

오목 규칙 및 프로토콜 설계

BoB 10th 보안제품개발트랙

김진욱

INDEX

01 Rules

02 Protocol

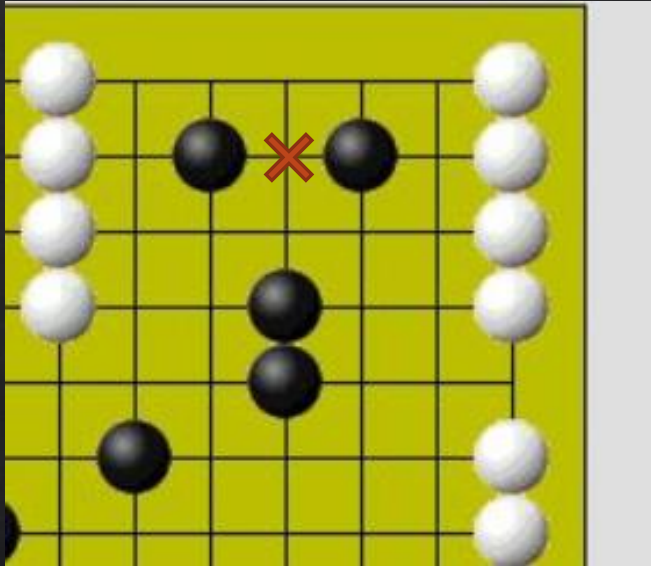
03 Flow-Chart

01. Rules

- 기본적인 규칙은 렌즈 룰을 사용한다.
 - 15*15 경기판 사용
 - 흑의 경우, 3-3, 4-4, 장목 불가능
 - 백의 경우 3-3, 4-4, 장목 가능
- 알고리즘 작성의 용이함을 위하여 3-3의 착수 금지점 중, 열린 4 조건 중 특수한 상황을 배제한다.

01. Rules

- 열린 4 조건이란?



- 렌주 룰에서 '3'이라 함은 한 수를 추가 했을 때 '열린 4' 를 만들 수 있는 상태를 말한다
- X자가 쳐져 있는 위치의 경우, 정식 렌주 룰에서는 3-3이 아니다.
- X위치에 돌을 둔다고 하더라도, 가로의 경우 한 수를 추가하여 열린 4를 만들 수 없기 때문에, 3이라고 볼 수 없음.

01. Rules

- 렌즈 룰도 흑돌이 약간 유리
- 열린 4 조건 중 이어진 3을 만드는 예시에서의 경우만 배제함으로써 흑돌에게 페널티를 추가
- 양쪽중 하나라도 막혀 있는 경우 3-3인지 판단할 필요가 없기 때문에, 알고리즘 작성에 용이함

01. Rules

- 시간 제한은 AI vs AI 게임의 경우 15초로 제한
- 서버에 먼저 연결되는 PC가 흑, 후 연결되는 PC가 백
- 2대의 클라이언트 까지만 서버에 연결 가능

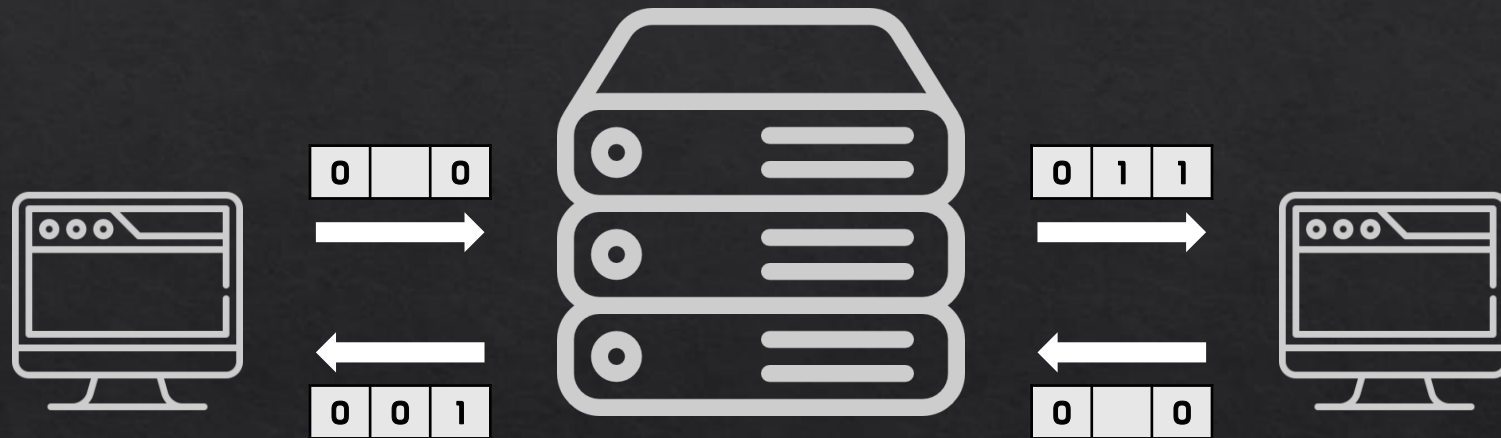
02. Protocol

- TCP Data를 이용하여 오목 프로토콜을 전달
- 프로토콜 size – 고정 3Bytes
 - Command(1Byte) : 어떤 과정을 수행하는지 확인
 - Turn(1Byte) : 차례 확인 or 승자 확인에 사용
 - Data(1Byte) : 각 명령별로 필요한 데이터

Command(1Byte)	Turn(1Byte)	Data(1Byte)
----------------	-------------	-------------

02. Protocol

Command 0: 접속(Connect)



Client → Server

- Command : 0
- Turn : 사용하지 않음
- Data : 접속 시도(0)

Server → Client

- Command : 0
- Turn(순서) : 첫번째 접속(0)
두번째 접속(1)
- Data(접속 여부) : 접속 성공(1)
접속 실패(2)

02. Protocol

Command 1: 준비(Ready)



Client -> Server

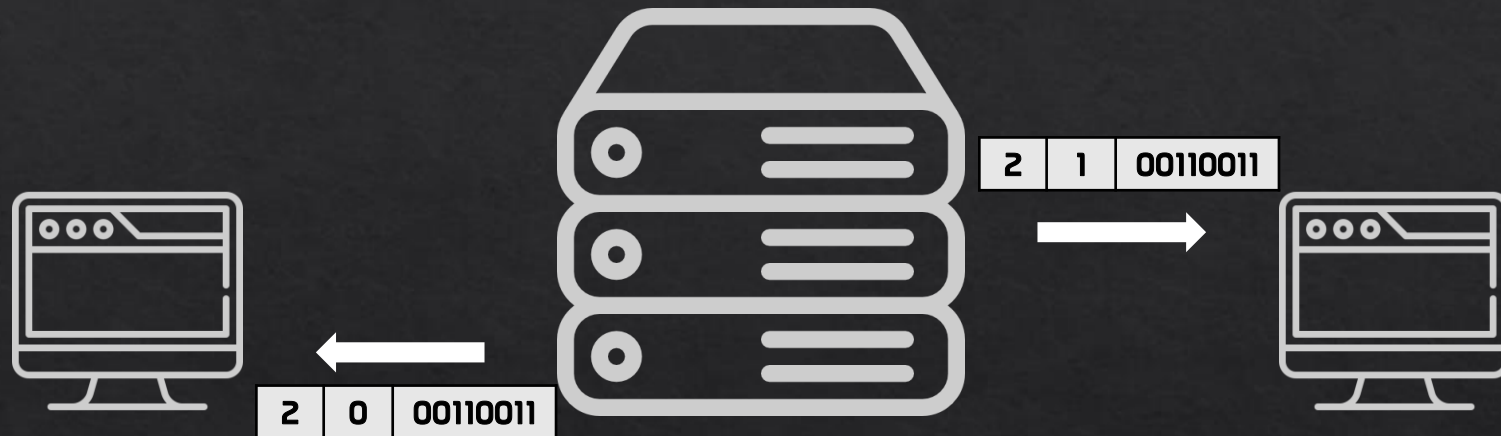
- Command : 1
- Turn : 흑돌(0), 백돌(1)
- Data(준비 여부) : 준비 완료 해제(0)
준비 완료(1)

Server -> Client

- 사용하지 않음

02. Protocol

Command 2: 갱신(Update)



Client → Server

- 사용하지 않음

Server → Client

- Command : 2
- Turn(차례) : 상대방 차례(0)
 자신 차례(1)
- Data(좌표) : 시작 시(0)
 돌을 두는 경우(상위 4bit x, 하위 4bit y)

02. Protocol

Command 3: 돌 두기(Put)



Client → Server

- Command : 3
- Turn : 사용하지 않음
- Data(좌표) : 상위 4bit x좌표, 하위 4bit y좌표
- 15초 제한

Server → Client

- 사용하지 않음

02. Protocol

Command 4: 종료(End)



Client → Server

- 사용하지 않음

Server → Client

- Command : 4
- Turn(승패) : 패배(0)
 승리(1)
- Data(사유) : 오류(0)
 시간 초과(1)
 오목 완성(상위 4bit x, 하위 4bit y)

03. Flow-Chart



Thank you.