BÀI TẬP THỰC HÀNH: THIẾT KẾ APPLICATION PROTOCOL CHO TRÒ CHƠI CỜ CA-RÔ SỬ DỤNG TCP SOCKETS

**Môn học**: TH Lập Trình Mạng  
**Chương 7**: Thiết kế Application Protocol

**1. Mục tiêu**

* Thực hành thiết kế giao thức truyền thông giữa server và client cho trò chơi Cờ ca-rô nhiều người chơi đơn giản.
* Cài đặt trò chơi bằng cách trao đổi thông điệp giữa client và server sử dụng giao thức đã thiết kế.

**2. Định dạng thông điệp (Protocol)**

Mỗi thông điệp gồm:

* **Loại Thông Điệp** (1 byte)
* **Payload** (n bytes, tùy loại)

| **Loại Thông Điệp** | **Mã Hex** | **Vai Trò** | **Payload** |
| --- | --- | --- | --- |
| MOVE | 0x02 | Client gửi nước đi (hàng, cột) | [row (1 byte)] [col (1 byte)] |
| STATE\_UPDATE | 0x03 | Server gửi cập nhật bảng cho cả hai client | [9 bytes: trạng thái 3x3] |
| RESULT | 0x04 | Server thông báo kết quả (win/lose/draw) | [1 byte: 1=X thắng, 2=O thắng, 3=hòa] |
| TURN\_NOTIFICATION | 0x05 | Server thông báo đến lượt client | [1 byte: người chơi 1=X, 2=O] |

**3. Luồng trò chơi (Game Flow)**

1. **Khởi động server**: lắng nghe kết nối trên cổng 8080.
2. **Kết nối hai client**: chấp nhận hai kết nối TCP.
3. **Vòng chơi** (lặp lại đến khi kết thúc):
   1. Server gửi TURN\_NOTIFICATION cho client hiện tại (thêm thông tin người chơi).
   2. Client nhận thông điệp và nếu đúng lượt, gửi MOVE (row, col).
   3. Server kiểm tra hợp lệ, cập nhật game\_board và tăng move\_count.
   4. Server gửi STATE\_UPDATE (9 bytes) cho cả hai client.
   5. Server kiểm tra điều kiện thắng/hòa.
      * Nếu kết thúc, gửi RESULT cho cả hai client và đóng kết nối.
   6. Đổi lượt và tiếp tục.

**4. Cấu trúc dữ liệu (Data Structures)**

#define EMPTY 0

#define PLAYER\_X 1 // Người chơi 1

#define PLAYER\_O 2 // Người chơi 2

int game\_board[3][3]; // 0=trống, 1=X, 2=O

int move\_count = 0;

Các hàm chính:

* void print\_board(int board[3][3]); // In bảng ra console.
* int check\_winner(int board[3][3]); // Trả về 1 hoặc 2 nếu có người thắng, 3 nếu hòa, 0 nếu chưa kết thúc.

**5. Mã nguồn Server (server.c)**

...

// Gửi TURN\_NOTIFICATION (kèm người chơi)

buffer[0] = 0x05;

buffer[1] = current\_player;

send(turn\_sock, buffer, 2, 0);

...

**6. Mã nguồn Client (client.c)**

...

if (type == 0x05) {

int player = buffer[1];

printf("It's your turn! You are player %c\n", player == 1 ? 'X' : 'O');

int row, col;

print\_board(board);

printf("Your move (row col): ");

scanf("%d %d", &row, &col);

buffer[0] = 0x02;

buffer[1] = (unsigned char)row;

buffer[2] = (unsigned char)col;

send(sock, buffer, 3, 0);

}

...

**7. Kết quả thử nghiệm**

* **Kịch bản 1**: Người chơi X thắng theo hàng ngang.
  + Ảnh chụp màn hình server: screenshot\_server.png
  + Ảnh chụp màn hình client X: screenshot\_client1.png
  + Ảnh chụp màn hình client O: screenshot\_client2.png

**8. Những thách thức và giải pháp**

* **Xử lý nước đi không hợp lệ**: Cần kiểm tra chỉ số hàng/cột và ô trống. Giải pháp: nếu không hợp lệ, server bỏ qua và yêu cầu client gửi lại.
* **Đồng bộ trạng thái giữa hai client**: Luôn gửi STATE\_UPDATE cho cả hai sau mỗi nước đi.
* **Đóng kết nối đúng lúc**: Khi trò chơi kết thúc, phải gửi RESULT và sau đó close() cả hai socket.

ảnh chụp:

Người chơi O:  
A screenshot of a computer

AI-generated content may be incorrect.

Người X:  
A screenshot of a computer

AI-generated content may be incorrect.  
server

A screenshot of a computer

AI-generated content may be incorrect.