENPM696

1. While trying to load the hello2.exe file into the IDA, I encountered this alert. So, either the file is compressed or modified.  
   

When “file” command is used, we came to know that it was UPX compressed. To analyze this file, we need to decompress it first. “upx -d hello2.exe -o hello2\_decompressed.exe” command is used to decompress the file.

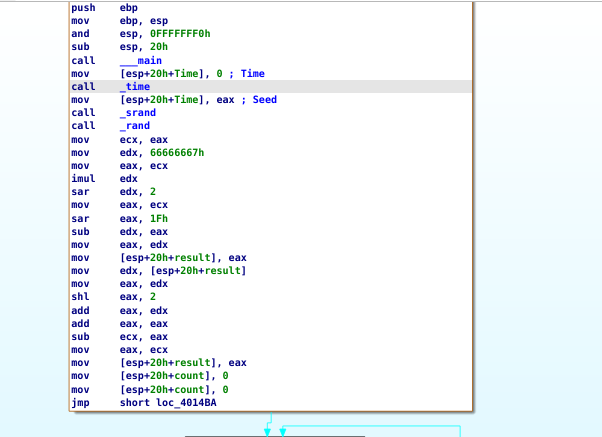




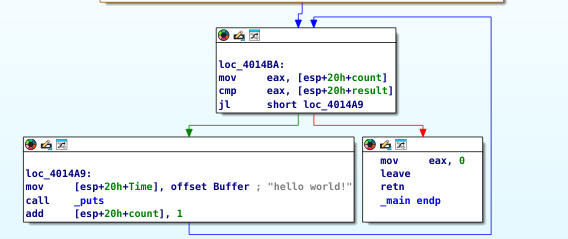
After that, we were able to analyse the file in IDA.  


1. Analysis of the behavior

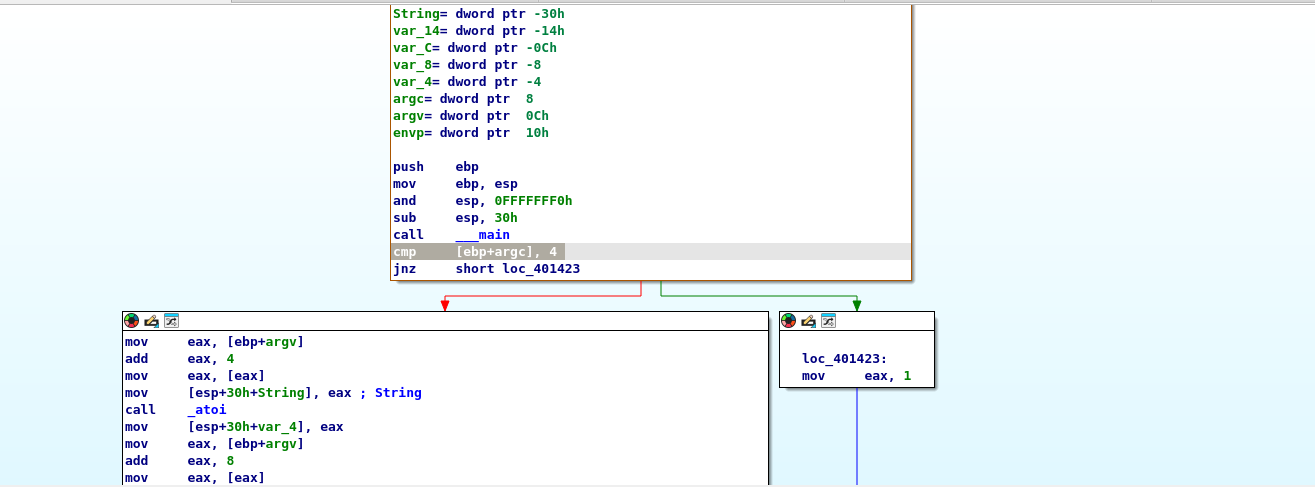
First the program calls the time() function. And then, the result is used in the srand() function to initialize the rand() with a seed (result of the time()). After that, rand() is called to generate a random number. Later, it was divided by 10. (Multiplication by 0x66666667 and then doing right shift on it by 2). Thus, the random number is generated between 0 and 9. Before proceeding, the program refines the calculation to ensure a result within 0–9. The result is stored in a local variable. And then, another local variable is initiated and stored 0 in it.

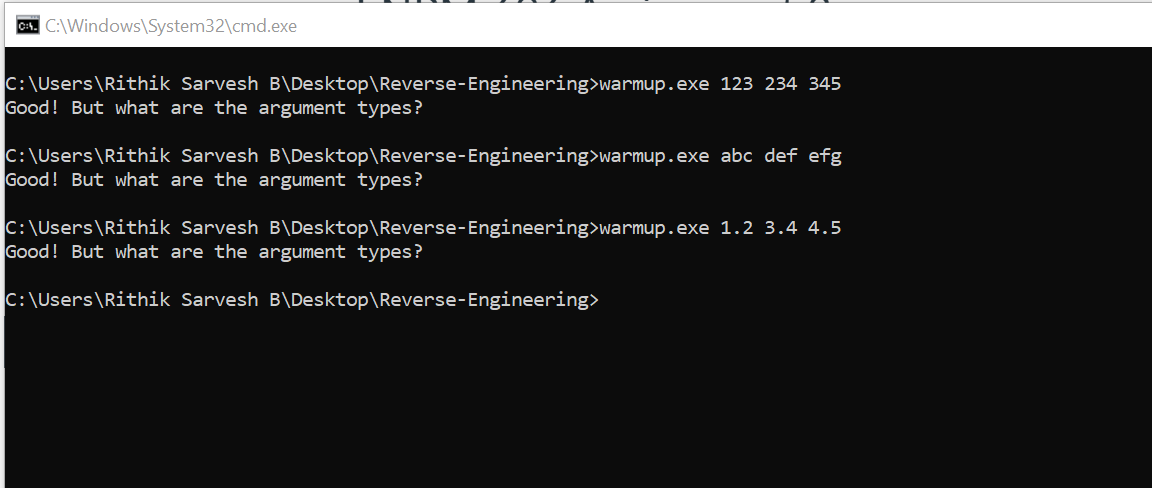


Later, the value of the newly created local variable and the result were compared. When it was lesser than the result, print of “hello world!” was done. At the end, counter variable is incremented by 1. The loop keeps on running until the counter variable matches the result variable. So, for (result-1) times, print statement is done.



1. The highlighted line of the warmup.exe says that the argc is compared to 4. Thus, 4 arguments (3 actual arguments + program name) were needed to execute this program.



Irrespective of the input types, the program prints the same same statement everytime.   


Four functions were called inside the main().   
atoi() – Converts the given string into an integer

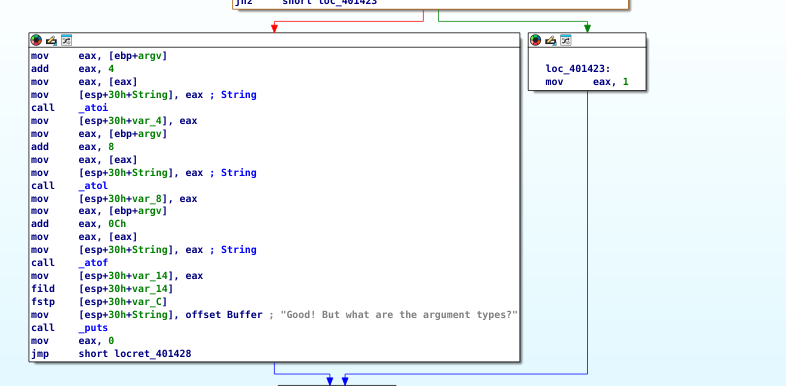
atol() – Converts the given string into a long integer

atof() – Converts the given string representation of a float number into a data of double-precision floating point value

puts() – Prints the given string.

For the first three functions (atoi, atol, atof), argv(User-defined) is used as the input. Irrespective of the input data type, these functions treat them as a character string.

For puts(), the string was already present in the program.



Flow of the program:  
1. The program checks whether there are 3 inputs. (3 user-defined input + program name).   
2. If that fails, it returns 1 and exits.

3. If the condition is true,

i. It converts the first user-defined input as an integer using atoi()

ii. And then, it converts the second user-defined input as a long integer using atol()

iii. Later, it converts the third user-defined input as a double-precision floating value using atof()

iv. At the end, the puts() is called to print the statement “Good! But what are the argument types?” and then the program exits.

4. All the converted data-types are stored in local variables but never used elsewhere.