

Wyo Data Structures

Ch4Proj4.cpp

Modify Ch4Proj3.cpp with the following additional specifications. There are exactly three email subscribers with the username and password combinations george/wash, john/adam, and tom/jeff. An email subscriber's data must be stored in a struct named with the following definition:

```
struct User
{
    apstring username;
    apstring password;
    // apqueue <apstring> unreadMessages;
};
```

Note - The unreadMessages member of User should be commented out for this project.

You must also use the following global apvector to store the usernames and passwords:

```
apvector <User> subscribers(3);
```

The username and password combinations must be stored in a sequential access file named subscribers.txt using the following format where the first record is the number of username and password combinations in the file:

3
george
wash
john
adam
tom
...

Also use the sequential access files unreadgeorge.txt, unreadjohn.txt, and unreadtom.txt in place of the Ch4Proj3 unread.txt file to separately store the unread messages for each user. You may use the c_str member function of the apstring class to efficiently work with these separate files as in:

```
apstring filepath;
filepath = "Z:\\Data Structures\\Ch 4\\unread" + username +
    ".txt";
unreadMessageFile.open( filepath.c_str() );
```

You must also use a function named login that validates a user's entered username and password.

some of the Ch4Proj3 pseudocode:

- read subscriber username and passwords from subscribers.txt and store usernames and passwords into the subscribers apvector.
- prompt and get the user's username and password
- use the login function to determine if the inputted username and password combination is valid
- if the user logged in successfully, allow the user to send and receive messages (to himself as in Ch4Proj3)