National University
of computer and emerging sciences

# Computer Networks - Fall 2021
# Assignment No. 2

**Submission Deadline: 20th October, 2021 - 11:59pm (on google classroom)**

## Submission Guidelines

1. You can code your programs in either c or c++.
2. Late submissions will be marked zero. 1 minute late is also LATE.
3. Only zipped files are acceptable. Any other format will be marked zero.
4. Make two folders for Q1 and Q2. Zip them both together and submit your zipped file with the naming convention: RollNo_YourName_Section.zip
5. Plagiarism in any of the questions will lead to zero marks in the whole assignment category.

## Question No. 1:

**Write a Program in C/C++ [Handling Multiple Clients using UDP] for the below-given scenario.**

**ATM:** An automated teller machine **(ATM)** or cashpoint is an electronic telecommunications device that enables customers of financial institutions to perform financial transactions, such as cash withdrawals, deposits, funds transfers, or account information inquiries, at any time and without the need for direct interaction with bank staff.

The main steps involved in an ATM are:

- Card Reading
- Pin Verification
- Cash Withdraw

These steps are performed in the given sequence. Your task is to simulate it in a client-server architecture.

### PART A (Server-Side)

To simulate this, you need to design a server program that will be responsible to provide the smooth functionality of the ATM. The server will perform the tasks below:

- There are three numbers **1, 2, 3** associated with their functionalities and **exit** to terminate the ATM function.
    - 1 - Correspond to Card Reading
    - 2 - Pin Verification
    - 3 - Cash Withdraw
    - 0 - Exit
- ATM functions should be performed in the sequence that is given. The server will initiate the process and assign the tasks to the three clients.
- The client will check the status of the current task and work accordingly. For example, if client **A** completed its execution, next client **B** should check the status of the previously completed execution and then move on. (It can ask from the server, before the start of its execution)
- There should be a main menu from where the user will request ATM withdrawal. There is no need to store the customer records. Rather, your server will work as: once a user selects the option for withdrawal, the server will generate a number from 0 to 50000 (through random function) that will represent the current amount in the customer account. Then, the server will ask the customer how much amount he wants to withdraw, and

according to the current balance you need to withdraw the amount accordingly. Also, the amount of withdrawal cannot exceed 25000. (Hint: You first need to check the current balance and then add a further check for withdrawal limit)

- The server should generate a random number from 0 to 3. If the number generated is 0, the system should report an error message **"Transaction Failed"** And then get another random number (>0).`
- . For 1 to 3 you need to assign it to the 3 clients respectively.
    - Once the server gets the request, it should ask the active clients to further proceed with the process. The client should be accepting here. Once the server gets all of the **3** clients' requests, it will generate a random number as mentioned in the above point. The server will send the number to each client randomly. **You need to check that there should not be any repetition in the random numbers.** (For example, two clients don't get a **2**).
- Once all the clients send the acceptance status, the server should assign the task to the clients according to the sequence number.
- Once the task is completed, the server should print out the IP addresses of each client along with the output of the task that is performed by the client.

### PART B (Client-Side)

- Once all the tasks are assigned to the clients in sequence, each client should perform the required functionality, send the output to the server. For example, the first client will send card read status. The second will send the pin verified status, and the third will handle the deduction of cash from the current balance, update it, and show the stats accordingly.
- For the functionality of clients **A** and **B,** you are free to choose any strategy for the pin and card verification and that will be asked at the time of demo.

**Note:** Write a 1-page report to explain how your code works. Mention your strategy used for the functionality implemented.

---

## Question No. 2:

**Create a client-server program [TCP using fork()] to connect 5 clients simultaneously.**
As soon as the server starts, it displays a menu containing the following options:

1. **Concatenate two strings**
   Client enters 1 and server asks for two strings to be concatenated. Server concatenates two strings and returns the resulting string to the client.
2. **Find the first occurrence of a character in a string**

Client enters 2 and server asks for string and a character to be checked. Server returns the position of the character in a string and if the character is not present, it returns -1.

3. **Compares two strings**

   Client enters 3 and server asks for two strings to be compared. Server compares two strings and returns ‘True’ if two strings are the same irrespective of their case (case insensitive) else ‘False’.

4. **Checks whether string is palindrome or not**

   Client enters 4, server asks for a string, client sends the string, server checks whether the string is a palindrome, and sends the result to the client, client displays it.

5. **String Number Addition**

   Client enters 5, enters two numbers to be added in string, and takes answer in string i.e., “Two” and “Three” answer must be “Five”. You have to make the addition of first 10 numbers possible.

**Note: For this question, all 4 clients must be connected simultaneously. Any client can send any number. After one client has finished its work, it should terminate.**