Assignment: Playing with String

You are given a skeleton C file called **StringAssignment.c** for this assignment. Do necessary modifications on it.

Task 1: Signup System

In the **StringAssignment.c** file there are two global arrays of String namely *emailArr* and *passArr*. Implement the function *signupUser()* that takes two String as input, one for user email and another for password. You need to do the following checks:

1. Is the email in valid format?
2. Does the password contain at least 8 characters?
3. Does the password have both uppercase and lowercase letters?
4. Does the password contain digits?
5. Does the password contain any of the special characters ($ # * @)?

If any of the checks returns *false*, the signup will not be successful. In that case you need to show proper error message, and return FAIL_VAL. Otherwise, the signup will be successful if all of these checks return *true*. In that case show success message, return the array index, and store the user email and the password into *emailArr* and *passAarr* respectively.
N.B. You need to discard duplicate signup.

Task 2: Login System

Implement the function *loginUser()* that takes two String as input, one for user email and another for password. It will search for the given user email in *emailArr*. If the user email exists and the given password matches with the stored one against that email, then login will be successful and you should return the array index plus show a success message. Otherwise, login will be unsuccessful and you should return FAIL_VAL plus show a proper error message.

Task 3:Build Profile

In the **StringAssignment.c** file there are two more global arrays of String namely *nameArr* and *addressArr*. Implement the function *buildUserProfile()* that takes four String as input, one for user email, one for first name, one for last name, and another for the address. You need to do the following thing:

1. Concatenate first and last name to a single name field, and store it into *nameArr* at the proper index.
2. Store the address into *addressArr* at the proper index.
3. If no user is found against the given email, show error message.

Assignment designed by Md. Rafi-Ur-Rashid, Lecturer, CSE, UIU

Task 4: Search

1. Implement the function searchByName*()* that takes one String input for name. You need to show the email, name, and address of those users found against that name.
2. Implement the function searchByAddress*()* that takes one String input for address. You need to show the email, name, and address of those users found against that address.

For both of these searches: the search should be case insensitive and allow substring properties. If search fails to find the data, show error message.

Points:

Task	Points
1	15
2	10
3	10
4	15

Important Instructions:

1. You **must** perform all the necessary modifications in the given file. No additional file is allowed.
2. Try to perform as much as you can. There is a plenty of scope for partial marking.
3. There will be a **viva evaluation** for this assignment. **Remember, your final marking of this assignment highly depends on the viva.**
4. Submit the single file which contains your solution. Any extra files apart from that will be discarded.
5. Submission deadline: September 25, 11:59 am

   Evaluation date & time: September 26, 2:30 pm

Assignment designed by Md. Rafi-Ur-Rashid, Lecturer, CSE, UIU