**EEE206 PROGRAMMING**

**PROJECT-II REPORT**

Submitted by:

*Enis Ata Erkol (EEE)*

*Ahmet Berke Karaca (EEE)*

*Umut Sedefoğlu (EEE)*

Project Mentors:

Doğan Çörüş

Faculty of Engineering and Natural Sciences

Kadir Has University

Spring 2022

**TABLE OF CONTENTS**

1 INTRODUCTION 3

2 CODE STRUCTURE 4

2[.1. Code Level 1 4](https://euc-word-edit.officeapps.live.com/we/wordeditorframe.aspx?ui=tr%2DTR&rs=tr%2DTR&wopisrc=https%3A%2F%2Fstukhasedu-my.sharepoint.com%2Fpersonal%2Fe_erkol_stu_khas_edu_tr%2F_vti_bin%2Fwopi.ashx%2Ffiles%2F5f1ef8de104442bba60a0858d4741e64&wdenableroaming=1&wdfr=1&mscc=1&wdodb=1&hid=72F036A0-0029-4000-34B2-37B558977F64&wdorigin=Other&jsapi=1&jsapiver=v1&newsession=1&corrid=72b67464-7440-4f65-8496-2215fc2f5739&usid=72b67464-7440-4f65-8496-2215fc2f5739&sftc=1&mtf=1&sfp=1&instantedit=1&wopicomplete=1&wdredirectionreason=Unified_SingleFlush&rct=Medium&ctp=LeastProtected#_Toc87804110)

2.2. Code Level 2 5

2.2.1 First Step6

2.2.2 Second Step7

2.2.3 Third Step8

3 [SIMULATION RESULTS 12](https://euc-word-edit.officeapps.live.com/we/wordeditorframe.aspx?ui=tr%2DTR&rs=tr%2DTR&wopisrc=https%3A%2F%2Fstukhasedu-my.sharepoint.com%2Fpersonal%2Fe_erkol_stu_khas_edu_tr%2F_vti_bin%2Fwopi.ashx%2Ffiles%2F5f1ef8de104442bba60a0858d4741e64&wdenableroaming=1&wdfr=1&mscc=1&wdodb=1&hid=72F036A0-0029-4000-34B2-37B558977F64&wdorigin=Other&jsapi=1&jsapiver=v1&newsession=1&corrid=72b67464-7440-4f65-8496-2215fc2f5739&usid=72b67464-7440-4f65-8496-2215fc2f5739&sftc=1&mtf=1&sfp=1&instantedit=1&wopicomplete=1&wdredirectionreason=Unified_SingleFlush&rct=Medium&ctp=LeastProtected#_Toc87804121)

4 CONCLUSION 18

**1 INTRODUCTION**

This project aims to create a code that can be used as a voting system. The main idea of the code is to enter, as a user, with the user id and password given and defined in the code. Thus, it should also present the results when the voting comes to an end. However, after the termination of level two code it is been crucial to change some of the scenarios, such as adding more options to the voting as abstain or adding an option which results in “quitting” of the voting and such. As an end of the code, after user enters their user id and password with their choice about election, user gets the message about sticking with the decision of the user or choosing some other option. Afterwards, when all these actions have been made the code’s termination stop and gives the voting results just as described in the project description.

**2 CODE STRUCTURE**

We decided to explain the two levels of the code in different headings. However, the level one code is the foundation of the code itself while the level two code might be called as the detailed version of the level one and with different aspects also added.

**2.1 Code Level 1**

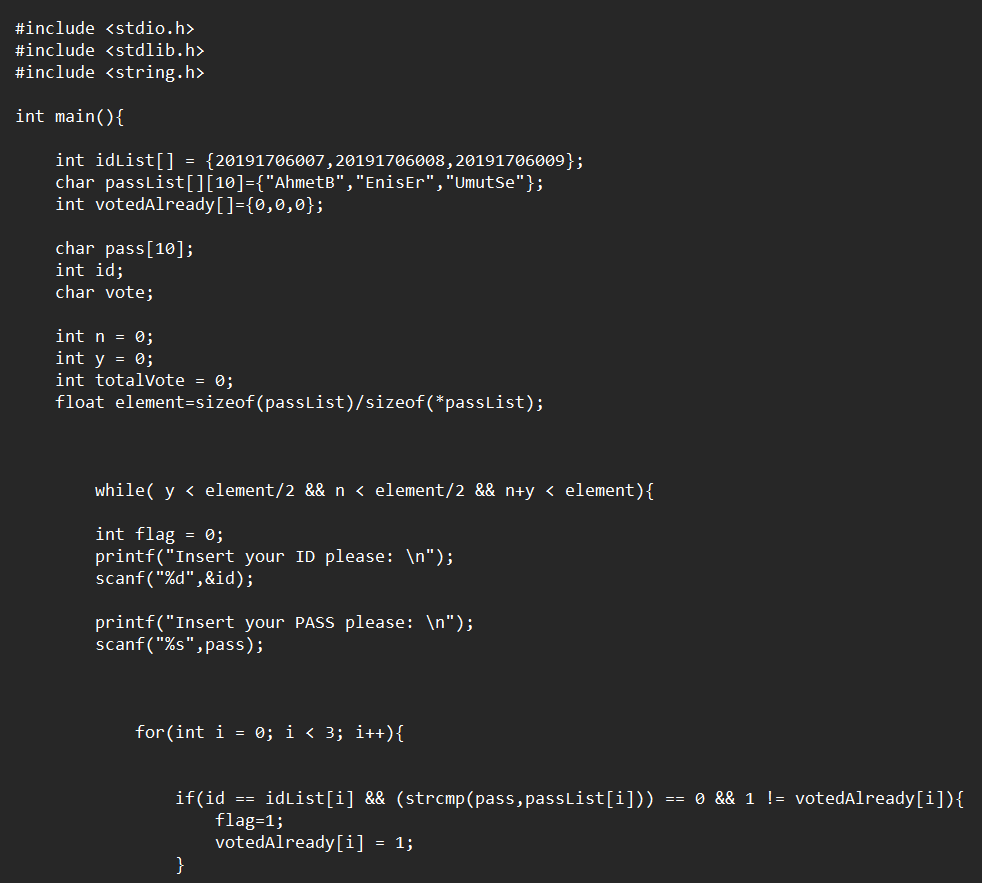


Figure 1.1: Level 1 code - first part

In this part of the Level 1 code, We started by entering our data. First of all, we entered our ID List that we created with our own numbers, then we created the data that receives the password with "char pass". (Max 10). Afterwards, we provided another data entry called "VotedAlready" to ensure that those who voted do not vote again. We also set the "yes/no" number and "TotalVote" data to be entered later to 0. After that, we created another data that we called "element" and made it equal to the number of people in the ID List. We also set the "yes/no" number and "TotalVote" data to be entered later to 0. After that, we created another data that we called "element" and made it equal to the number of people in the ID List. Then we created a while loop, it checks for any omissions that will be in the vote. The for loop just below allows us to check the inputs of the users.

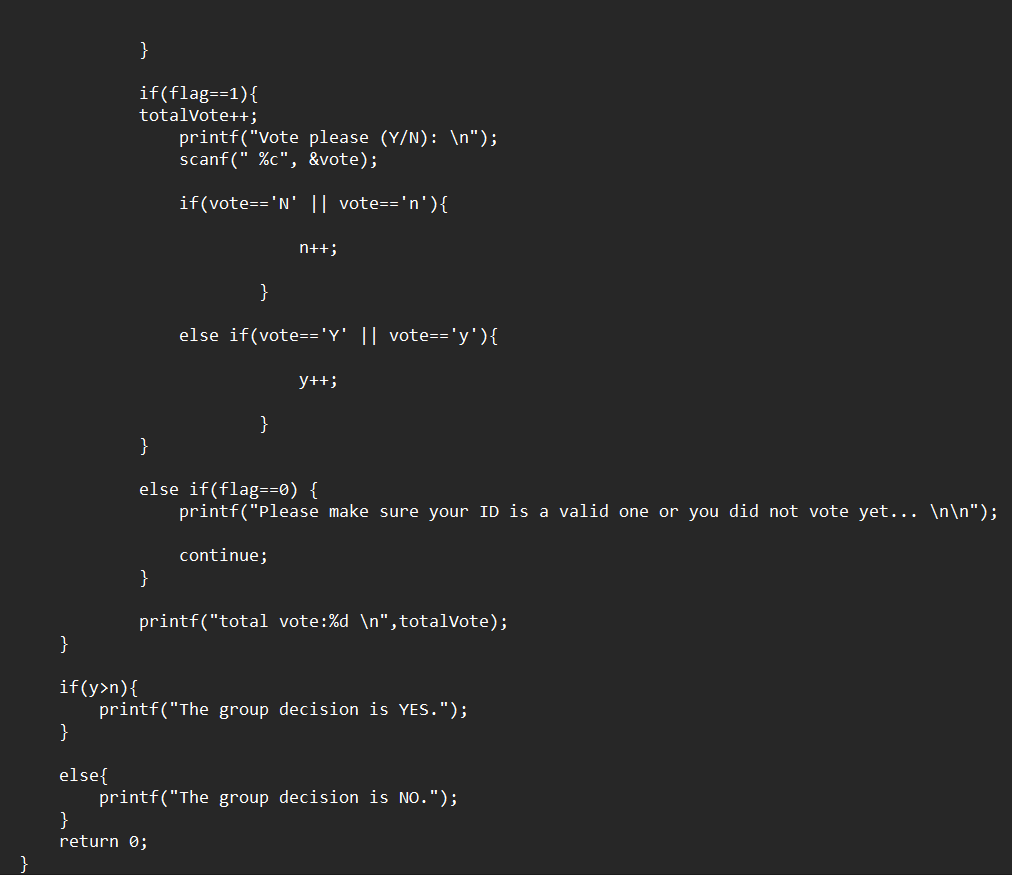


Figure 1.2: Level 1 code – second part

At the end of the part we wrote for Level 1, we now collect our votes from the users and decide that the result is "Yes/No" according to the majority of the votes.

**2.2 Code Level 2**

The second level code will have the two different header files, “stdio.h” and “stdlib.h”. “Stdio.h” is the header file we need for the standard issue input output, while “stdlib.h” is the header for the memory allocation, process management, conversions, and other functions are included in the C programming language's general-purpose standard library.

**2.2.1 First step**

The main structure for the code starts with defining the inputs and outputs. The first step, the code will be explained in this part step-by-step in order to eliminate any complications, taken described in the above can be seen in under:

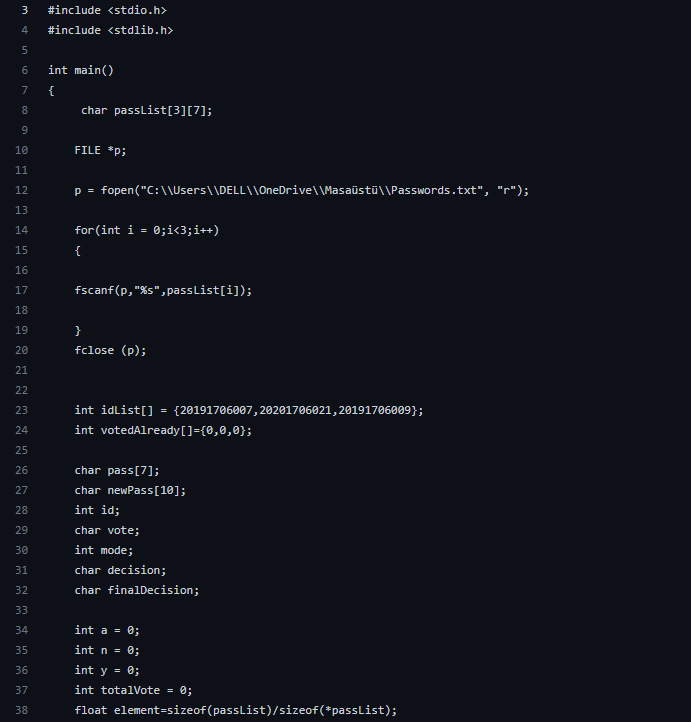


Figure 2.1: First step taken in level 2 code

Main idea of the first step, as have been mentioned in above, to define the inputs and outputs of the whole code and to define the password list already existing in the main computer by using the specified part in the first step illustrated in under:

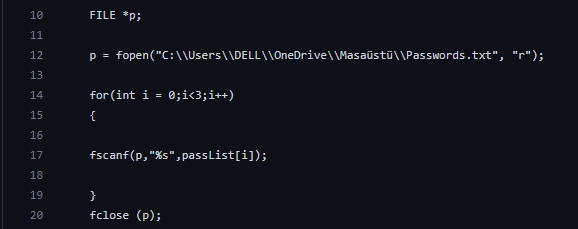


Figure 2.2: Password list definition placed in the first step

**2.2.2 Second step**

In the second step of the code level two we tend to create an while statement for the password and id entering interface. This interface will help the user to type their password and user id and the code will apply a for loop after guaranteeing the correct user id with correct password. This for loop is essential mainly for the mode selection, which is the next interface after the user id and password interface. The illustration under presents the access granting and mode selection part of the code:

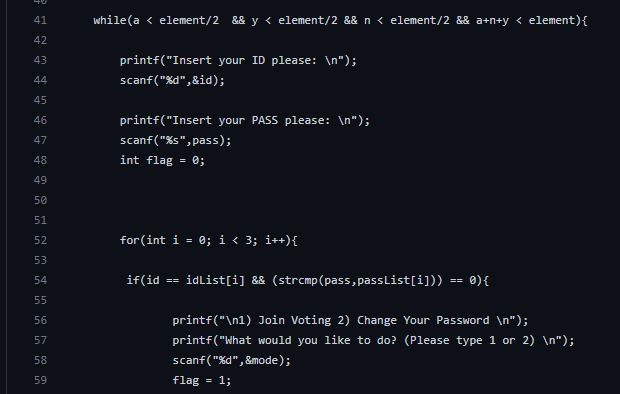
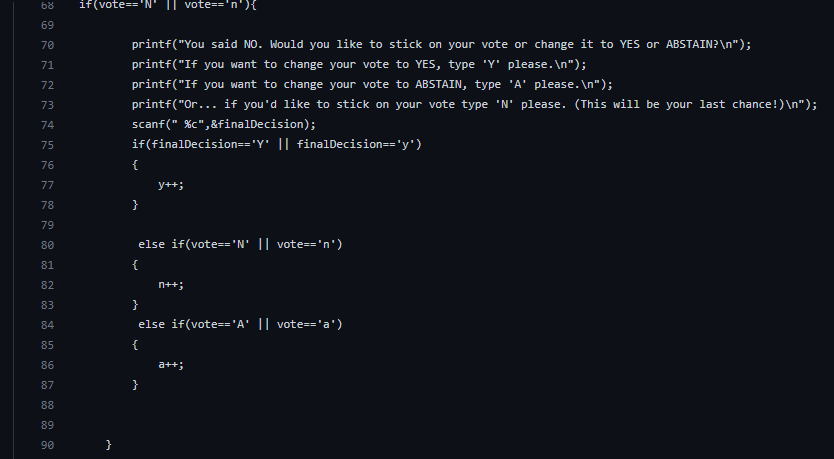
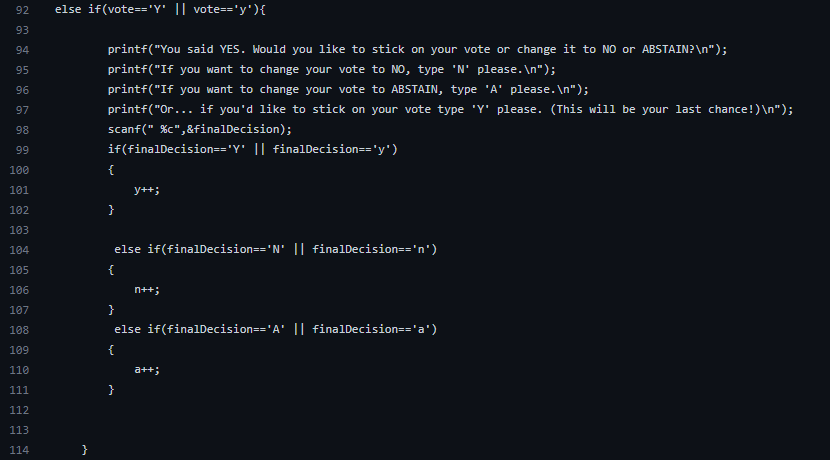


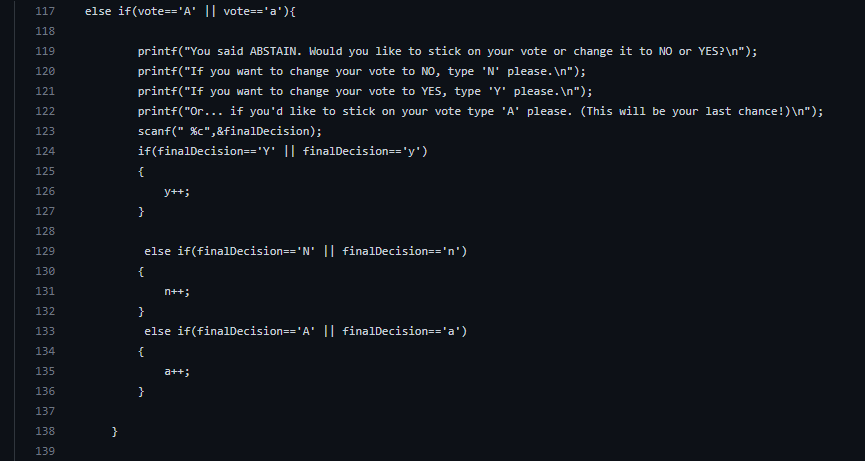
Figure 2.3: Comparison and mode selection part

**2.2.3 Third step**

The third and the crucial part of the code is about if, else and if else statements which essentially will power up the vote selection process. The illustrations under presents the yes, no and abstain voting codes which adds one more point to the Y, N or A vote count:

Figure 2.4: ’No’ decision-making part

Figure 2.5: ’Yes’ decision-making part

Figure 2.6: ’Abstain’ decision-making part

Other crucial part of the third step and decision-making process is that the triple if-else statement placed in the decision-making parts. These triple statements are the place where the final decision integer gets plus one and expressed to the system. After the decision-making has been made, the mode selection part comes forward. In the mode selection part, we tend to choose whether we are going to change our password and change it in the password list or if the user tries to vote again give a warning message to the interface as shown in the above:

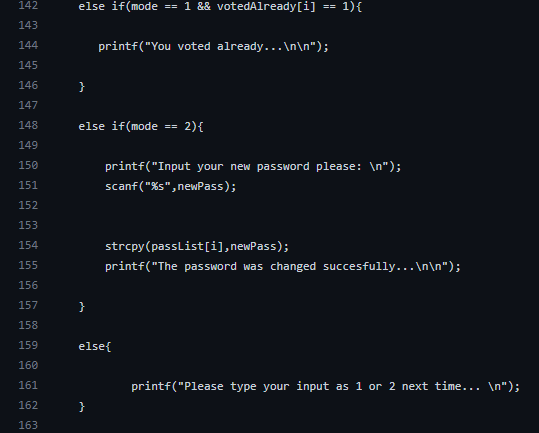


Figure 2.7: Mode selection and “voted already” warning

Lastly, we tend to express the final vote values by the part available in the under illustration:

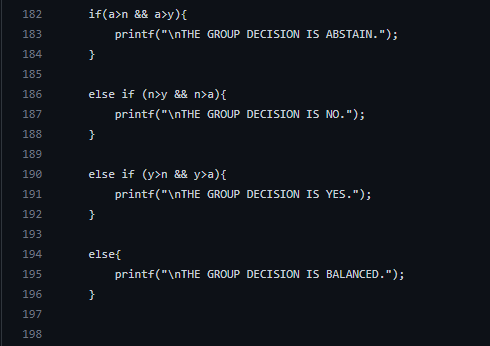


Figure 2.8: The group decisions expression to the user

**3 SIMULATION RESULTS**

In this part of the report, we will examine the simulation reports of the project code we have written. To do this, we will run our code in the CodeBlocks compiler and provide various inputs. Based on these inputs, we will obtain some outputs and examine how much of the requirements given in the project introduction are met in the CodeBlocks’ panel.

**3.1 Login Process**

During the login process, the code we have written asks the user for an integer ID number and a string password, respectively. The IDs we use in the voting system consist of the student numbers of the group members of the group members. In addition, the default passwords used are set to be all of our first name and the first two letters of our surnames.

If the input ID and passwords match in the system, the system will present an interface to the user asking what he wants to do. If there is an ID, password that does not exist in the system, or any entries that do not match, the system will warn and request ID and Password from the user again. An example of the login process is the following one.

Let us the group member Ahmet Berke Karaca wants to login the system for this example. So in the following figure, Ahmet input his corresponding ID and password data in the system. That means the code must say welcome to Ahmet and come up with an interface that Ahmet can decide the available options that he can do in the system.

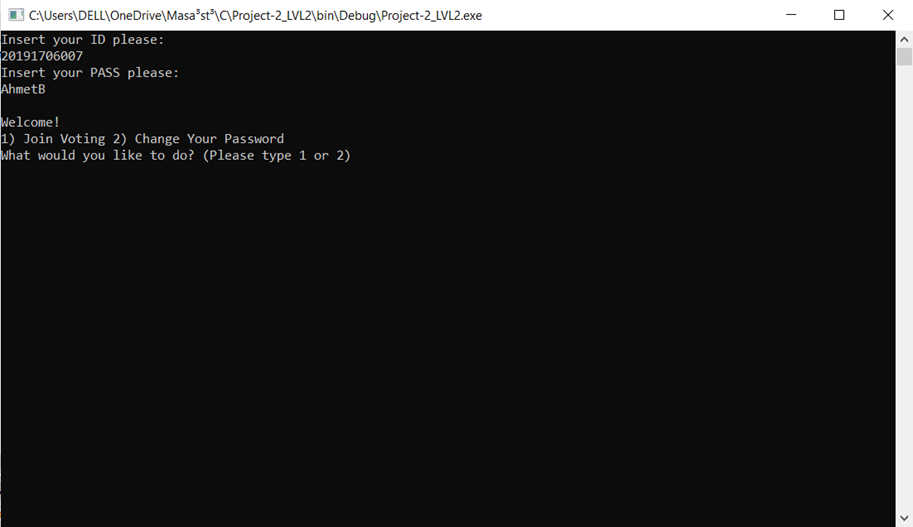


Figure 3.1: An example of a successful login

As we can see in figure 3.1, the system asks Ahmet what he wants to do (either join the voting or Change his personal password). That means the ID and password are matched successfully.

When it comes to the failure in the login process, the system shall warn Ahmet that the ID or password he input into the system are wrong. After this warn the system will ask him to input again. In figure 3.2, the example of a failure in login is shown.

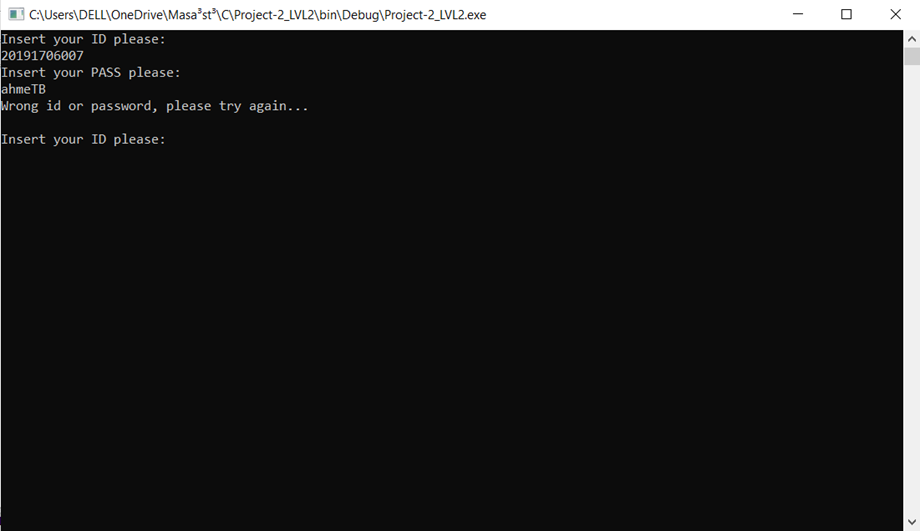


Figure 3.2: An example of a failed login

As we can see in the figure, although Ahmet used his ID properly, the password he used is not matched. That’s why the system indicated that he used the wrong ID or password and made him rn back. There are a lot of possible outputs to specify the problem exactly to the user in the programs we see in daily life, however, we decided this warning is enough for this project.

**3.2 Joining the Voting**

In the user interface, the program asks if the user wants to join the voting or change his/her password. To do that we represented these options as 1 and 2 as it was seen in the previous figures. So, if the user selects 1, that means he wants to join the voting. If the user selects 2, that means the program will lead the user to change his or her password. In figure 3.3, an example of joining the voting is shown:

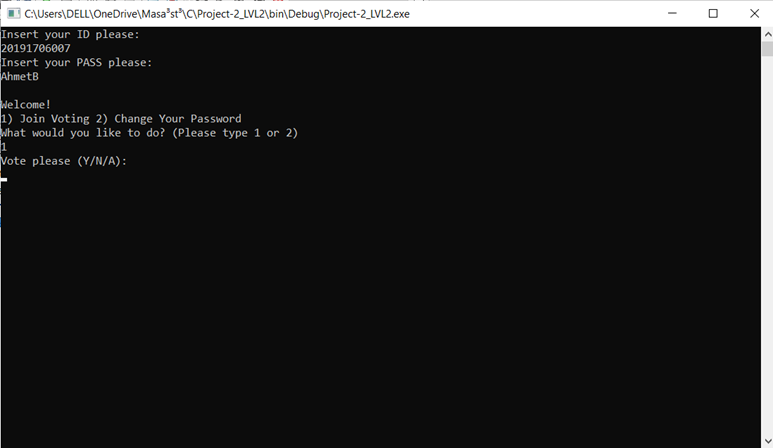


Figure 3.3: An example of a failed login

As we can see in figure 3.3, the user selected 1 and got into the voting interface. That means our group member Ahmet can vote now for his decision. The program will take the input from the user and ask if it is his or her decision is the last decision and give a second chance to change it.

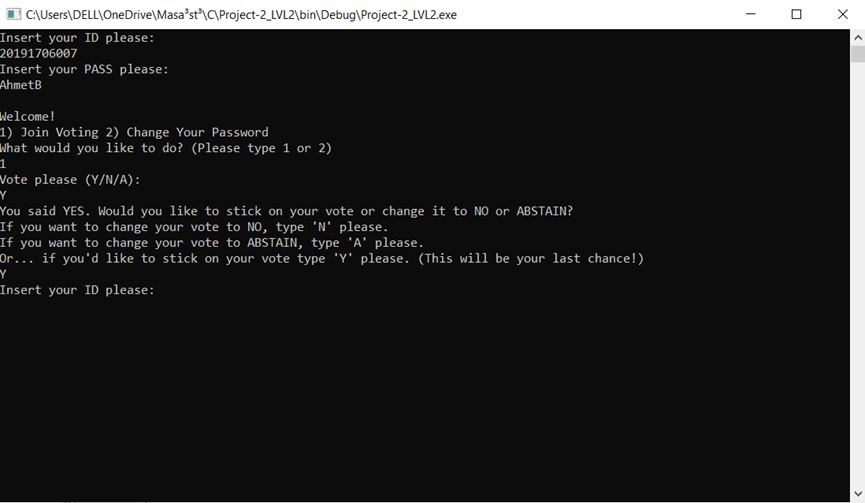
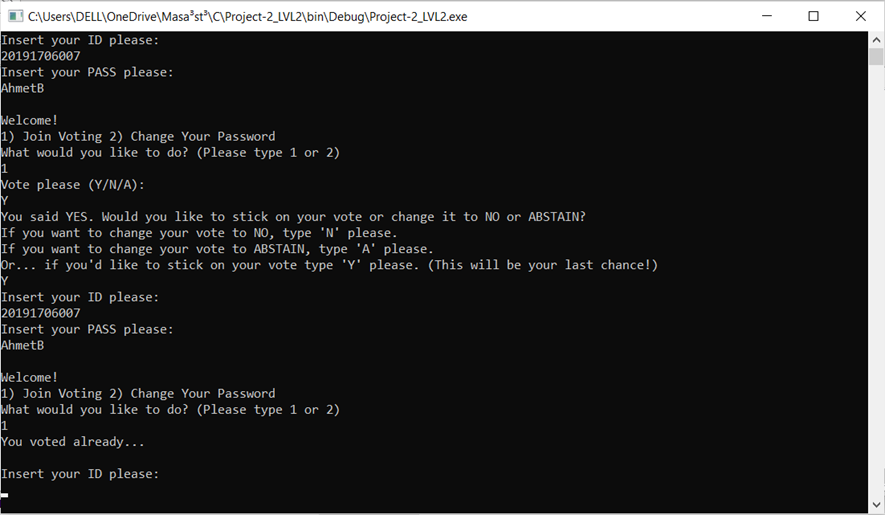


Figure 3.4: Changing the user’s vote option

In figure 3.4, the user voted first and got a warning from the program that he can change his decision to other options, and it will be his last chance. And finally, as soon as the decision is made by the user, the system asks another ID for the other users.

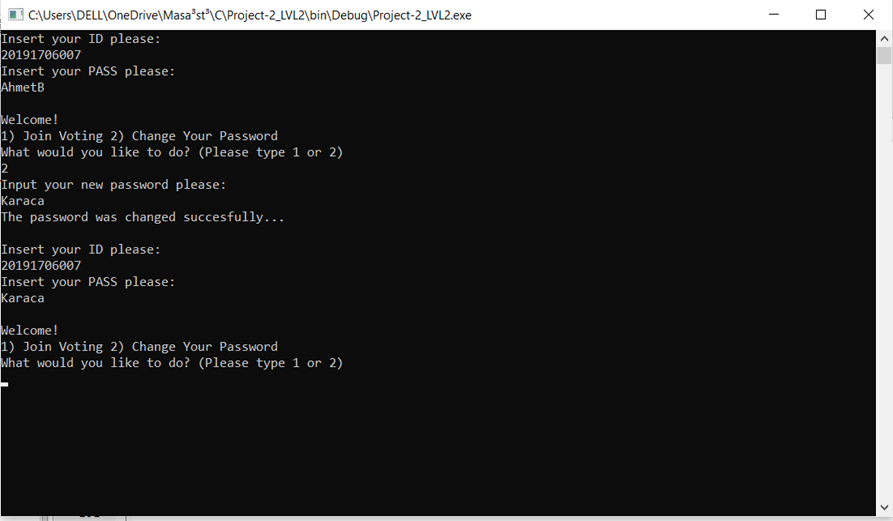
In addition, there should be a precaution for multiple voting for one user. That means if a user tries to vote for a second time the system will not allow it to happen. In figure 3.5, an example of this precaution is shown.

Figure 3.5: The precaution for multiple voting

As it can be seen from the figure 3.5, the user is not allowed to vote again since he voted already.

**3.2 Changing the User Passwords**

As it was discussed earlier, if the user selects 2 after his or her login successfully, he or she can change the user password. However, we should note that the user can use the new password if the program is run. If the program is stop, that means the user’s password will be the default one again. So, the implementation of it can be seen in figure 3.6.

Figure 3.6: Implementation of password change

As we can see from figure 3.6, our friend Ahmet changed his password as Karaca. And after the change, he successfully logs in to the system for the second time by using the new password.

**3.4 Obtaining the Results**

As was covered in previous parts of the report, the system will end the program when the result is certain. That means when a vote input (NO, YES, and ABSTAIN) take most of the votes, the program will give the results. In other words, if the number of any input votes reaches 2, the voting process will end, and the group will obtain the result. An example of a voting process is given in figure 3.7.

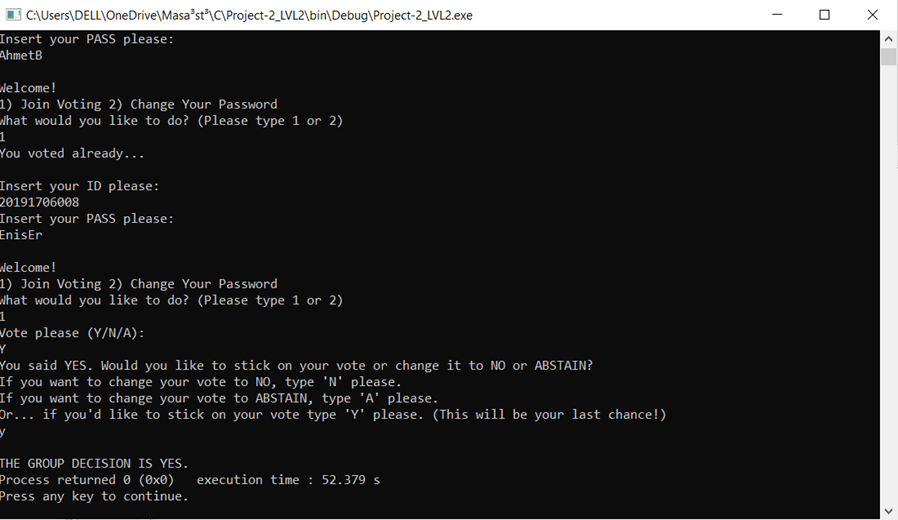


Figure 3.7: The voting process

As it can be seen in figure 3.7, the system ended gave the result as soon as the number of yes reached 2. Of course, the system has other possible results too. And, as we covered all these possible results during the code’s explanation, we did not consider it necessary to mention it again.

**4 CONCLUSION**

The code written fulfills and works with any due error as proposed in the project description. Working principle is just as proposed in the project description, which we need to enter password and user id in order to gain access to mode selection phase. After that, we encounter two selections in mode selection part, which is designed to be applicable and parallel to the promised code, where we can choose to vote or to change password without any due errors. Lastly, we enter our vote selection, and the system gives the last vote value at the end of the termination, again as proposed.