

**CODE:**

```
#include <iostream>

#include <string>

using namespace std;

string encryptMessage(const string &message, int key){

    string encryptedMessage = message;

    for (char &ch : encryptedMessage){

        if (isalpha(ch)){

            char base = isupper(ch) ? 'A' : 'a';

            ch = static_cast<char>((ch - base + key) % 26 + base);

        }

    }

    return encryptedMessage;

}

class InstantMessagingAccount{

private:

    string username;

    string password;

public:

    InstantMessagingAccount(const string &username, const string &password)

        : username(username), password(password) {}

    void login(const string &enteredUsername, const string &enteredPassword){
```

```

        if (enteredUsername == username && enteredPassword == password){

            cout << "Login successful!" << endl;

        }

        else{

            cout << "Login failed. Invalid username or password." << endl;

        }

    }

};

```

```

int main(){

    InstantMessagingAccount myAccount("jay_madane", "password");

    string enteredUsername, enteredPassword;

    cout << "Enter your username: ";

    cin >> enteredUsername;

    cout << "Enter your password: ";

    cin >> enteredPassword;

    myAccount.login(enteredUsername, enteredPassword);

    string message;

    int encryptionKey;

    cout << "Enter the message you want to send: ";

    cin.ignore(); // Ignore the newline character left in the buffer

    getline(cin, message);

    cout << "Enter the encryption key: ";

    cin >> encryptionKey;

```

```
string encryptedMessage = encryptMessage(message, encryptionKey);  
  
cout << "Encrypted message: " << encryptedMessage << endl;  
  
return 0;  
  
}
```

**OUTPUT:**

Enter your username: jay\_madane

Enter your password: password

Login successful!

Enter the message you want to send: this is a secret message

Enter the encryption key: 3

Encrypted message: wklv lv d vhfuhw phvvdjh