**21CY682**

**Secure Code Lab -1**

**Name: RAMYA AJAY**

**Roll Number: CB.EN.P2CYS22004**

**Date: 28-09-2022**

1.Write a C Program to authenticate a user using username and password. Have a list of 5 usernames and passwords in an array. If the entered username and password matches with the username / password combination in the array, then print as “Authentication Successful” else print “Authentication failed, try again”. The user is permitted to enter the wrong password only 3 times. If the user exceeds the limit, then print “Limit exceeded. Try later”.

#include<stdio.h>

#include<string.h>

int main()

{

char usernames[10][10] = {"user1", "user2", "user3", "user4", "user5"};

char passwords[10][10] = {"amrita1", "amrita2", "amrita3", "amrita4", "amrita5"};

char user[10];

char pwd[10];

int count=0,p=0;

while(count<3)

{

printf("Enter Username:");

scanf("%s", user);

printf("Enter Password:");

scanf("%s", pwd);

for(int i=0;i<5;i++)

{

if((strcmp(usernames[i],user)==0) && (strcmp(passwords[i],pwd)==0))

{

p = 1;

printf("Authentication is successful\n");

break;

}

}

if(!p)

{

printf("Authentication Failed, Try again\n");

count+=1;

}

else

{

break;

}

}

if(!p)

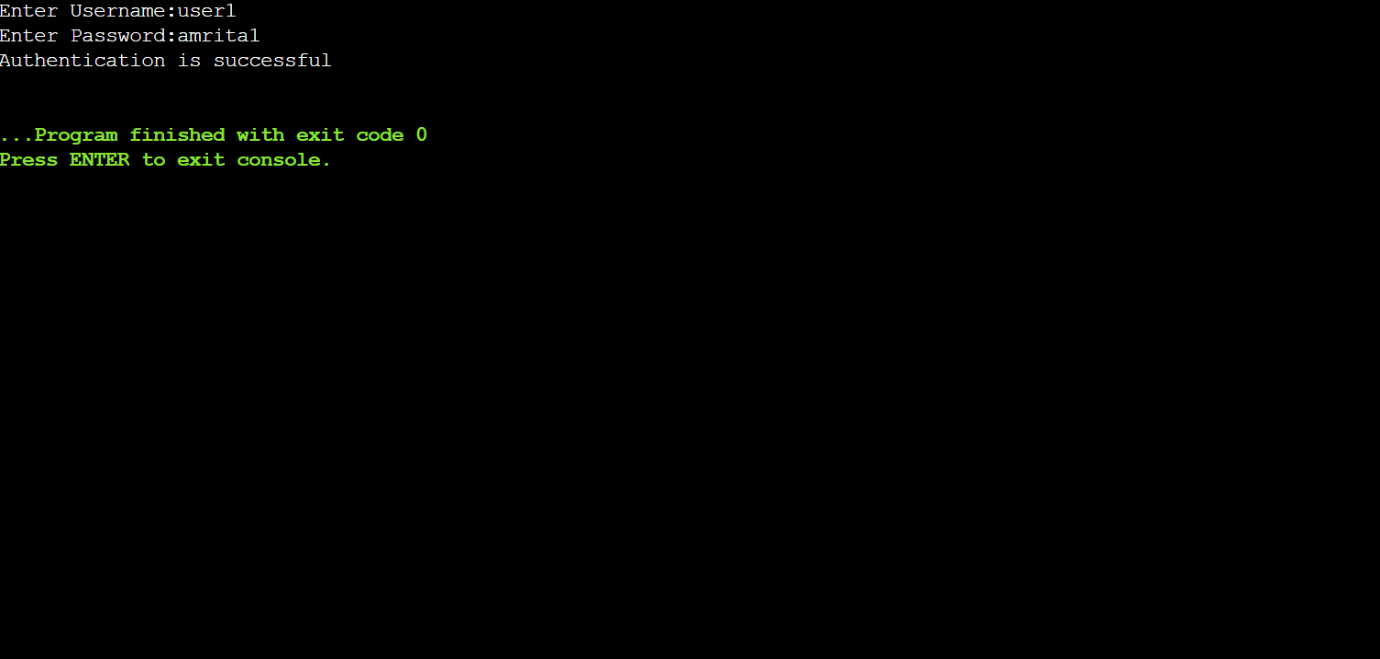
{

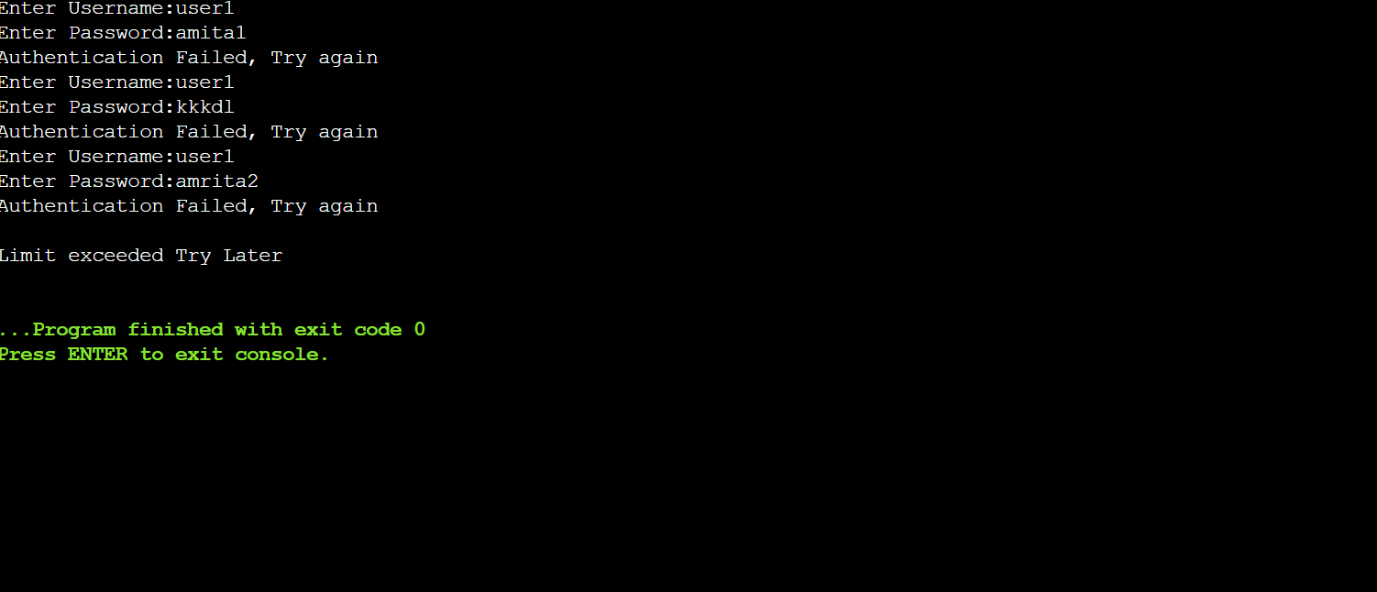
printf("\nLimit exceeded Try Later\n");

}

}

**Output:**





**Explanation:**

In this program, set of passwords corresponding to usernames are initialized. If the password is correct at the first try itself, then the authentication will be successful or else we will get 3 attempts to try again. If we try more than 3 times, then the limit exceeds .

2.Write a C program to create a password strength meter. A password is said to be strong if it is at least 8 characters long and contains at least one lowercase character, one uppercase character, one special character ( !@#$%^&\*()) and one digit. The program should obtain a password string from the user and compute the password strength (in percentage) based on the 5 criteria listed above for strong passwords

#include<stdio.h>

#include<string.h>

#include<ctype.h>

void main(){

char password[100];

int percentage=0, lower=0, upper=0, digit=0, spec=0, n=0;

printf("Enter the Password: ");

scanf("%s",password);

n = strlen(password);

for(int i=0; i < n; i++){

if(islower(password[i]))

lower++;

else if(isupper(password[i]))

upper++;

else if(isdigit(password[i]))

digit++;

else

spec++;

}

if (n < 6)

percentage = 0;

else if (n < 8)

percentage = 20;

else if (n < 10)

percentage = 40;

else

percentage = 60;

if(lower!=0)

percentage += 10;

if(upper!=0)

percentage += 10;

if(digit!=0)

percentage += 10;

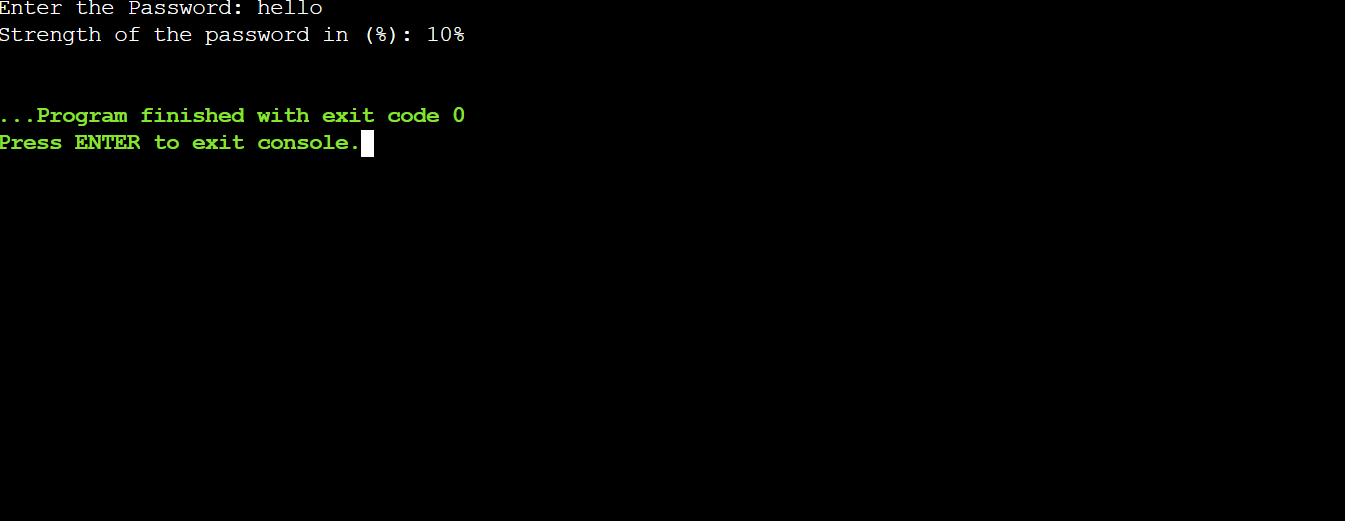
if(spec!=0)

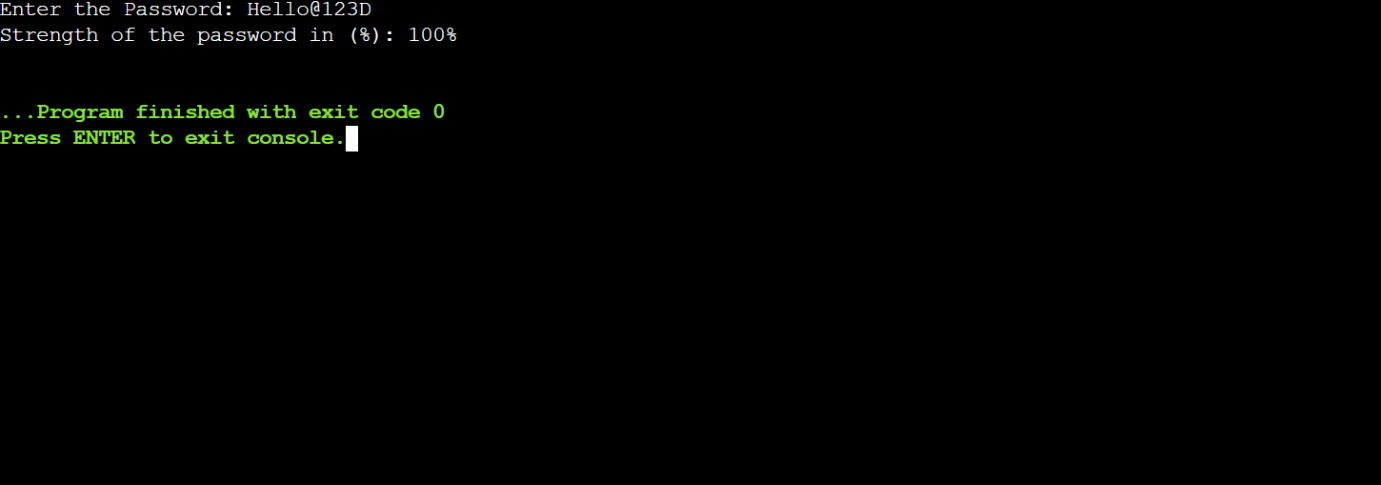
percentage +=10;

printf("Strength of the password in (%%): %d%%\n",percentage);

}

**Output**:





**Explanation**:

In this program, strength of the password is calculated in percentage.

The password will show 100% when it meets all the mentioned 5 criteria first,then length of the password is checked and allotted a percentage with the highest percentage being when the length is more than 10.The rest of 4 criteria are given 10% each, if it is included in the password.

3. Write a C program to generate strong passwords of a length specified by the user.

#include<stdio.h>

#include<time.h>

int main(){

srand(time(NULL));

char alphabet[100] =

"ABCDEFGHIJKLMNOPQRSTUVWXYZ";

char num[10] = "0123456789";

char spc[10] = "!@#$%^&\*()";

int i=0, x=0, y=0, z=0,w=0, n=0;

while(n < 8){

printf("Enter the number of characters: ");

scanf("%d", &n);

if(n<8)

printf("The password length should be atleast 8,Try Again\n");

}

printf("\nThe Password is: ");

for(int i=0; i<n/4; i++){

x = (rand()%26) + 1;

y = (rand()%9) + 1;

z = (rand()%9) + 1;

w = (rand()%26) + 1;

printf("%c%c%c%c", alphabet[x], num[y], spc[z], tolower(alphabet[w]));

}

if(n%4 == 3){

z = (rand()%9) + 1;

y = (rand()%26) + 1;

x = (rand()%26) + 1;

printf("%c%c%c", spc[z], tolower(alphabet[y]),alphabet[x]);

}

if(n%4 == 2){

x = (rand()%26) + 1;

y = (rand()%26) + 1;

printf("%c%c", tolower(alphabet[x]), alphabet[y]);

}

if(n%4 == 1){

z = (rand()%9) + 1;

printf("%c", spc[x]);

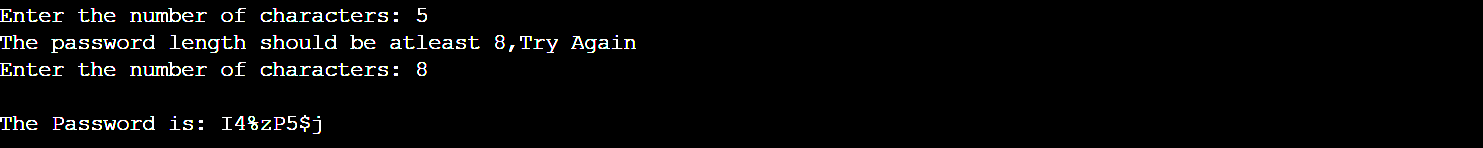
}

printf("\n");

return 0;

}

**Output:**



**Explanation:**

In this program, a password is randomly generated based on the password length input.

We make use of srand() because rand() produces the same output in each run. If the password length is not a multiple of 4, then based on remainder the rest of the characters are randomly added. If the password length is a multiple of 4, then an upper case, lower case, digit and a special character in sequence is added randomly till it reaches the password length.