**LOG OF MY PROJECT**

**PROJECT DESCRIPTION:**

This is a program in C for a login screen with basic authentication features like sign up, login, forgot password, change password, and delete account. It uses file handling to store and retrieve user account information. The program starts by including necessary header files like stdio.h, stdlib.h, conio.h, string.h and time.h. Next, the program defines a structure account to hold user information like first name, last name, username, phone number, gender, date of birth, and password. After that, the program defines several functions like menu(), signup(), login(), forgot\_pass(), change\_pass(), delete\_acc(), and brokenlines(). The menu() function displays a menu with options like sign up, login, forgot password, change password, and delete account. It takes user input and returns the selected option. The signup() function prompts the user to enter their information and password, validates it, and writes the data to a file. The login() function prompts the user to enter their username and password, validates it by reading the data from the file, and displays their account information if the login is successful. The forgot\_pass() function prompts the user to choose a recovery method (username or password), validates it by reading the data from the file, and displays the password or username. The change\_pass() function prompts the user to enter a code that is generated randomly, validates it, and allows them to change their password. The delete\_acc() function is commented out and not used in the program. The brokenlines() function prints a line of dashes to separate different sections of the program output. Finally, the main() function runs an infinite loop that displays the menu and calls the corresponding functions based on the user's selection.

**Error:** subscripted value is neither array nor pointer nor vector.

**Reason:** did not use “printf” instead used “print”.

**Note:** “printf” is for “stdio.h” library.

Encountered a problem; if conditional statement not working, had to change IDE

**Meaning of the takepassword function code:**

This function **takepassword(char pass1[20])** is designed to securely take a password input from the user in a console application. Here’s a breakdown of what it does:

1. **Variable Initialization:** It starts by declaring a char variable ch to hold the character input from the user, and an int variable i initialized to 0 to serve as the index for the password string.
2. **Infinite Loop:** The function then enters an infinite loop (while(1)) which will keep running until explicitly broken.
3. **Character Input:** Inside the loop, it uses getch() to get a character input from the user without echoing it to the console.
4. **Enter Key Check:** If the user presses the “Enter” key (ASCII value 13), it adds a null character ('\0') to the end of the password string and breaks out of the loop.
5. **Backspace Key Check:** If the user presses the “Backspace” key (ASCII value 8), it decreases the index i by 1 (if i > 0), effectively “deleting” the last character entered. It also prints "\b \b" to move the cursor back one space and overwrite the last asterisk with a space, visually erasing it from the console.
6. **Tab or Space Key Check:** If the user presses either the “Tab” key (ASCII value 9) or “Spacebar” key (ASCII value 32), it simply continues to the next iteration of the loop without adding anything to the password string.
7. **Other Characters:** For any other character input, it adds that character to the password string and increments i by 1. It also prints an asterisk ('\*') to the console to represent a password character.

Please note that there’s an issue with this code: inside this function, there’s a line int i=0; which redeclares i inside the loop. This line should be removed, as it resets i to 0 in every iteration of the loop, causing only the first character of pass1 to be set.

**Color Code for c**

COLOR [attr]

attr Specifies color attribute of console output

Color attributes are specified by TWO hex digits -- the first

corresponds to the background; the second the foreground. Each digit

can be any of the following values:

0 = Black 8 = Gray

1 = Blue 9 = Light Blue

2 = Green A = Light Green

3 = Aqua B = Light Aqua

4 = Red C = Light Red

5 = Purple D = Light Purple

6 = Yellow E = Light Yellow

7 = White F = Bright White

The first is the background while the second is the text

Example: “Color 70” produces a white background and black text.