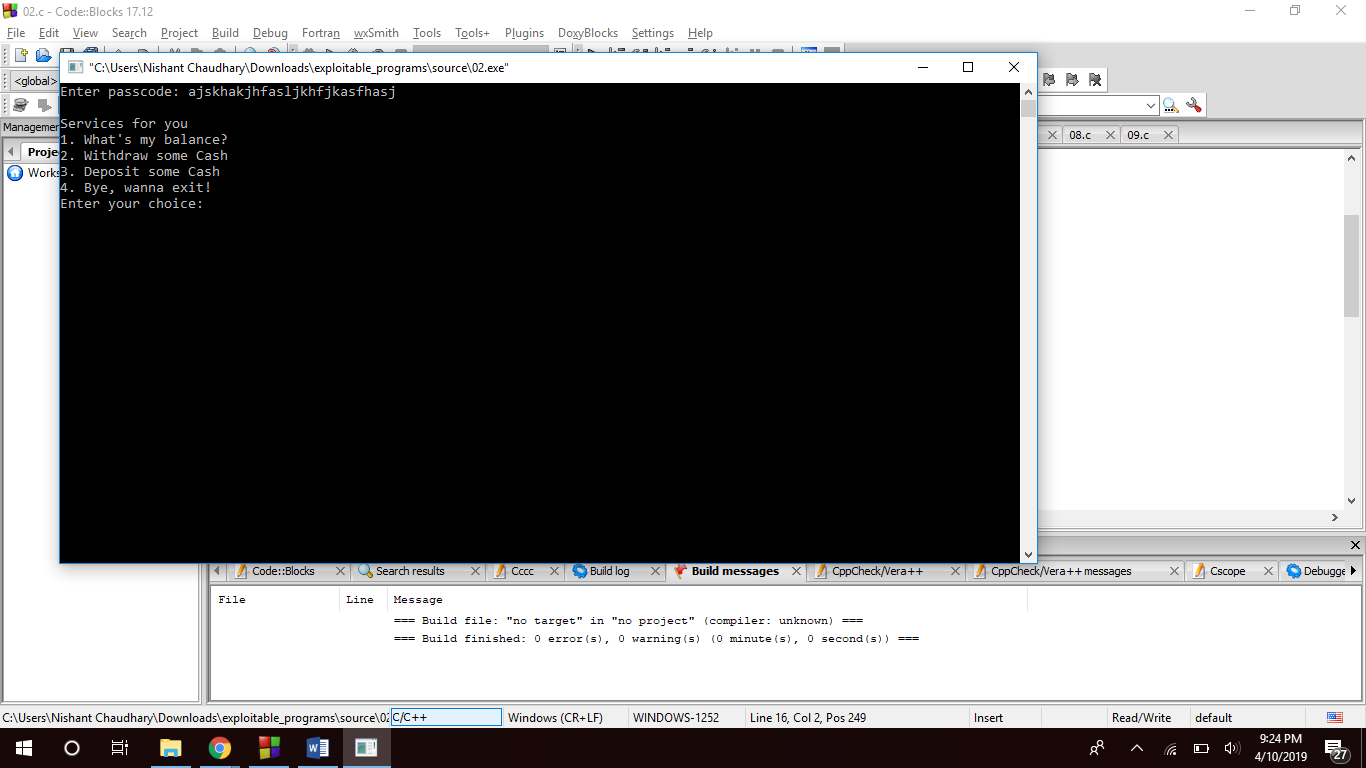
**Nishant Chaudhary**

**(Assignment 3 – Part 2)**

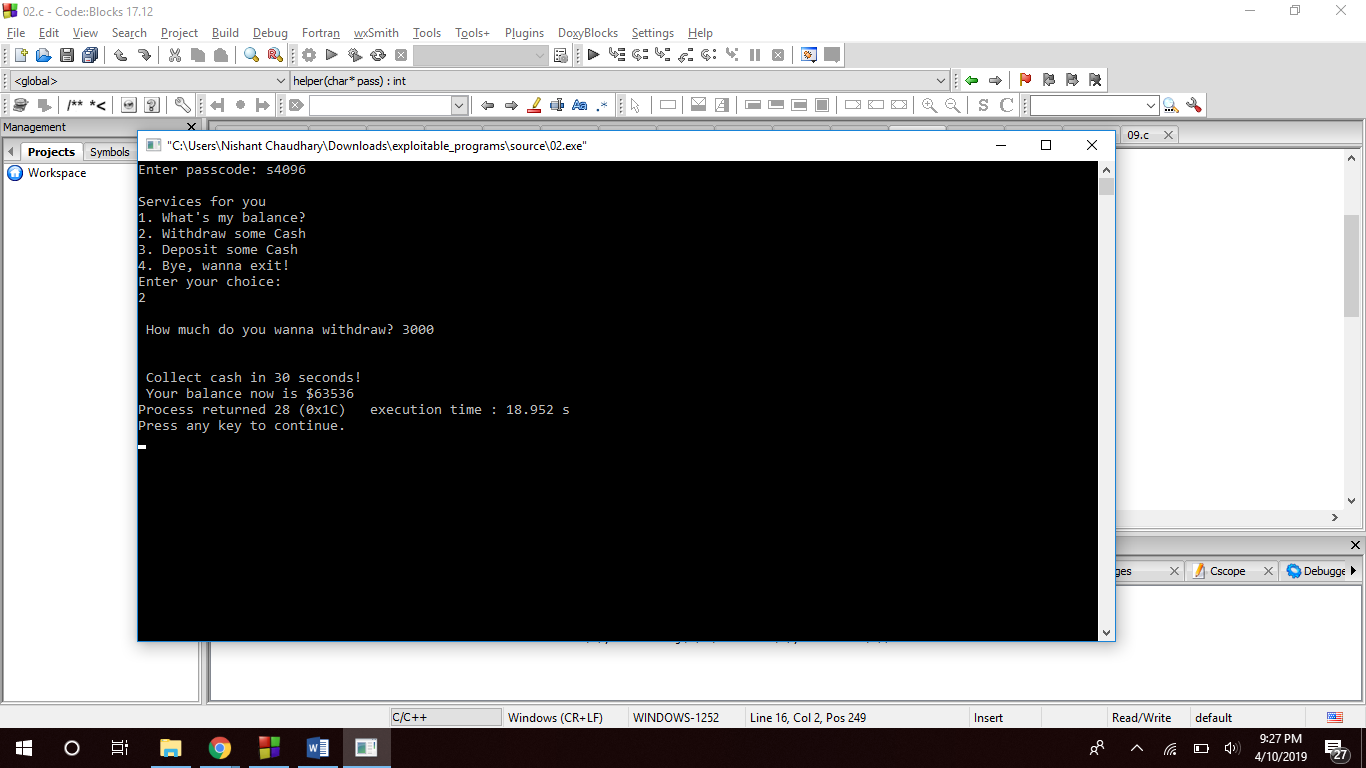
**CSE 5382**

**Question 2:**

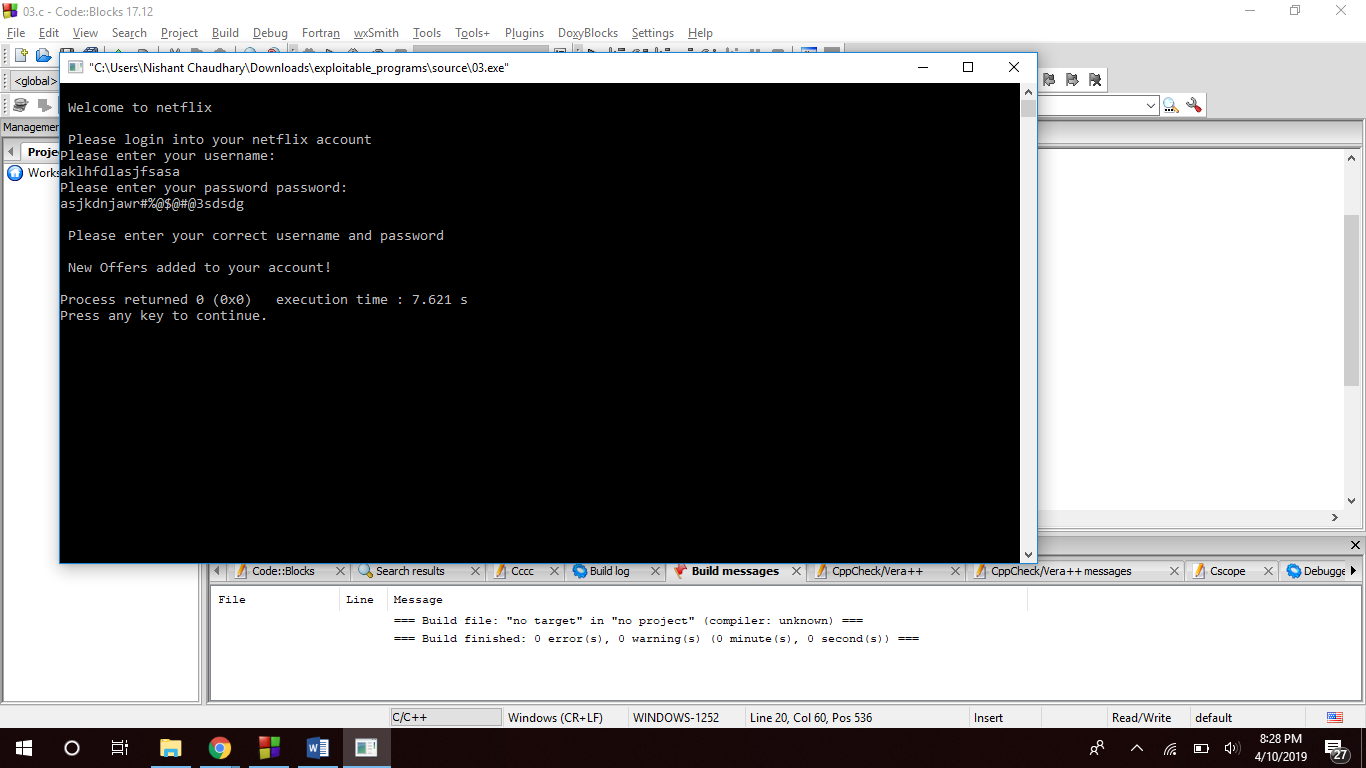


The character array passcode is of size 16, when we exploit it by overflowing the buffer the helper function returns the value of variable flag as 1 even though we didn’t entered the right password.

Even though if we enter the right password, there is integer overflow in this program. The available balance is $1000 but if we withdraw an amount greater than the available amount, it approves the transaction and updates the value of the current account amount to some random which is much greater than the expected value.

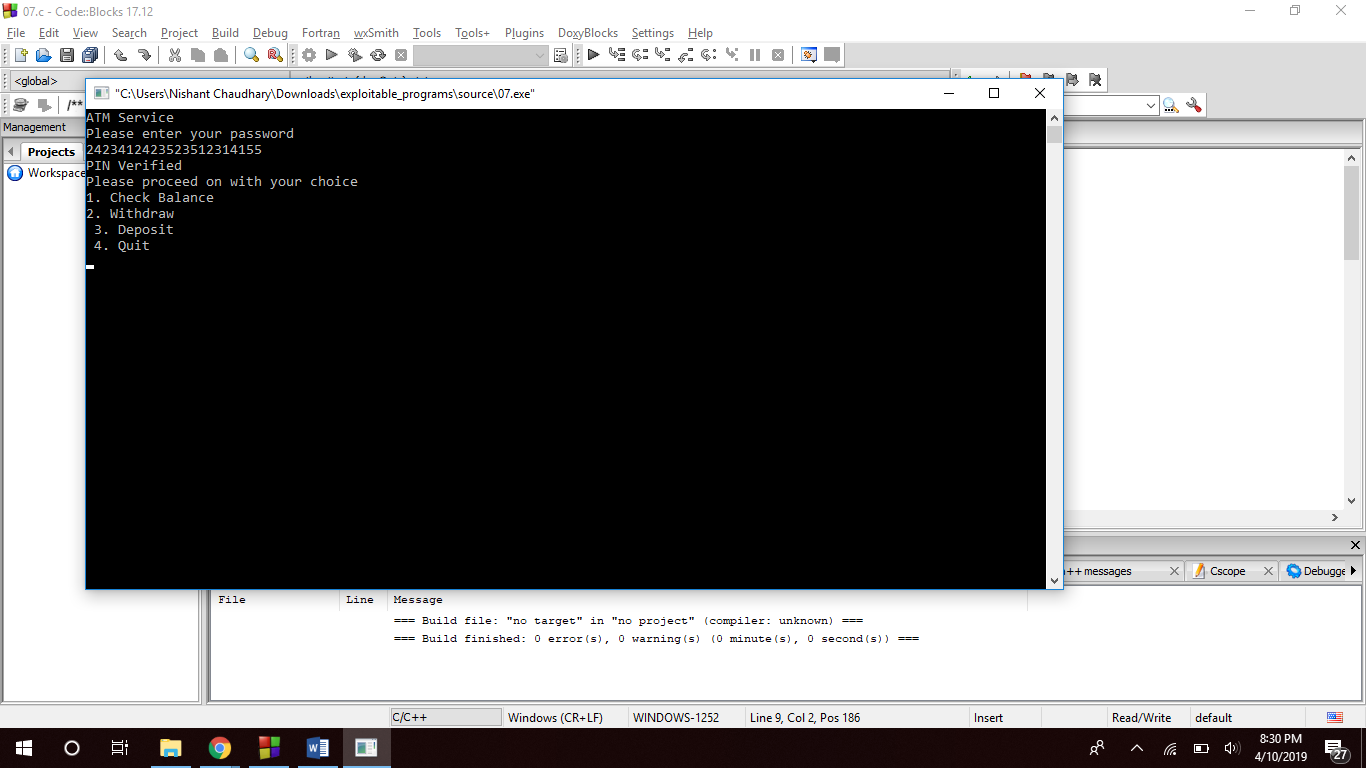


**Question 3:**



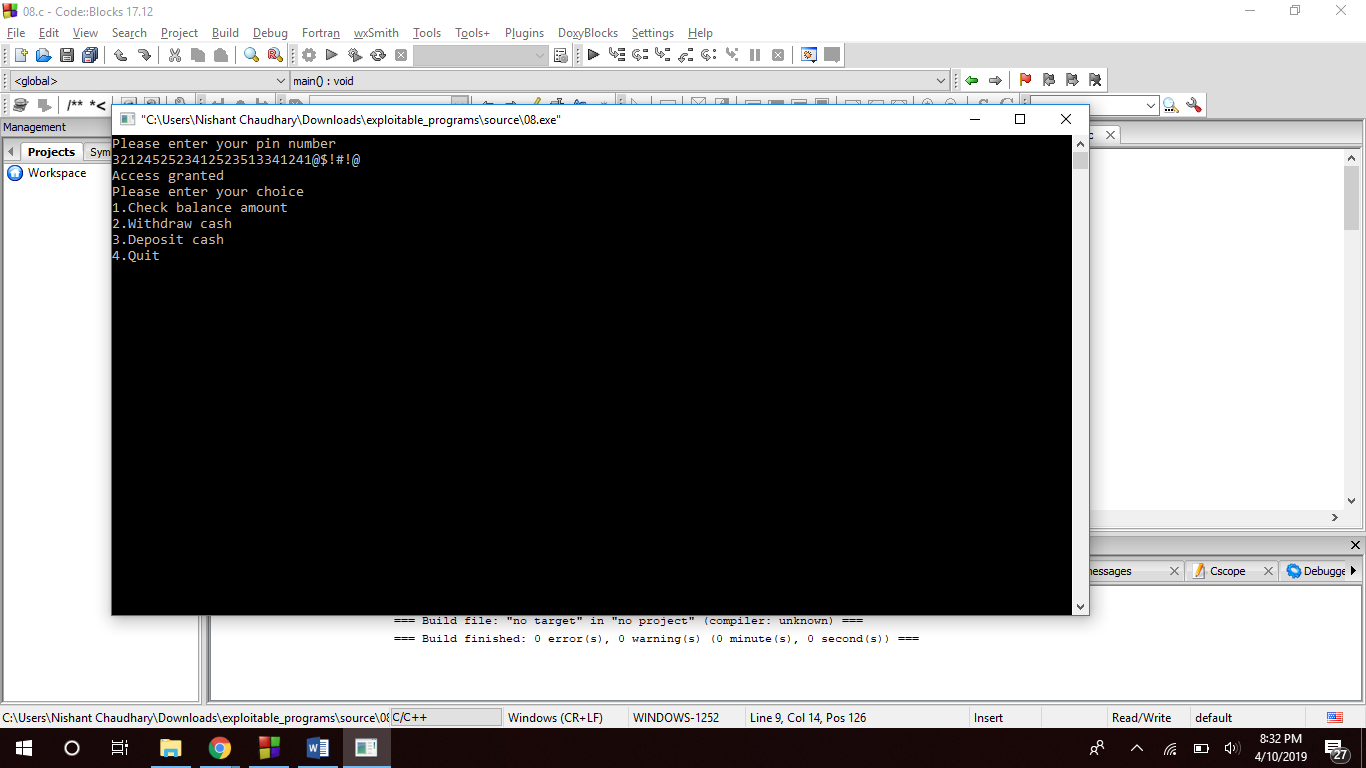
The size of character array username and password is 5 but if we overflow the array, it asks for correct username and password but updates the netflix.pass to 1 at the same time which gives us access to the account even though we entered a wrong username and password.

**Question 7:**



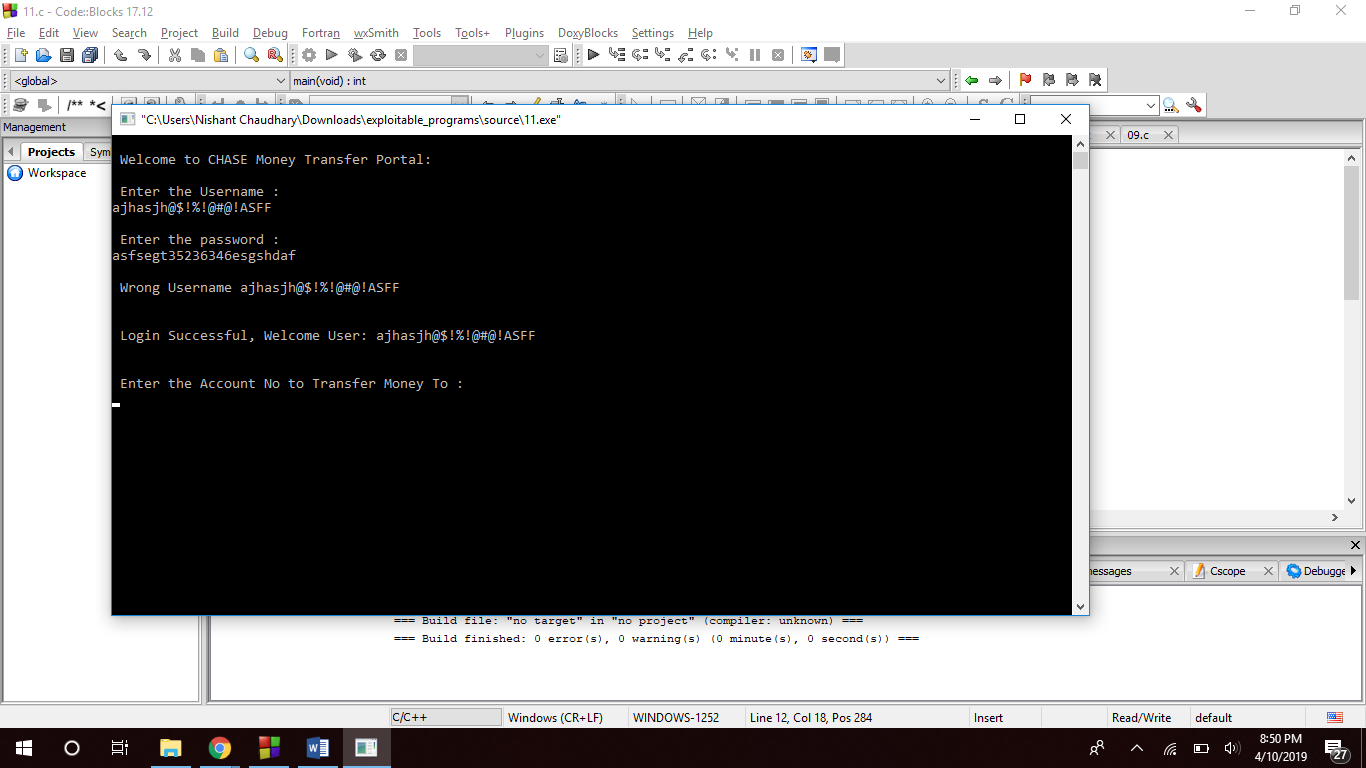
In this program the buffer we are using for storing the pin is 10 bytes in size and when we overflow it with a bigger value it gives us access to the ATM service even if it doesn’t match with the correct pin ‘1633’ but still we can access all the options via overflowing the buffer.

**Question 8:**



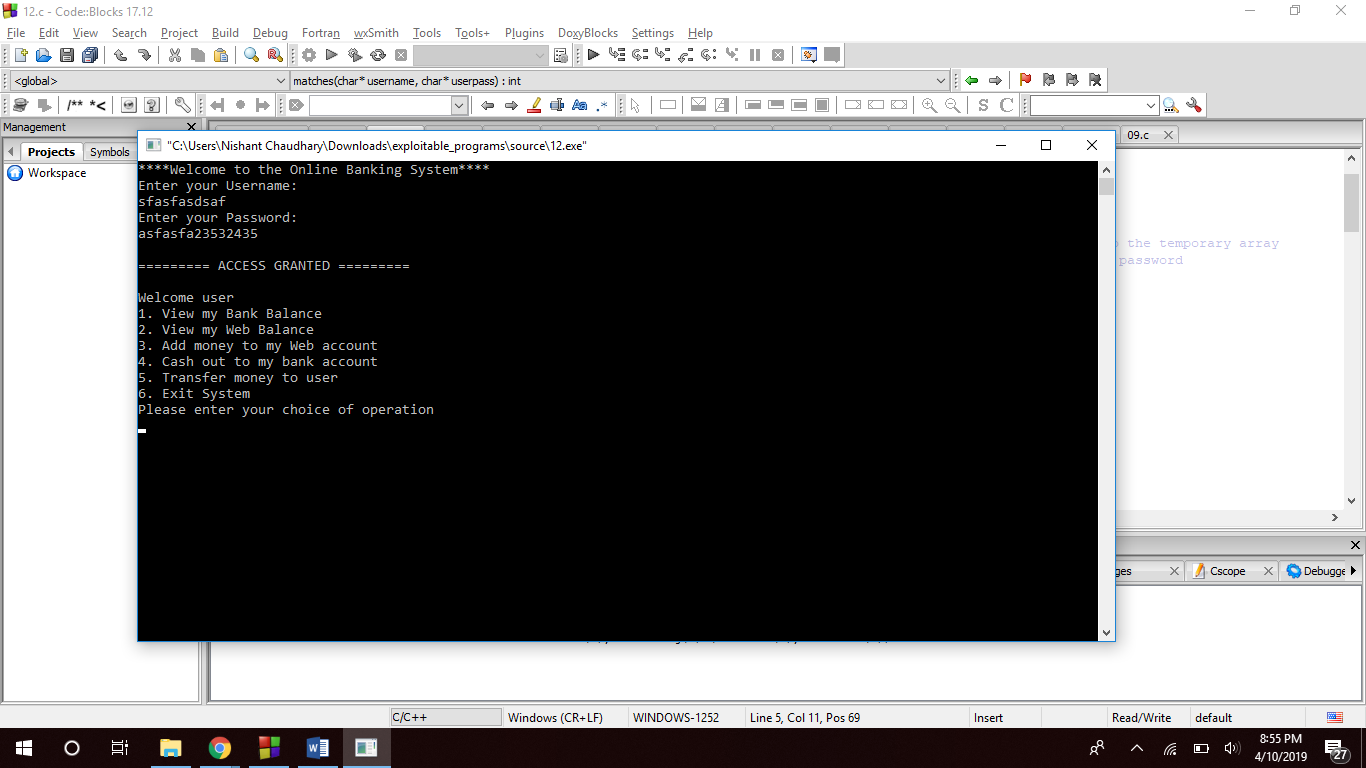
The noentered character is 6 bytes in size but if we overflow it with a bigger value, it gives us access to the account even though we didn’t enter the correct pin. Overflowing the buffer sets auth to 1, hence access to the account.

**Question 11:**



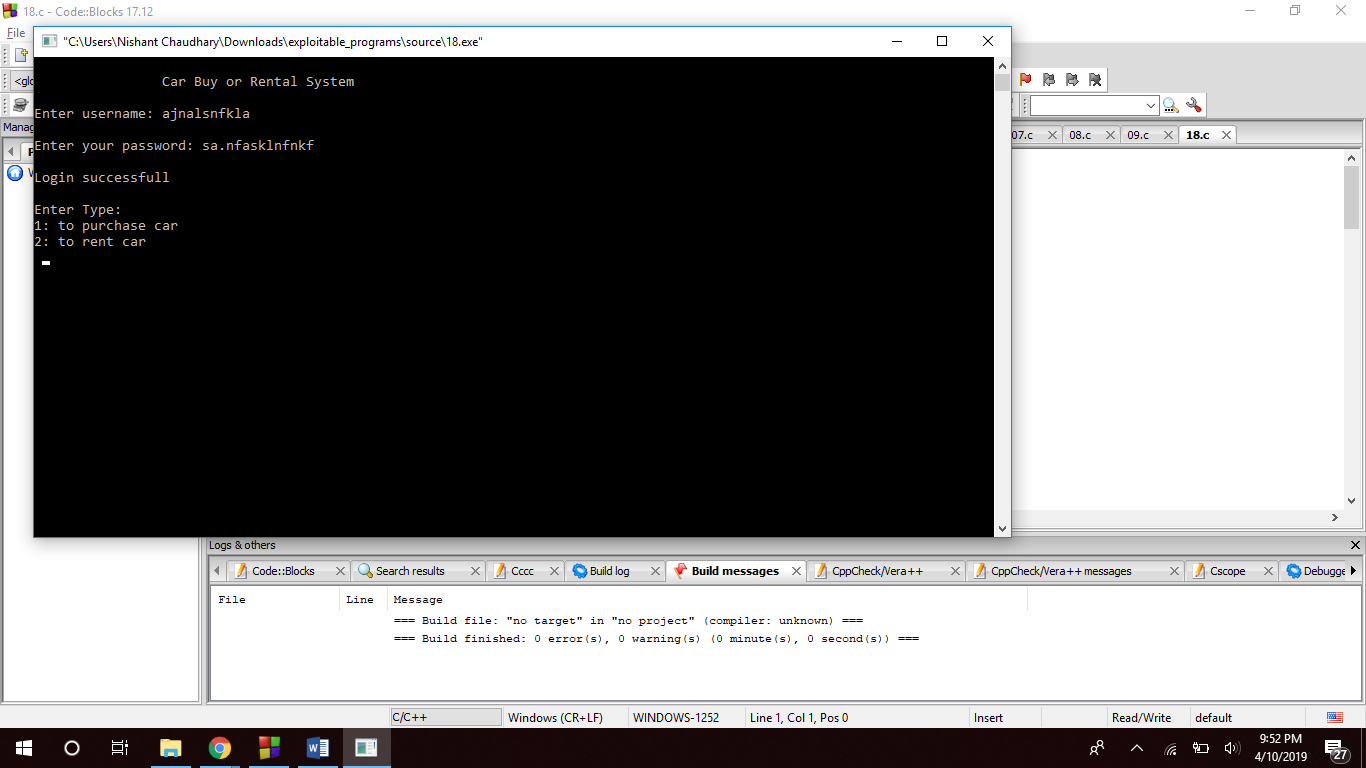
The size of username and pass character array is 15, and the flag is set to 0 but if you overflow the buffers. The program shows that the username is not correct but still it updates the flag to 1 and gives us access to the account.

**Question 12:**



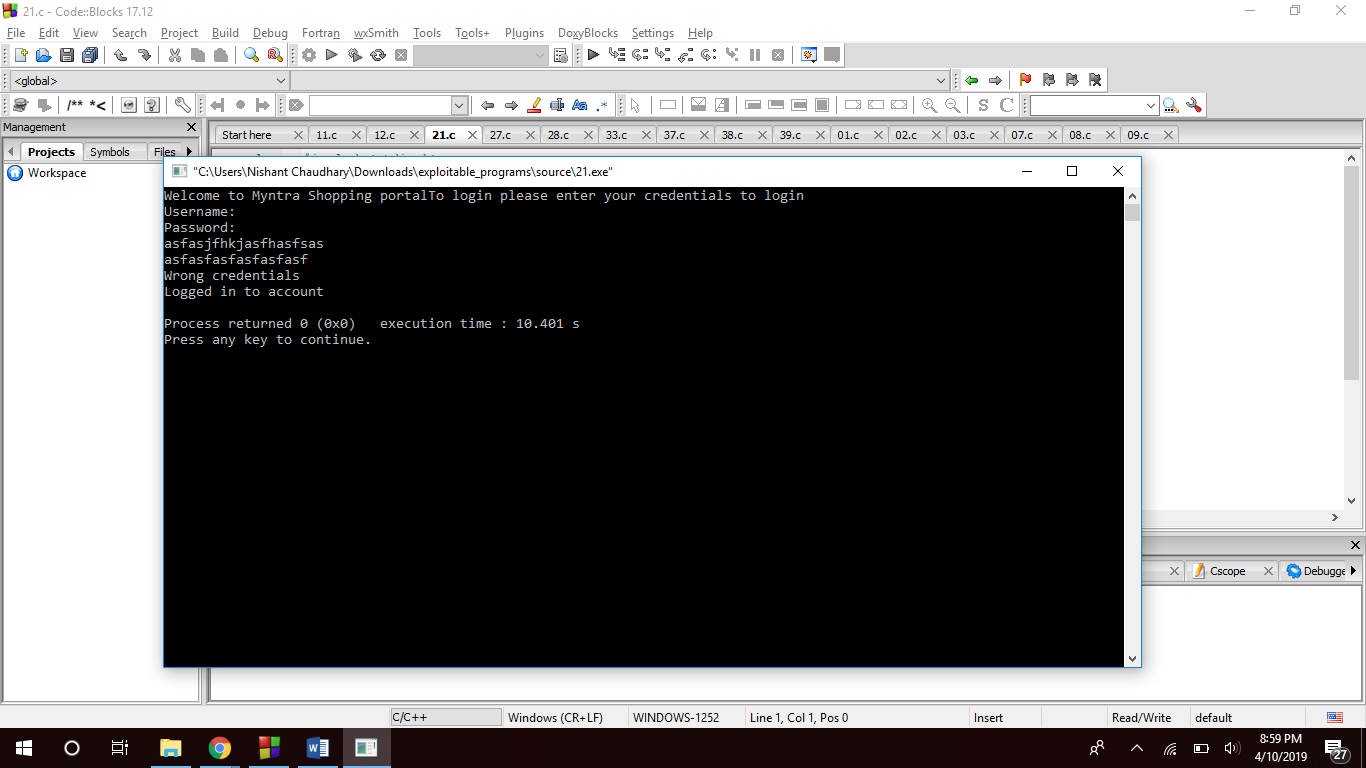
The userpass array is 8 bytes in size but if you overflow it with a bigger value it grants us the access to the account. Because of the overflow the value of the variable pig will be updated to 1 and hence we will get the permission to access the account even though the password was wrong.

**Question 18:**



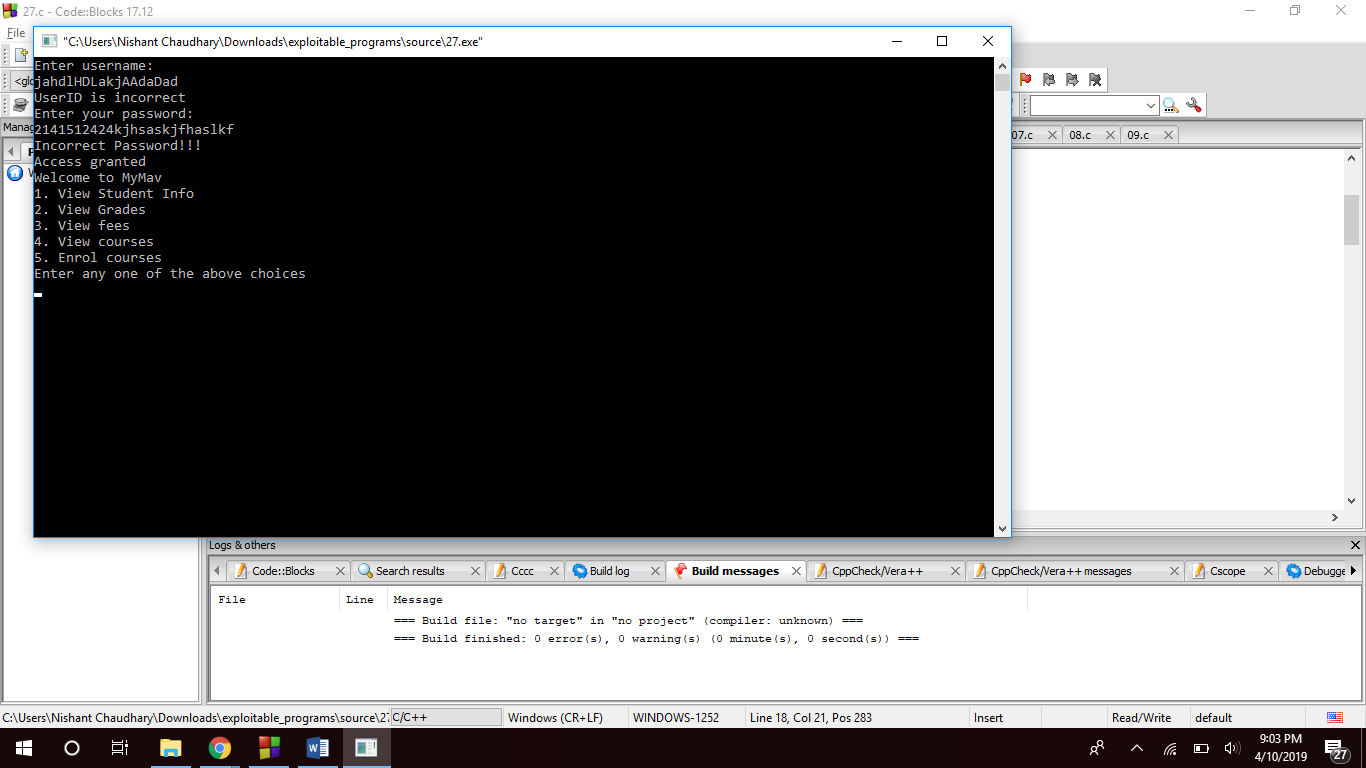
We have two character arrays username of size 6 and password of size 16 but when these arrays are passed to the function authenticate, password is moved to an array of 6 as well. When we overflowed these arrays, it changed the value of variable flag to 1 and ends up on giving access to the account.

**Question 21:**



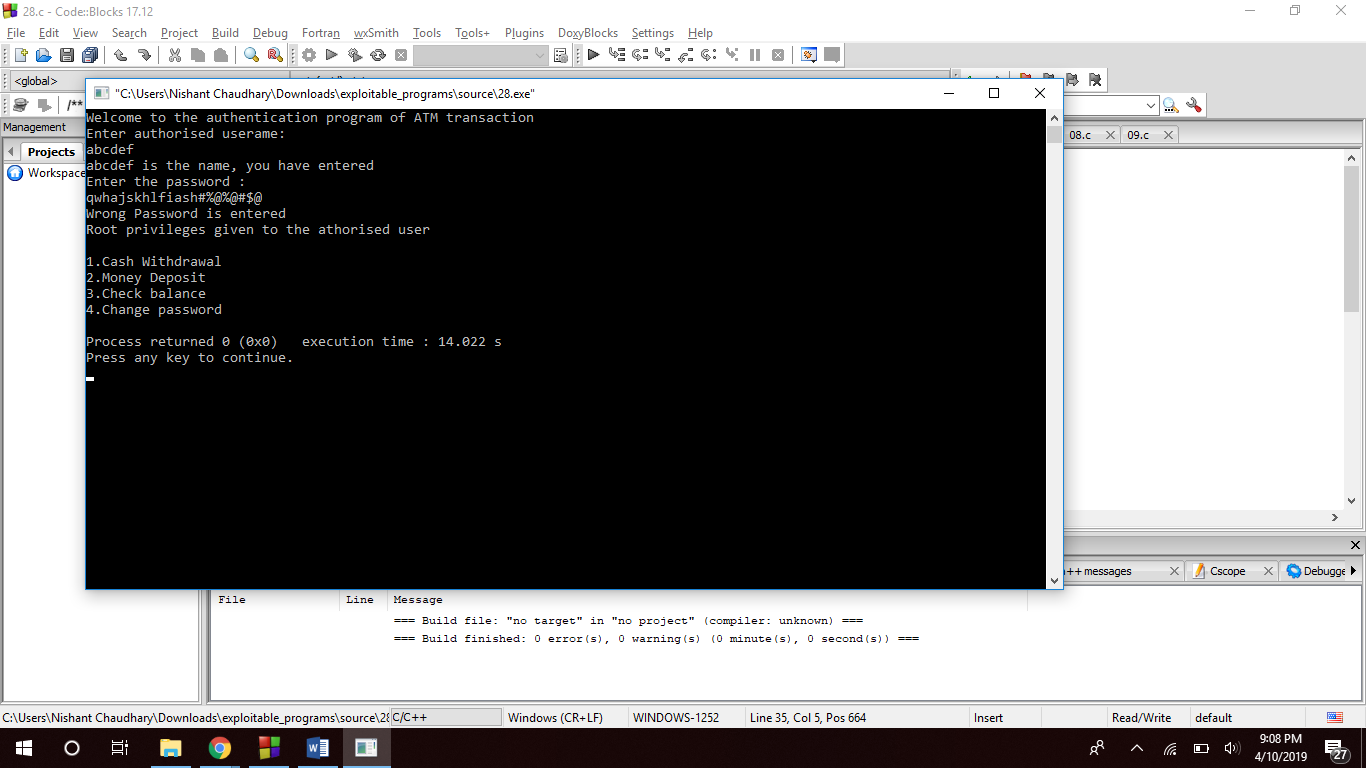
The username character array is of size 10 while password is of size 8, if we overflow these character arrays, even though we entered wrong credentials but still it updates the value of flag to 1 hence it ends up in logging us into to the account.

**Question 27:**



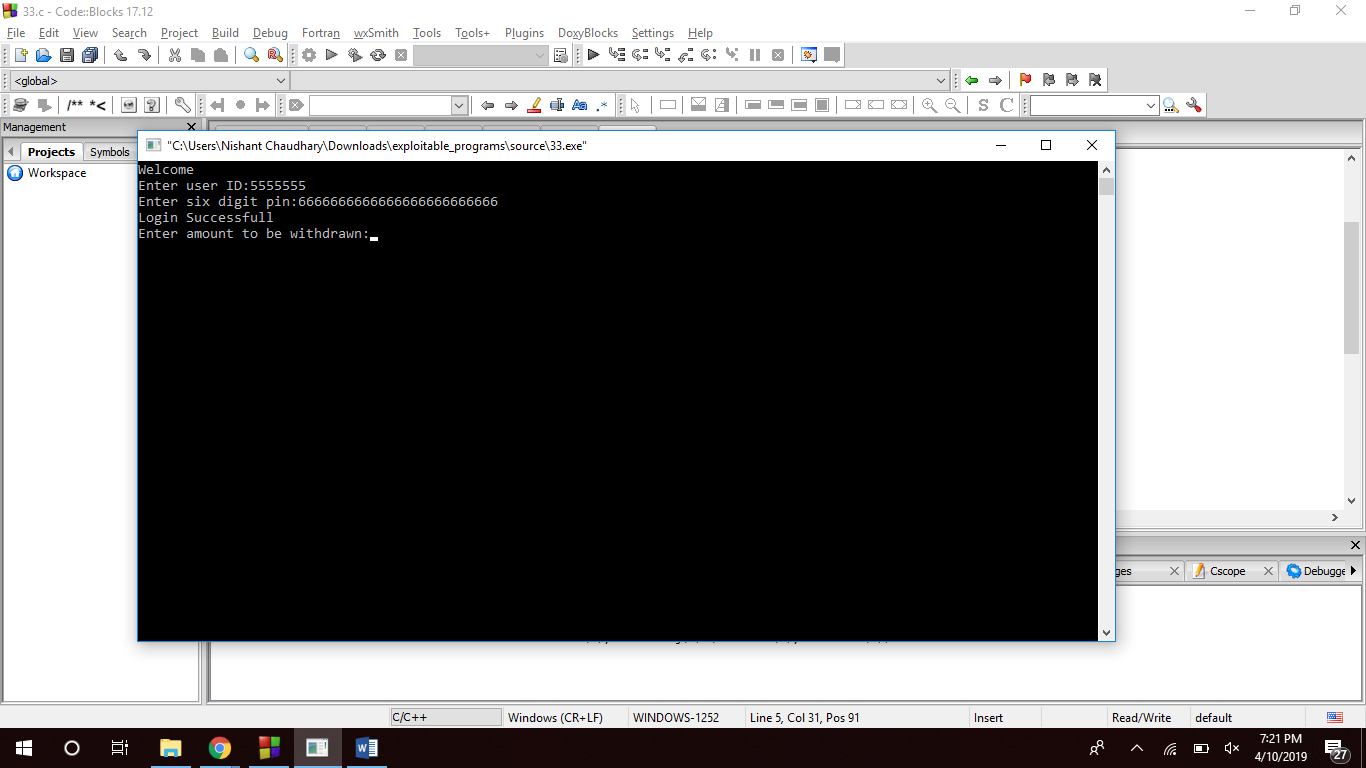
The id\_buff is a character array of size 7 while password\_buff is of size 16. To gain access to the account variable password should be updated to 1. In this caser we entered a wrong value for username and password, we overflowed these character arrays and even though the program knows that the username and password are wrong but still updates the variable to password to 1 because of buffer overflow and gives us the access to the account.

**Question 28:**

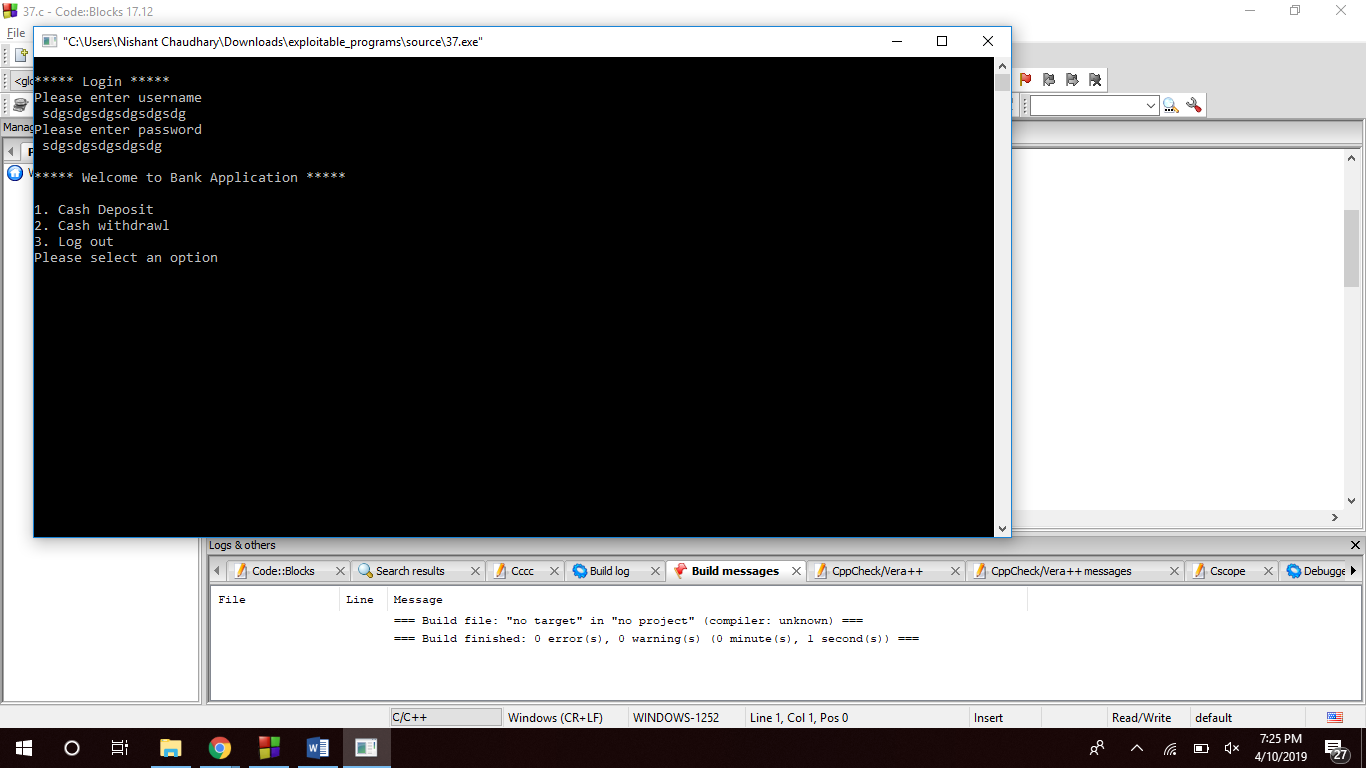


In this program we just entered a random username and then overflowed the buff character array(having size 12 bytes) which stores the password. Even though the password is incorrect but still it updates the value of variable passs to 1 which gives us the access to the account.

**Question 33:**

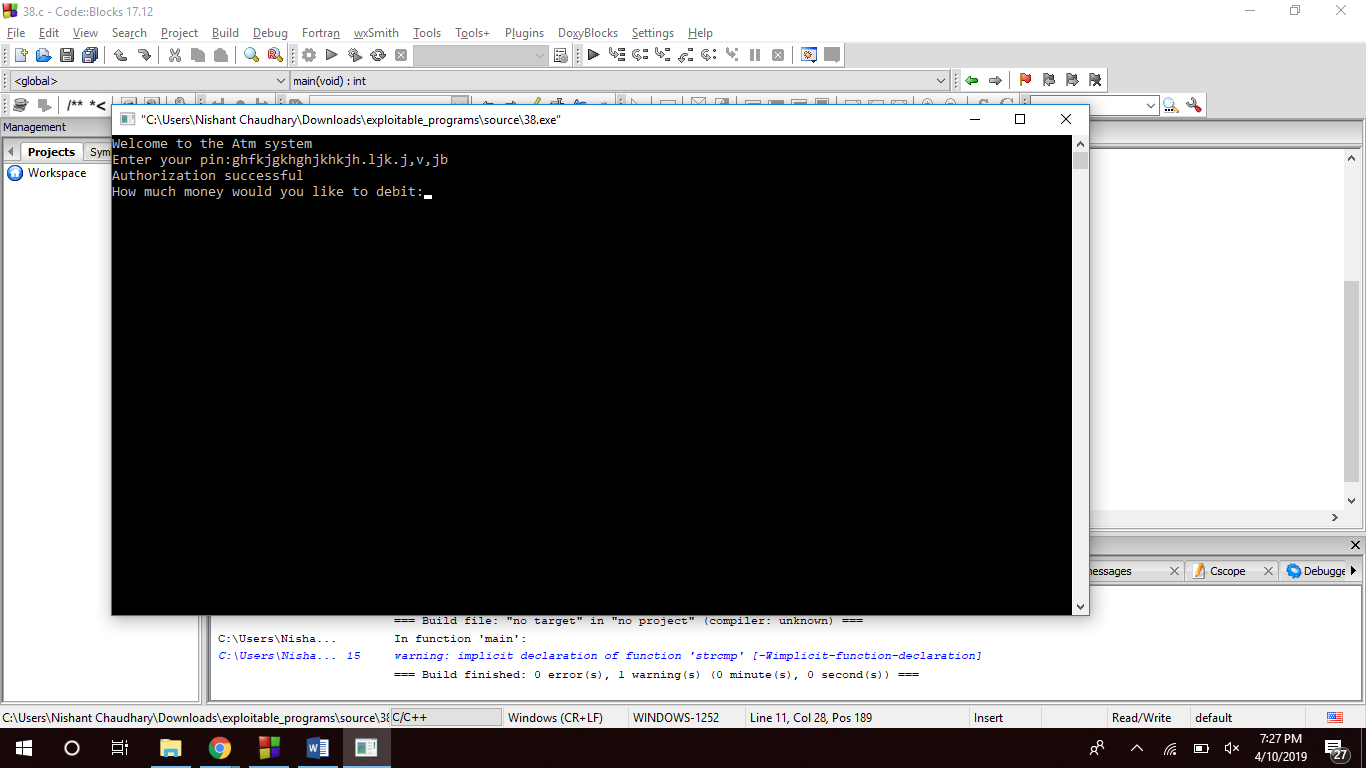


We just entered a random user id and then overflows password. The program was expecting a 6 digit pin but we overflows the buffer and even thought the password entered by us is incorrect but because of the overflow it updates the value of variable result to 1 and ends us giving access to the account.

**Question 37:**

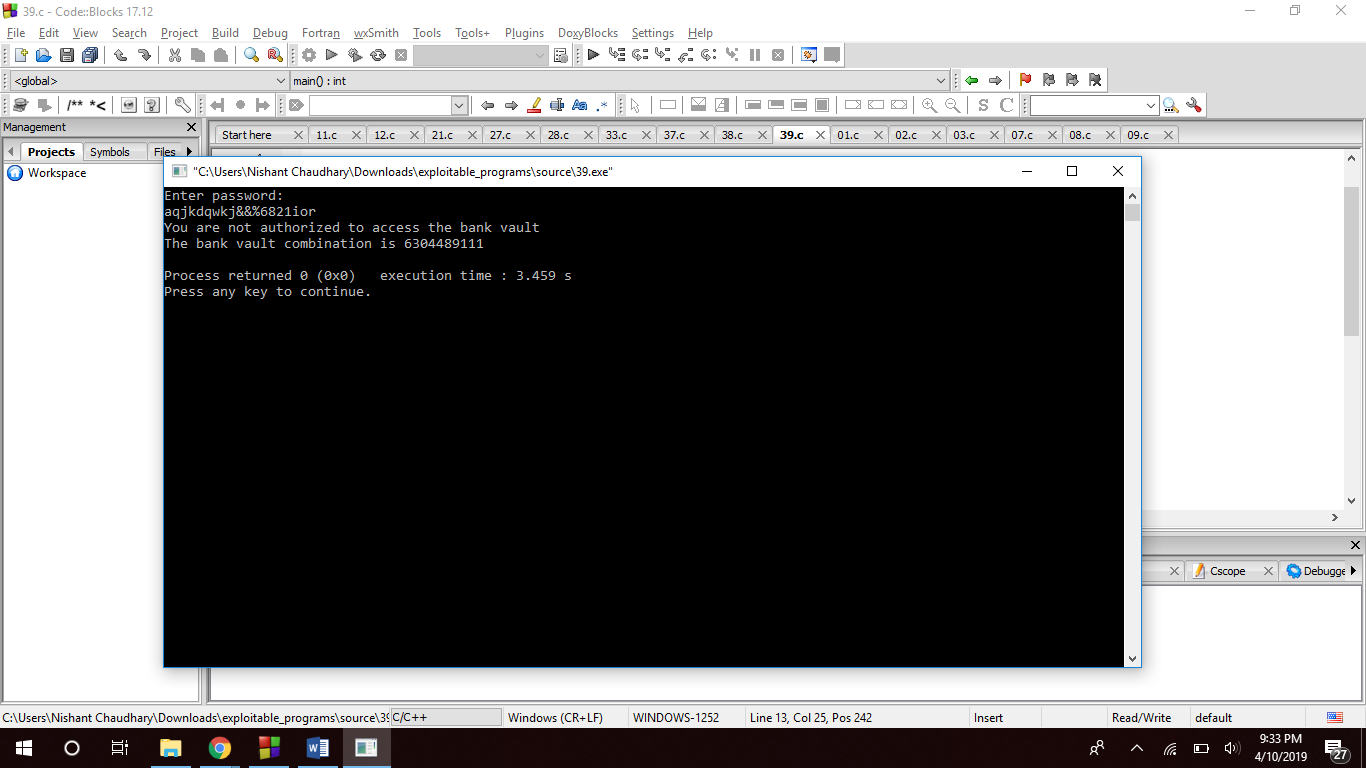
We had two character arrays of size 5(usr\_name and usr\_pass), when we overflowed these character arrays with a bigger value than expected, the value of variable login is updated to 1 even though the values we gave to both the character arrays are not correct.

**Question 38:**



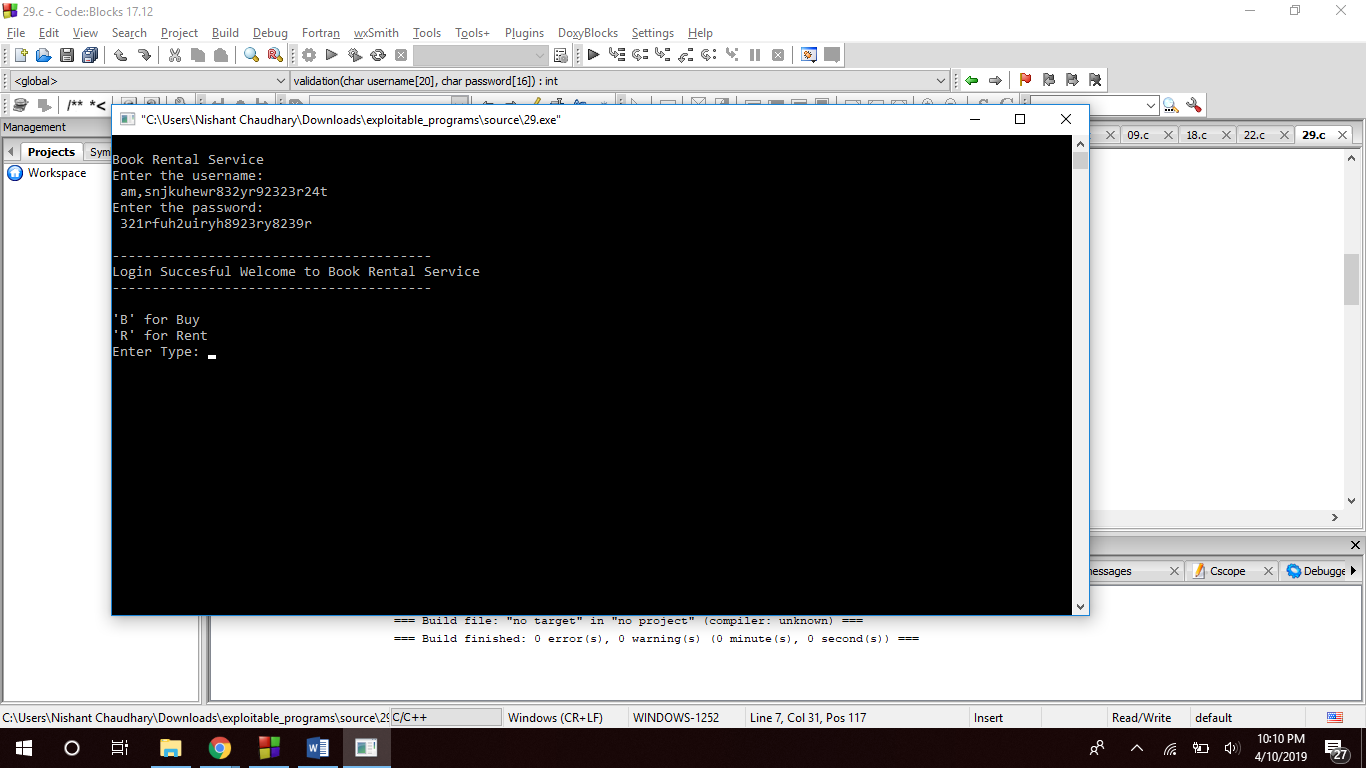
The character array enteredpass is of size 16, and when we overflowed this buffer with a wrong value, it authorized our login because the overflow caused the program to update the value of variable authorization to 1 giving us the access to the account.

**Question 39:**



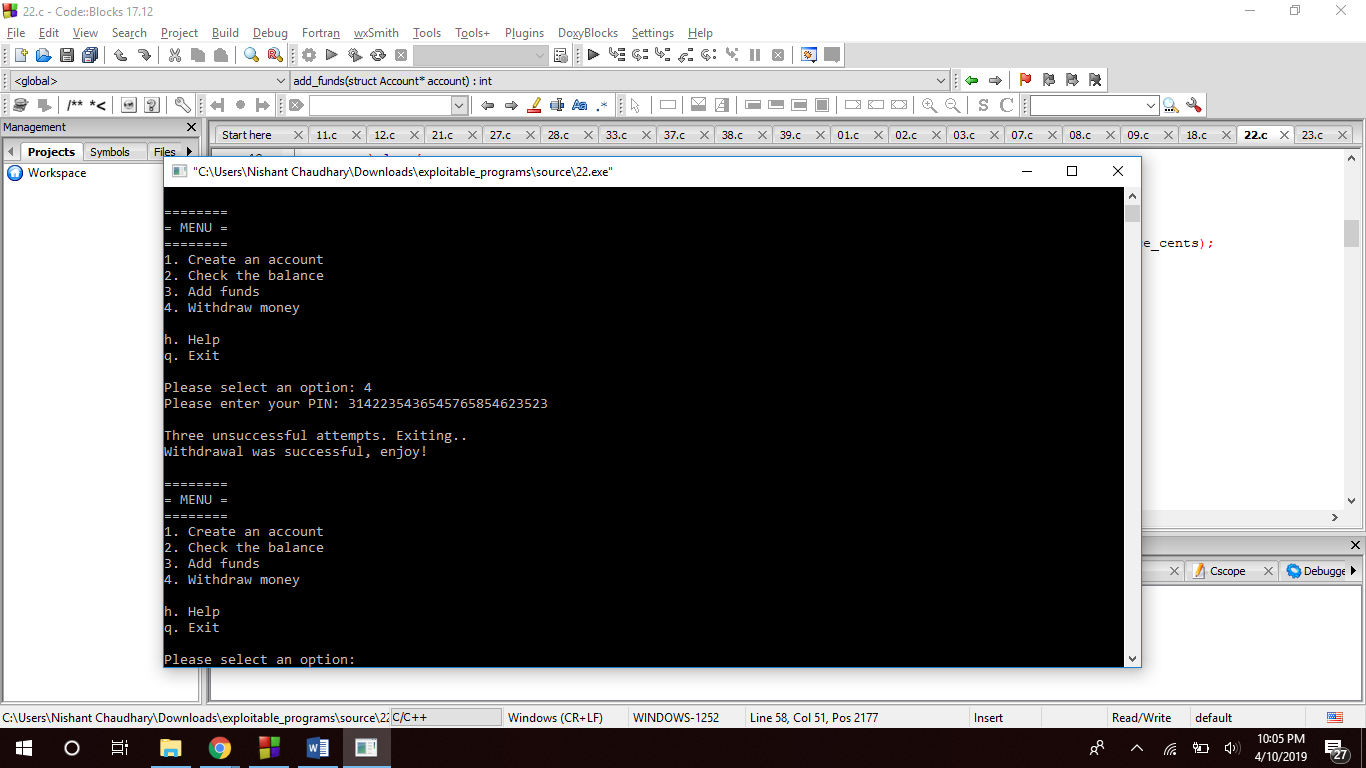
userPasswd is a character array of size 15, when we overflows this buffer it updates the value of variable u to 1 even though the password is not right. By doing so we can access the details of the vault combination even if you don’t know the right password.

**Question 29:**



The character arrays username and password are of size 20 and 16 respectively, when we overflowed the buffers the validation function changes the value of variable flag to 1 giving us access to the Book Rental System.

**Question 22:**



In this program even though we entered a wrong PIN, and we just did it once but the program says that there were three unsuccessful attempts (this happened because of the overflow). Even though the withdrawal was successful even if the password entered was wrong.