

**ĐẠI HỌC BÁCH KHOA HÀ NỘI**  
**TRƯỜNG CNTT & TT**



# **BÁO CÁO BÀI TẬP LỚN**

*Bộ môn: Thực hành Lập trình mạng*

*Đề tài: Ứng dụng thi trắc nghiệm trực tuyến*

Giảng viên hướng dẫn: PGS. Trương Thị Diệu Linh

Sinh viên thực hiện:

Đào Việt Chiến	20184050
Phạm Văn Tú	20184215
Lê Thế Tài	20184190

## Mục lục

<b>I. Mô tả</b>	<b>3</b>
<b>II. Cơ sở lý thuyết</b>	<b>3</b>
1. TCP và hàm fork()	3
2. Các hàm sử dụng	5
<b>III. Xây dựng hệ thống</b>	<b>6</b>
1. Mô tả cách thức hoạt động	6
2. Thiết kế giao thức	7
2.1 Học sinh	8
2.2 Giáo viên	8
3. Dữ liệu	9
<b>IV. Cài đặt và kiểm thử</b>	<b>10</b>
1. Cài đặt	10
2. Kiểm thử	10
<b>V. Phân công công việc</b>	<b>11</b>
<b>VI. Tổng kết</b>	<b>11</b>

## **I. Mô tả**

- Do ảnh hưởng của dịch bệnh, việc ứng dụng Công nghệ vào thi cử sẽ giúp cho các thầy cô bớt vất vả hơn trong sự nghiệp trồng người.
- Đề tài lấy ý tưởng từ việc trong trường vẫn sử dụng thi trắc nghiệm bằng giấy và chấm điểm bằng tay. Vì vậy, nhóm chúng em xây dựng một chương trình hỗ trợ việc ra đề thi và chấm thi một cách nhanh chóng hiệu quả hơn.
- Ngôn ngữ sử dụng: C; giao thức: TCP

Nhóm chúng em cũng xin cảm ơn cô Trương Thị Diệu Linh đã tận tình chỉ bảo trong môn học Thực hành Lập trình Mạng để nhóm chúng em có được những hiểu biết cần thiết cho việc lập trình chương trình này.

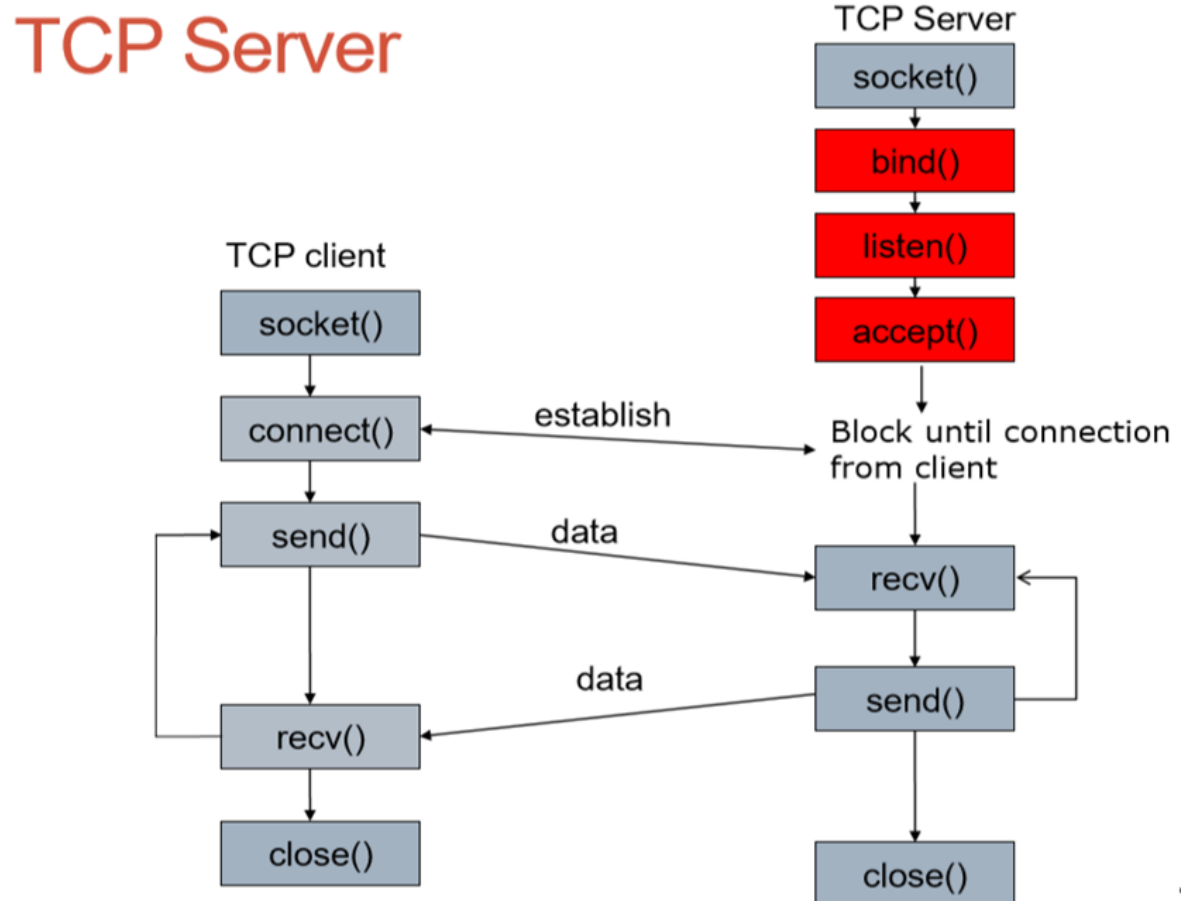
## **II. Cơ sở lý thuyết**

### **1. TCP& hàm fork()**

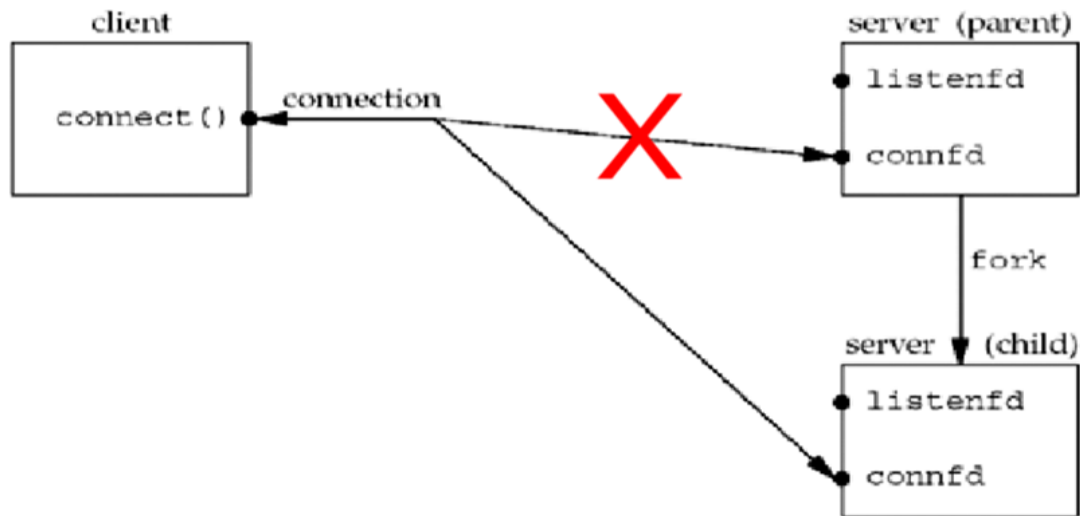
- TCP (Transmission Control Protocol - "Giao thức điều khiển truyền vận") là một trong các giao thức cốt lõi của bộ giao thức TCP/IP. Sử dụng TCP, các ứng dụng trên các máy chủ được nối mạng có thể tạo các "kết nối" với nhau, mà qua đó chúng có thể trao đổi dữ liệu hoặc các gói tin. Giao thức này đảm bảo chuyển giao dữ liệu tới nơi nhận một cách đáng tin cậy và đúng thứ tự. TCP còn phân biệt giữa dữ liệu của nhiều ứng dụng (chẳng hạn, dịch vụ Web và dịch vụ thư điện tử) đồng thời chạy trên cùng một máy chủ [1,2]
- Trong mô hình client/server một trình chủ có thể phục vụ đồng thời cho nhiều trình khách. Hàm accept() chờ kết nối đến, xử lý xong kết nối rồi mới quay lại nhận kết nối tiếp theo. Đây là cách xử lý tuần tự và thường không phù hợp với việc nhiều trình khách yêu cầu phục vụ cùng lúc.

- Ta có thể sử dụng lệnh `fork()` để kiến tạo tiến trình con mới. Tiến trình con mới này hoạt động độc lập với trình chủ và chịu trách nhiệm phục vụ trình khách theo cách riêng của nó. Trình chủ hoàn toàn tự do để tiếp nhận ngay kết nối khác.

## Sơ đồ TCP và `fork()`



## Mô hình TCP Server



### Mô hình fork()

## 2. Các hàm sử dụng

### · Phía Server

- `socket()`: Tạo socket để kết nối
- `bind()`: Gán địa chỉ cho socket
- `listen()`: Thiết lập socket để chờ kết nối
- `accept()`: Chấp nhận kết nối. Bị block cho đến khi có kết nối mới tới
- `recv()`, `send()`: Gửi và nhận dữ liệu
- `close()`: Đóng socket

### · Phía Client

- `Socket()`: Tạo socket để kết nối

- Connect():Kết nối tới server
- Send(), recv(): Gửi và nhận dữ liệu
- Close(): Đóng socket

### **III. Xây dựng hệ thống**

#### **1. Mô tả cách thức hoạt động**

Bắt đầu thực hiện từ bước gửi và nhận thông tin giữa client và server.

##### **1. Các chức năng của chương trình**

###### **Đăng nhập**

Server yêu cầu client đăng nhập

Client gửi chuỗi tên đăng nhập và mật khẩu cho server

Server nhận chuỗi tên đăng nhập và mật khẩu.

Server kiểm tra tài khoản và mật khẩu trong file acc\_stud.txt. hoặc acc\_teacher.txt

###### **Login Student**

Nếu login thành công server sẽ show ra các phòng đang tồn tại

Nếu sai yêu cầu client login lại từ đầu

###### **Login teacher**

Nếu login thành công client sẽ show ra menu chức năng cho giáo viên

- Show room
- Create room
- Delete room
- Show point
- Add question

Nếu sai yêu cầu client login lại từ đầu

### Xử lý bộ câu hỏi

File câu hỏi cho từng phòng riêng biệt do giáo viên thêm đề vào, gồm các câu hỏi với 3 mức độ: Dễ, Trung bình, Khó.

Đề bài gồm 10 câu hỏi với 3 mức độ. Dễ, Trung Bình, Khó được lấy ngẫu nhiên file câu hỏi.

Sau khi chọn phòng thi, server gửi từng câu hỏi của đề bài cho client. Khi client trả lời, server sẽ gửi câu hỏi tiếp theo cho client.

Client nhận đề bài, trả lời và gửi đáp án lại cho server

### Xử lý điểm

Client nhập đáp án trả lời từng câu hỏi

Server nhận đáp án và tiến hành đối chiếu với chuỗi đáp án đúng.

Với 1 câu trả lời đúng được cộng 1 điểm. Trả lời sai không bị trừ điểm. (Tối đa được 10 điểm. Tối thiểu được 0 điểm)

Server gửi lại kết quả cho client.

### Xử lý phòng thi

Show room: Server sẽ gửi cho giáo viên các phòng thi đang tồn tại

Create room: Giáo viên nhập tên phòng gửi lên server phòng

Delete room: Giáo viên nhập tên phòng gửi lên server xoá phòng

Show Point: Giáo viên nhập tên phòng gửi lên server. Server gửi kết quả của phòng thi về client (tên học sinh - điểm)

Add Question: Giáo viên nhập tên phòng thi và path file đề. Server nhận thông tin và thêm file đề thi cho phòng tương ứng.

## **2. Thiết kế giao thức**

## 2.1 Học sinh

Chức năng	Client	Server
1. Đăng nhập	LOGIN_STUDENT <username> <password>	- Nếu thành công LOGIN_STUDENT_OK <danh sách phòng>  - Nếu thất bại LOGIN_STUDENT_NOT_OK
2. Tham gia phòng	JOIN_ROOM <room name>	- Nếu thành công JOIN_ROOM_OK  - Nếu thất bại JOIN_ROOM_NOT_OK
3. Bắt đầu bài làm	START_TEST  <câu trả lời 1> <câu trả lời 2> ... <câu trả lời 10>	- Nếu thành công START_TEST_OK <câu hỏi 1>  <câu hỏi 2> ... <câu hỏi 10>  - Nếu thất bại START_TEST_NOT_OK
4. Trả lời câu hỏi	ANSWER	- Nếu thành công ANSWER_OK <điểm>  - Nếu thất bại ANSWER_NOT_OK

## 2.2 Giáo viên

Chức năng	Client	Server
1. Đăng nhập	LOGIN_TEACHER <username> <password>	- Nếu thành công LOGIN_TEACHER_OK



		- Nếu thất bại LOGIN_TEACHER_NOT_OK
2. Xem danh sách phòng	SHOW_ROOM	- Nếu thành công SHOW_ROOM_OK  - Nếu thất bại SHOW_ROOM_NOT_OK
3. Tạo phòng	CREATE_ROOM <room name>	- Nếu thành công CREATE_ROOM_OK  - Nếu thất bại CREATE_ROOM_NOT_OK
4. Xoá phòng	DELETE_ROOM <room name>	- Nếu thành công DELETE_ROOM_OK  - Nếu thất bại DELETE_ROOM_NOT_OK
5. Thêm đề	ADD_QUESTION <room_name> <file_content>	- Nếu thành công ADD_QUESTION_OK  - Nếu thất bại ADD_QUESTION_NOT_OK
6. Xem điểm 1 phòng	SHOW_POINT <room_name>	- Nếu thành công SHOW_POINT_OK  - Nếu thất bại SHOW_POINT_NOT_OK

### 3. Dữ liệu

- File “acc\_stud.txt” : Chứa dữ liệu về thông tin các tài khoản học sinh
- File “acc\_teacher.txt” : Chứa dữ liệu về thông tin các tài khoản giáo viên
  - Lưu trữ theo cú pháp: Tên đăng nhập Mật khẩu
  - Ví dụ: taile 12345
  - Thư mục “question” : Chứa dữ liệu về thông tin các câu hỏi cho từng phòng
    - Lưu trữ theo cú pháp: STT|Mức độ|Câu hỏi|Đáp án A|Đáp án B|Đáp án C|Đáp án D|Đáp án chính xác

- Ví dụ: 1|EASY|Lãnh tụ vĩ đại của Việt Nam là ai?|A. Hồ Chí Minh|B. Lenin|C. Võ Nguyên Giáp|D. Trần Đại Nghĩa|A
- Trong đó: Mức độ bao gồm 3 loại: EASY, NORMAL, HARD
- Thư mục “client-ques” : Chứa các đề bên client để test chức năng thêm đề
- Thư mục “result”: Chứa kết quả thi của từng phòng

#### IV. Cài đặt và kiểm thử

##### 1. Cài đặt

- Yêu cầu : Máy chủ cài đặt trình biên dịch C
- Các bước cài đặt: Mở cửa sổ Terminal và gõ các dòng lệnh:
  - git clone https://github.com/upabs/20211\_LTM
  - make

##### 2. Kiểm thử

Chức năng	INPUT	OUTPUT
LOGIN_STUDENT	LOGIN_STUDENT student1 111	_____List room_____ room1 room2 1night1A
LOGIN_STUDENT	LOGIN_STUDENT student1 222	_____ HOME _____  1. Login Student 2. Login Teacher 0. Exit
LOGIN_TEACHER	LOGIN_TEACHER teacher1 111	1. SHOW ROOM     2. CREATE ROOM     3. DELETE ROOM     4. SHOW POINT     5. ADD QUESTION     0. EXIT
LOGIN_TEACHER	LOGIN_TEACHER teacher1 222	_____ HOME _____  1. Login Student 2. Login Teacher 0. Exit
JOIN_ROOM	JOIN_ROOM room1	Do you want start test?

		1. Yes 2. No
START TEST	START TEST	<Câu hỏi>
SHOW_ROOM	SHOW_ROOM_OK room1 room2 1night1A..	room1 room2 1night1A <Menu>
CREATE_ROOM	CREATE_ROOM room3	[CREATE_ROOM_OK] <Menu>
CREATE_ROOM	CREATE_ROOM room3	[CREATE_ROOM_NOT_OK] <Menu>
DELETE_ROOM	DELETE_ROOM room3	[DELETE_ROOM_OK] <Menu>
DELETE_ROOM	DELETE_ROOM room3	[DELETE_ROOM_NOT_OK] <Menu>
SHOW_POINT	SHOW_POINT room1	_____Resutl_____
		student1      2 student2      0 <Menu>
SHOW POINT	SHOW POINT room3	<Menu>
ADD_QUESTION	ADD_QUESTION room_name file_content	[ADD_QUESTION_OK]
ADD_QUESTION	ADD_QUESTION room_name file_content	[ADD_QUESTION_NOT_OK]

## V. Phân công công việc

### Thành viên

### Nhiệm vụ thực hiện

Lê Thế Tài

Đăng nhập teacher, Thêm đề thi, Kết nối Ngrok,  
viết báo cáo

Phạm Văn Tú

Đăng nhập Student, Tham gia phòng thi, Trả lời câu  
hỏi, viết báo cáo

Đào Việt Chiến

Quản lý tiến độ chung của nhóm, tạo room, xóa room,  
xem danh sách phòng thi, xem kết quả phòng thi

## **VI. Tổng kết**

### **1. Những việc đã thực hiện**

- Sử dụng giao thức TCP để thực hiện kết nối giữa server và client
- Đạt được những chức năng cơ bản của yêu cầu đề ra
- Tài liệu tham khảo: Slide môn học

### **2. Hướng phát triển**

Tiếp tục hoàn thiện thêm các chức năng của học sinh và giáo viên như quay lại câu hỏi trước, sửa lại đáp án, ... và xây dựng giao diện để người dùng có thể dễ dàng thao tác.