MINISTRY OF EDUCATION AND SCIENCE OF UKRAINE

NATIONAL TECHNICAL UNIVERSITY

"KHARKIV POLYTECHNICAL INSTITUTE"

Department of Computer Engineering and Programming

«Software Means of Information Protection »

*Laboratory work report No 1*

*Topic: «* **Reversing simple programs with a password request** *»*

Student:

Group. KH 919 I.E

Ayoub El-Haddadi

Verified by:

Lecturer Viktor CHELAK

Kharkiv – 2020

***Purpose of work***:

To acquire practical skills in writing and applying password verification and password correction programs in exe files in the masm64 environment.

***Individual task:***

Variant 8:

8. A 4 x 6 matrix is specified. Define a row with the minimum sum of elements. (Password verification using basic commands).

**Algorithm of the program**

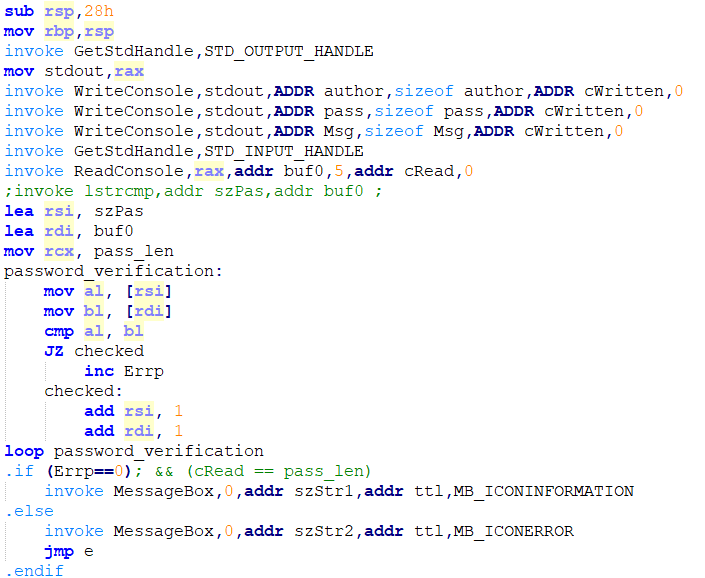
# .data section of the program



In this section, we initialize all variables and templates that we are going to use in the .code section, (Matrix, author, password, strings, length…)

# .code section of the program

Password verification:

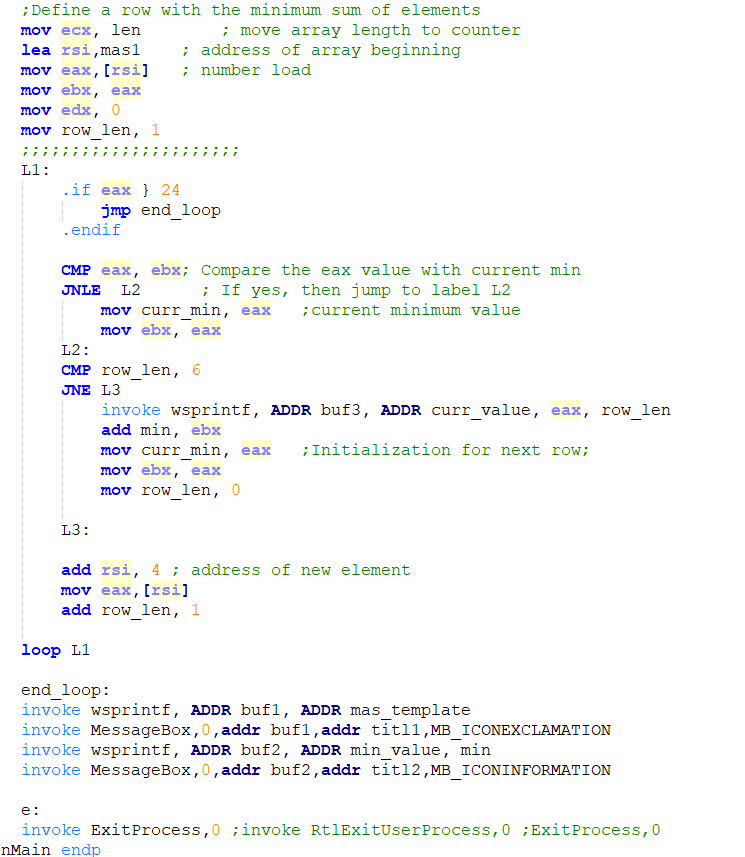


The first part of this section is for password verification, for this purpose I’ve used “Password verification using basic commands” as requested, we compare the inputted password with the correct password character by character and increase “Errp” variable if there are not the same in a loop. Next step is to check if this variable is equal to “0” or not if it is that means that the inputted password is correct.

If password is not correct we jump to the exit bloc named “e” using “jmp e”.

If password is correct we invoke a message box with success message.

Define a row with the minimum sum of elements.



This is the second part of .code section, where we calculation sum of minimum element of each row.

For that purpose, we initialize a “curr\_min” variable with the first element of the array, and, we are going to go into each element of the matrix and compare it to “curr\_min” variable if that element is lower that “curr\_min” it takes the new value.

And, after going through 6 elements that means that we have completed a row, and, we add “curr\_min” to “min” variable which holds the sum, and move first element of the next row to “curr\_min”, and repeat until we go through all elements.

Next steps are invoking two message boxes first with the matrix used in this program, second is the result.

**Source Code**

Full source code of this lab you can find it in:

[**https://github.com/Elh-Ayoub/RP\_Labs/tree/main/lab1**](https://github.com/Elh-Ayoub/RP_Labs/tree/main/lab1)

**Results of the program:**

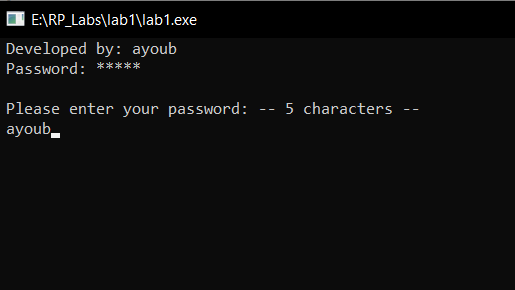


Figure 1 – Request of input password

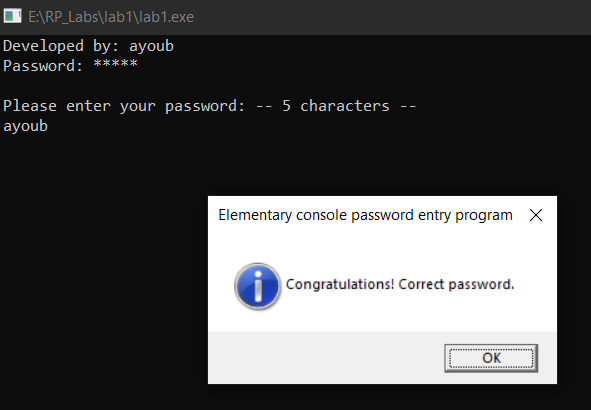


Figure 2 – Result of entring correct password

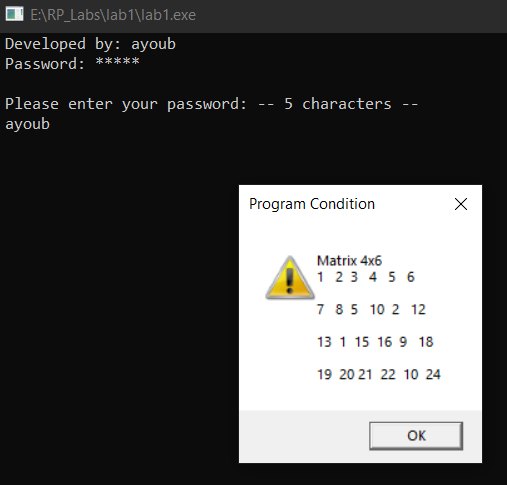


Figure 3 – Matrix used in the program

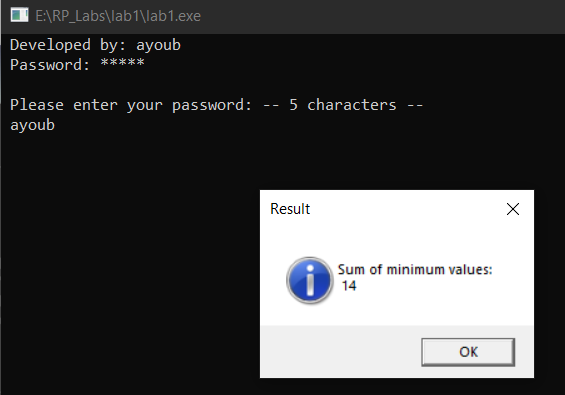


Figure 4 – Sum of minimum value

**Conclusions:**

As a result of laboratory work we gained a practical skills in writing and applying password verification and password correction programs in exe files using Masm64 environment, and using x64dbg to patch and modify exe file.

**You can also find this lab in:**

<https://github.com/Elh-Ayoub/RP_Labs/tree/main/Docs>