

# Tic Tac Toe Game

Socket Programming을 이용하여 컴퓨터 네트워크 통신을 이용한 Tic Tac Toe 게임

2271104 최수희

2271106 박서현

TCP 연결을 기반으로 한 소켓 프로그래밍을 통해 서버와 클라이언트가 통신할 수 있다.

TCP는 어플리케이션 간의 데이터 전송을 위해서 안정적이고 신뢰성 있는 연결을 제공하는 프로토콜이다. 패킷을 순서대로, 손실 없이, 중복 없이 전달하는 것을 보장한다. 소켓 프로그래밍을 네트워크 상의 두 컴퓨터가 데이터를 주고받을 수 있도록 하는 프로그래밍 기술이다.

#### Socket Programming 설명

1. 서버 소켓 생성 : 서버는 클라이언트의 연결을 요청 받기 위해서 서버 소켓을 생성한다. 서버 소켓은 특정 포트 번호에 바인딩 되어 클라이언트의 연결을 기다린다.
2. 클라이언트 소켓 생성 : 클라이언트는 서버에 연결하기 위해 클라이언트 소켓을 생성한다. 클라이언트 소켓을 서버의 IP 주소와 포트 번호를 지정하여 서버에 연결한다.
3. 연결 : 클라이언트는 서버 IP 주소와 포트 번호를 이용해서 서버에 연결을 요청하고, 서버는 이를 수락하여 클라이언트와 연결한다.
4. 데이터 송수신 : 클라이언트와 서버는 소켓을 통해 데이터를 주고 받는다. 클라이언트는 서버에 데이터를 전송하고, 서버는 클라이언트로부터 데이터를 수신하고 처리한다.
5. 연결 종료 : 클라이언트 또는 서버가 연결을 종료하고자 할 때, 소켓을 닫아 연결을 해제한다.

#### 서버 코드에 대한 설명

서버 소켓을 생성하고 클라이언트와의 연결을 처리한다.

랜덤으로 선 플레이어를 선택하고, 이를 클라이언트에게 전송한다.

클라이언트로부터 수신한 ACK 메시지가 올바르게 확인하고, 게임을 시작한다.

게임이 종료되면 소켓을 닫고 프로그램을 종료한다.

#### 클라이언트 코드에 대한 설명

클라이언트 소켓을 생성하고 서버에 연결하여 서버로부터 누가 먼저 시작할지 정보를 수신한다.

수신한 메시지를 검증하고 ACK 메시지를 서버에 전송한 후, 게임을 시작한다.

게임이 종료되면 소켓을 닫고 프로그램을 종료한다.

## Tic Tac Toe 코드에 대한 설명

- Get\_move 함수 : 다른 피어로부터 받은 이동 정보를 처리하는 역할을 한다.
  1. 메시지 수신 : 소켓을 이용하여 다른 피어로부터 메시지를 수신한다.
  2. 메시지 유효성 검사 : 수신한 메시지가 ETTTP 형식이 맞는지 확인한다. 유효하지 않으면 소켓을 닫고 프로그램을 종료한다.
  3. 좌표 추출 : 수신한 메시지에서 좌표를 추출하여 row와 col에 저장한다.
  4. ACK 메시지 생성 및 전송 : 추출한 좌표를 포함하는 ACK 메시지를 생성하고, 이를 다른 피어로 전송한다.
  5. 보드 위치 계산 : 좌표를 이용하여 보드에서의 위치를 계산한다.
  6. 보드 업데이트 및 턴 변경 : 보드를 업데이트하고, 게임 상태가 활성 상태이면 턴을 변경하여 자신의 턴으로 만든다.
  
- Send\_debug 함수 : 텍스트 박스에서 입력된 메시지를 피어에게 전송하는 역할
  1. 현재 사용자의 턴을 확인 : 현재 사용자의 턴이 아니면 텍스트 박스를 비우고 함수를 종료한다.
  2. 메시지 가져오기 : 텍스트 박스에서 메시지를 가져와 줄 바꿈 문자를 올바르게 변환한다.
  3. 위치 확인 : 메시지를 피싱하여 선택된 위치가 이미 차 있는지 확인한다. 이미 차 있으면 함수를 종료한다.
  4. 메시지 전송 : 피어에게 메시지를 전송한다.
  5. ACK 메시지 수신 : 피어로부터 ACK 메시지를 수신하고, 유효성 검사를 한다. 유효하지 않으면 게임을 종료한다.
  6. 보드 업데이트 : 메시지가 유효하면 보드를 업데이트 하고, 턴을 상대방으로 변경하여 상대방의 움직임을 기다린다.
  
- Send\_move 함수 : 버튼 클릭을 통해 선택된 이동을 피어에게 전송하는 역할을 한다.
  1. 좌표 계산
  2. 메시지 생성 : 선택된 버튼의 위치를 포함해 send 메시지를 생성한다.
  3. 메시지 전송 : 생성된 메시지를 소켓을 이용해 피어에게 전송한다.
  4. ACK 메시지 수신 : 피어로부터 ACK 메시지를 수신한다.
  5. 유효성 검사 : 수신한 ACK 메시지의 유효성을 검사하고 그 결과를 반환한다.

- Check\_result 함수 : 피어 간의 게임 결과를 비교하고 확인하는 역할을 한다.  
게임의 승자 정보를 비교하여 두 피어가 동일한 결과를 가지고 있는지 확인한다.  
Get 파라미터가 True일 때와 False 일 때가 다르게 작동한다.

Get 파라미터가 true일 때 : 상대방의 결과 메시지를 먼저 수신하고, 그 후에 자신의 결과 메시지를 전송한다.

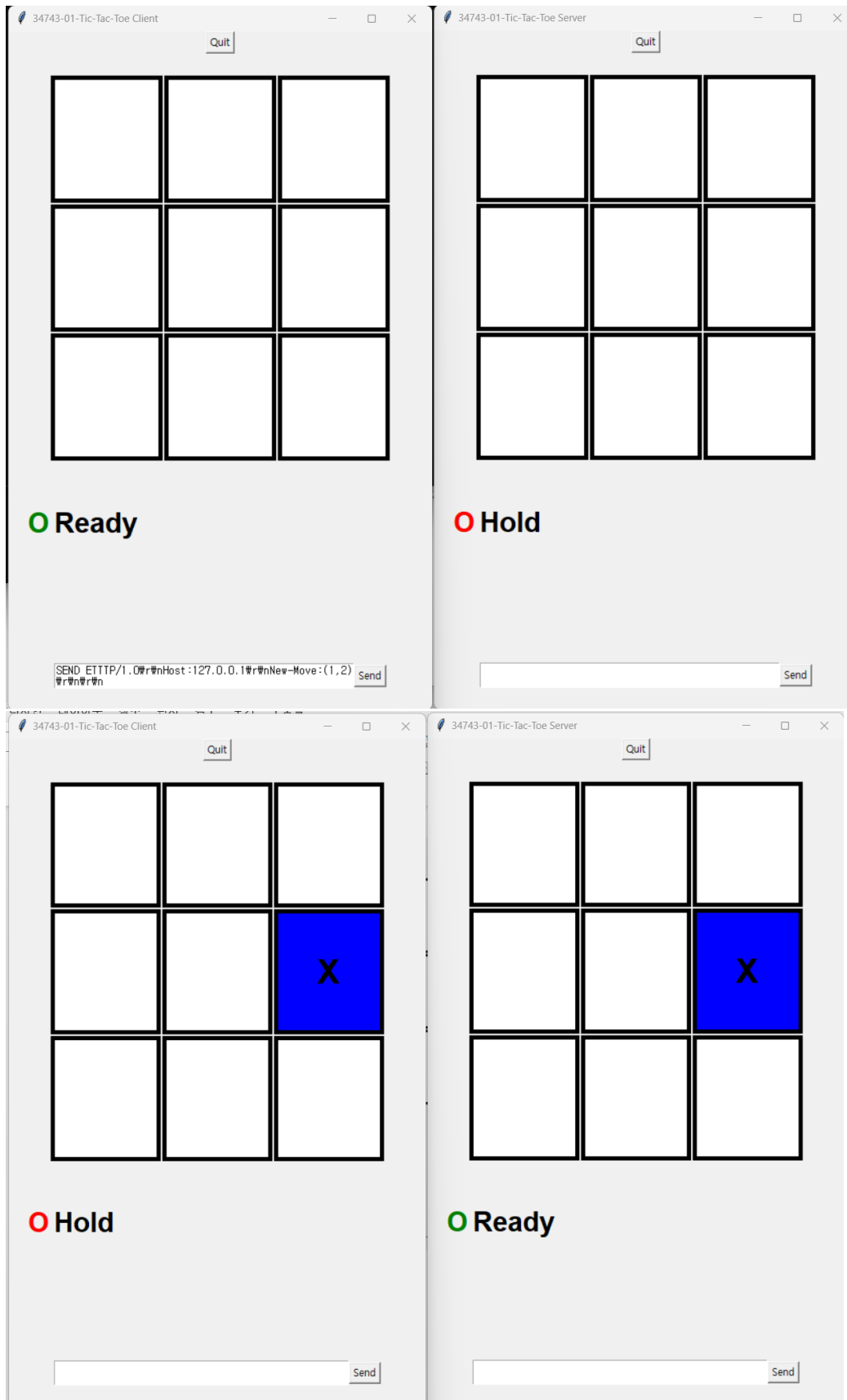
1. 상대방의 결과 메시지 수신 : 소켓을 통해 상대방으로부터 결과 메시지를 수신한다.
2. 유효성 검사 : 수신한 결과 메시지가 올바른 형식인지 검사한다.
3. 자신의 결과 메시지 생성 및 전송 : 자신의 결과 메시지를 생성하여 상대방에게 전송한다.
4. 상대방의 승자 정보 추출 : 수신한 결과 메시지에서 승자 정보를 추출한다.
5. 유효성 검사 및 승자 확인 : 메시지가 유효하고 승자 정보가 서로 일치하는지 확인한다.

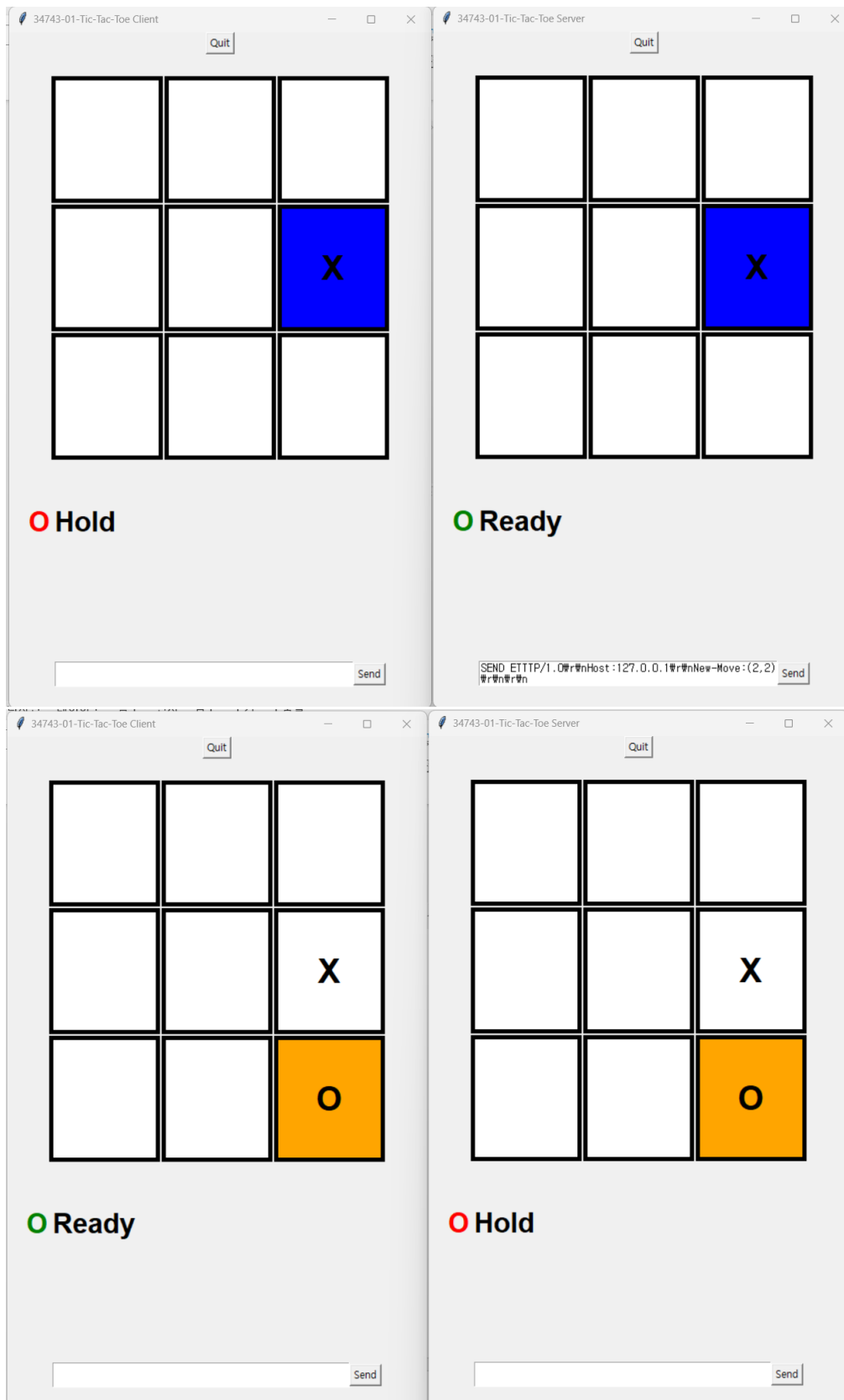
Get 파라미터가 false일 때 : 자신의 결과 메시지를 먼저 전송하고, 그 후 상대방의 결과 메시지를 수신한다.

1. 자신의 결과 메시지 생성 및 전송 : 자신의 결과 메시지를 생성하여 상대방에게 전송한다.
2. 상대방의 결과 메시지 수신 : 소켓을 통해 상대방으로부터 결과 메시지를 수신한다.
3. 유효성 검사 : 수신한 결과 메시지가 올바른 형식인지 검사한다.
4. 상대방의 승자 정보 추출 : 수신한 결과 메시지에서 승자 정보를 추출한다.
5. 유효성 검사 및 승자 확인 : 메시지가 유효하고 승자 정보가 서로 일치하는지 확인한다.

- Check\_msg 함수: 받은 메시지가 ETTTP형식이 맞는지 확인하는 역할을 한다.

## Send\_debug





## 마지막 결과

34743-01-Tic-Tac-Toe Client

Quit

		X
	X	X
O	O	O

**O Hold**

**Result: You Lost!**

Send

34743-01-Tic-Tac-Toe Server

Quit

		X
	X	X
O	O	O

**O Hold**

**Result: You Won!**

Send