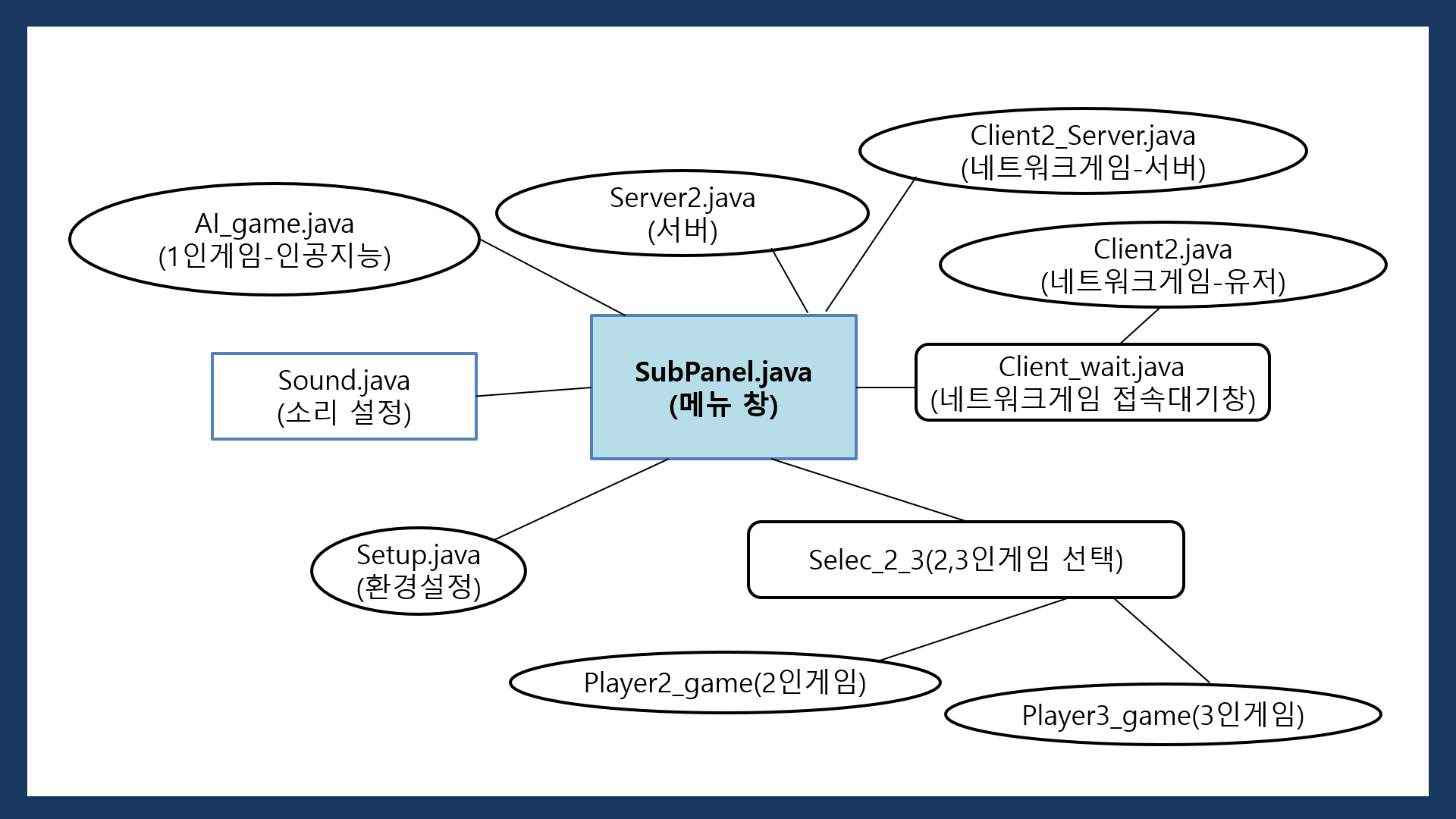
**미니 다이아몬드 게임 결과보고서**

201614868 이성민

코드 설계



1. 1인게임(AI\_game)

* 두개의 스레드(Timerthread, AI\_Control)를 사용했다. Timerthread는 환경설정에서 설정한 시간으로 구성된 타이머 역할을 하고, AI\_Control은 컴퓨터 turn이 되었을 때 스레드코드 내부에 있는 인공지능 알고리즘 코드를 실행한다.
* GameStatus 메서드를 이용해서 점수판을 출력한다. 기권했을때, ai또는 플레이어가 말을 다 옮겼을 때 점수판이 출력되게 설정했다.
* Mapping 메서드를 이용해서 말이 놓일 위치,색깔을 랜덤으로 정해 변수에 저장해준다. 그리고나서 stonedispose 메서드로 말을 패널 위에 배치한다. 배치관리자로 springlayout을 사용했다.
* Scan 메서드는 말주변을 탐색한다. 선택된 말이 이동할 수 있는 곳을 알려준다. 이동할 수 있는 곳을 player1\_null 이미지로 표시한다.
* NextJumpScan 메서드는 점프를 했는지 안했는지 판단을 한다. 점프를 했을 시 턴을 유지한다.
* Clear 메서드는 Scan메서드를 사용해 바뀐 이미지를 원래 빈칸(null\_image)으로 돌려놓는 역할을 한다.
* GameScore 메서드는 게임 스코어를 계산한다.
* NextTurn 메서드는 다음 턴으로 넘어갈 때 사용한다. 턴에 관련된 변수, turnstatus 등을 확인해서 변경해준다.
* 마우스 리스너를 통해 말이 움직이는 기능을 할 수 있도록 하였다. (1) 마우스가 눌러졌을 때 Scan메서드를 통해 고른 말이 이동할 수 있는 위치를 표시하도록 했고 (2) 말을 클릭했을 때는 말을 움직이지 않고 턴을 넘겨준다. (3) Scan을 해서 표시된 위치에 마우스가 올라갔을 때 이미지를 바꿔서(player1\_null\_p) 표시한다. 다시 그 위치에서 벗어나면 원래대로 수정해준다. (4) (3)에서 바뀐 이미지(player1\_null\_p) 위에서 마우스를 떼게 되면 말을 그 위치로 옮긴다. 콤보를 계산하고, 다음 점프가 가능하면 턴을 넘기지 않는다.

1. 2,3인 게임은 거의 기능이 동일 하므로 넘어가겠다.
2. 네트워크게임(서버)

* 메뉴에서 “네트워크 2/3인게임(게임생성)”를 클릭하면 “서버”, “네트워크게임-서버”를 실행한다. “네트워크게임-서버”이 “네트워크게임-유저”와 다른 점은 시작하기 버튼 유무이다.
* 서버는 TCP/IP 소켓통신을 통해 구현하였고 포트는 8544로 지정했다.
* 4개의 스레드를 이용했다. Timerthread를 이용해 서버에서 타이머를 재고, TimerSendThread에서 타이머가 0이 되었을 때 모든 클라이언트들에게 타이머를 초기화하라는 문자열을 전송한다. ConnectThread를 통해 접속하는 유저를 받고 유저정보를 List에 담아줬다. Userthread에서 핵심적인 기능을 수행한다. 유저들에게서 메시지를 받으면 그 유저를 제외한 유저들에게 메시지를 뿌려주도록 설정했다. 게임시작을 했을 때 총 유저의 숫자와 타이머 정보를 뿌려주고, 각 유저들의 이름을 받아와 공유해준다. 그 이외에 특정 명령어를 지정해줘서 각 클라이언트 끼리 동기화가 이루어 질 수 있도록 구현했다.

1. 네트워크게임(클라이언트)

* 메뉴를 선택하면 Client\_Wait(접속대기창)이 열린다. 접속버튼을 누르게 되면 입력한 아이피와 포트(8544)로 접속을 시도한다. 접속이 성공하면 게임 폼을 띄운다.
* 1인게임에서 설명한 기능과 모든 기능은 거의 동일하고 서버에서 받아온 메시지 처리하는 과정과 메시지를 보내는 과정 등이 추가되었다.
* 두가지 스레드가 더 추가되었는데 ClientReader 스레드에서 서버에서 메시지가 전송되면 그 메시지를 처리하는 부분을 담당한다. ConnectThread에서 0.5초마다 한번씩 반복을 실행하는데 Stonechangeinfor문자열에 유저가 서버에 보낼 명령이 입력이 되면 문자열을 전송하고 Stonechangeinfor에 빈값을 넣는다.

1. 소리 및 환경설정

* 소리는 Sound.java에서 메서드로 구현했다.
* 환경설정(Setup.java) 에서 플레이어이름, 타이머시간을 설정할 수 있도록 했다.
* 네트워크게임에서 직접 설정해준 player2, player3이름은 의미가 없다. 각 유저의 설정된 닉네임 값을 받아오기 때문이다. 그리고 타이머시간은 서버에서 설정한 시간을 따른다.

AI 구현 설명

* 가장 빨리 목적지에 도달할 방법은 최대한 많은 점프를 이용해서 목적지에 다가가는 방법이다. 따라서 자신이 가진 모든 말들이 어느 경로로 가야 최대한 얼마나 점프할 수 있는지 알아내고, 그 경로를 Vector<Vector<integer>>에 모두 저장한다 그러고나서, 그 경로들의 마지막이 되는 위치가 나은 선택이면 그 경로를 따라 이동하도록 했다.
* 만약 점프 할만한 괜찮은 경로가 없다면 적절한 위치로 한 두칸의 점프나 한칸 이동을 하도록 했다.

구현 못한 부분

* 일반게임에서 랜덤한 게임순서, 네트워크 게임에서 랜덤한 색깔, 위치, 순서를 넣지 못했다.
* AI에서 경로들 중에서 최선의 선택을 구하는 방법을 알 것 같은데 구현 못했다.