TU Dublin City Campus,
KEVIN STREET, DUBLIN 8.

CLIENT SERVER PROGRAMMING (CMPU3006)

Mr. D. Bourke

Stage 3

CONTINUOUS ASSESSMENT 2020/21

Due date: Friday, 4th December 2020, 11:59 am

This assessment accounts for 50% of the overall CA mark for this module.

Refer to the separate file, ***authClient.c***, which can be downloaded from the *Assignment* menu on Brightspace. This file contains the source code for the client component of a **Basic User Authentication System**. You are required to write the **server** component of this system in *C* using the Berkeley Socket API.

Name the source file "authServer.c" and **INCLUDE YOUR NAME AND STUDENT NUMBER IN THE FIRST LINE OF THIS FILE AS A COMMENT**.

When complete, upload the source file (*authServer.c*) onto the Brightspace platform.

Study the client code and determine the data exchange requirements for this system i.e. the ***Syntax*** and ***Semantics***. The following is the minimum functionality required of the server:

- The basic operation of this application is that the client makes a connection request to the server.
- In normal operation the client will request text input from the local user. The user is prompted to enter a *Username* and *Password*.
- The client application constructs a string containing the "username" and "password" and transmits the string to the server application.
- The **server** application should read the incoming byte-stream and be able to separate-out (parse) the "username" and "password" components into separate buffers.
- The **server** application should verify these components against local strings containing the correct username ("admin") and password ("pass").
- The **server** application should return **one** of the following messages for each submission by the client application i.e. for each attempt:
  - PROCEED
  - You have x attempt(s) left
  - DENIED

  The DENIED response should only be sent after **three** unsuccessful attempts.

  If PROCEED or DENIED is returned to the client application, the server application should close the connection and wait for the next connection. Otherwise the server application should return to reading the next attempt from the client application until a maximum of three attempts have been received.

(100 Marks)

Marks will be awarded for completeness, robustness and readability of the code so be sure to verify each step especially where data processing is expected, indent your code for readability and add comments.

Notes on running your code for testing purposes:

The client source code file (*authClient.c*) should be downloaded onto Virtual Machine 1 (VM1 has IP address: 192.168.1.11). It should be compiled using the *compile* script previously developed in lab. Once compiled, the client application can be used for testing against your server application.

The server application should be developed on Virtual Machine 2 (VM2 has IP address: 192.168.1.12). To test the *server* application, it must be compiled using the *compile* script. This command assumes that both the *compile* script and the server source code file are in the *same* directory:

./compile authServer

To run the server application, use the following command. This command assumes that Port **7000** is available:

./authServer 7000

To run the client application, use the following command:

./authClient 192.168.1.12 7000

Ensure to upload the “authServer.c” file to Brightspace.