**네트워크 게임 프로그래밍**

**팀프로젝트**

윤승규

신동원

김석현

**목차**

[1. 게임 기획 3](file:///C:\Users\tjrgu\Desktop\학교\넷겜플\새%20폴더\네트워크%20게임%20프로그래밍%20팀프.docx#_Toc55084126)

[1.1 게임 개요 3](file:///C:\Users\tjrgu\Desktop\학교\넷겜플\새%20폴더\네트워크%20게임%20프로그래밍%20팀프.docx#_Toc55084127)

[1.2 조작 방법 3](file:///C:\Users\tjrgu\Desktop\학교\넷겜플\새%20폴더\네트워크%20게임%20프로그래밍%20팀프.docx#_Toc55084128)

[2. High Level Design 5](file:///C:\Users\tjrgu\Desktop\학교\넷겜플\새%20폴더\네트워크%20게임%20프로그래밍%20팀프.docx#_Toc55084131)

[2.1 클라이언트 5](file:///C:\Users\tjrgu\Desktop\학교\넷겜플\새%20폴더\네트워크%20게임%20프로그래밍%20팀프.docx#_Toc55084132)

[2.2 서버 6](file:///C:\Users\tjrgu\Desktop\학교\넷겜플\새%20폴더\네트워크%20게임%20프로그래밍%20팀프.docx#_Toc55084133)

[3. Low Level Design 7](file:///C:\Users\tjrgu\Desktop\학교\넷겜플\새%20폴더\네트워크%20게임%20프로그래밍%20팀프.docx#_Toc55084136)

[3.1 프로세스 7](file:///C:\Users\tjrgu\Desktop\학교\넷겜플\새%20폴더\네트워크%20게임%20프로그래밍%20팀프.docx#_Toc55084137)

[3.2 함수 8](file:///C:\Users\tjrgu\Desktop\학교\넷겜플\새%20폴더\네트워크%20게임%20프로그래밍%20팀프.docx#_Toc55084139)

[4. 팀원 역할 9](file:///C:\Users\tjrgu\Desktop\학교\넷겜플\새%20폴더\네트워크%20게임%20프로그래밍%20팀프.docx#_Toc55084143)

[5. 개발 일정 10](file:///C:\Users\tjrgu\Desktop\학교\넷겜플\새%20폴더\네트워크%20게임%20프로그래밍%20팀프.docx#_Toc55084145)

[5.1 윤승규 일정 10](file:///C:\Users\tjrgu\Desktop\학교\넷겜플\새%20폴더\네트워크%20게임%20프로그래밍%20팀프.docx#_Toc55084146)

[5.2 신동원 일정 10](file:///C:\Users\tjrgu\Desktop\학교\넷겜플\새%20폴더\네트워크%20게임%20프로그래밍%20팀프.docx#_Toc55084147)

[5.3 김석현 일정 10](file:///C:\Users\tjrgu\Desktop\학교\넷겜플\새%20폴더\네트워크%20게임%20프로그래밍%20팀프.docx#_Toc55084148)

1. **게임 기획**
   1. **게임 개요**



게임: 봄버맨 모작게임

플랫폼: PC

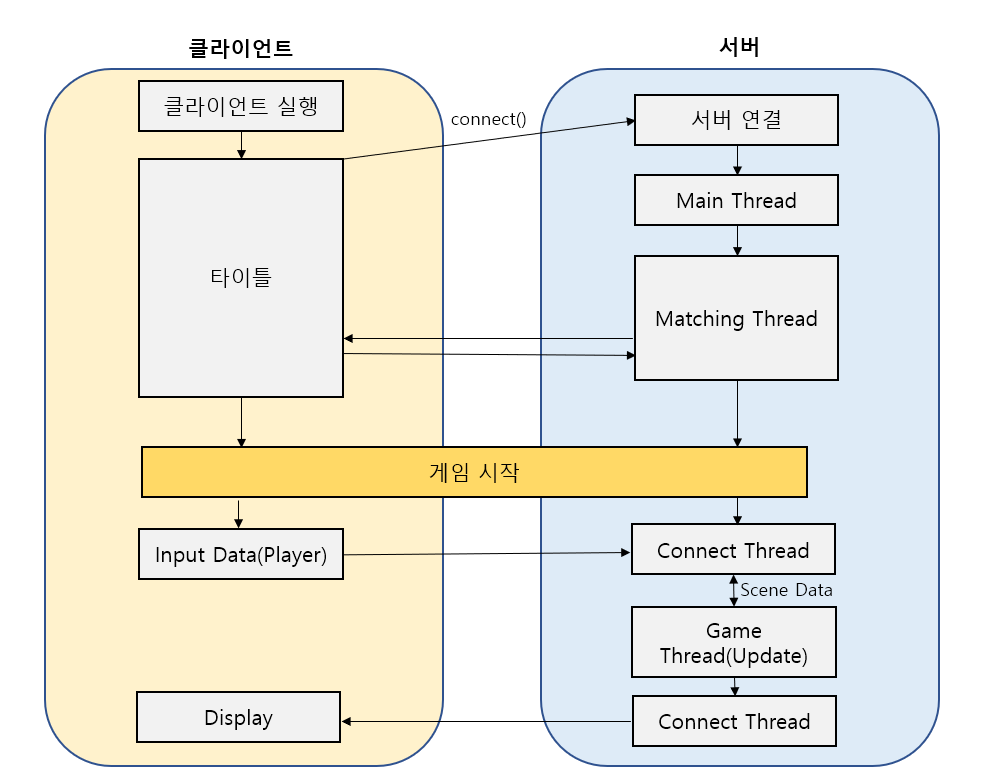
규칙: 2명의 플레이어가 서로 적대하여, 폭탄을 이용해 공격한다. 마지막까지 살아남은 플레이어가 승리한다.

* 1. **구현 내용**
* **클라이언트**
  1. **조작 방법**

|  |  |
| --- | --- |
| 조작키 | 설명 |
| W, A, S, D | 플레이어의 이동 방향키 W-위/A-왼/S-뒤/D-오 |
| Space bar | 플레이어의 위치에 폭탄을 설치한다. |

1. **High level design**

TCP/IP 프로토콜 방식

****

1. **Low level design**

DWORD WINAPI MatchingThread()

vector<SOCKET> MatchingQueue

unsigned char ClientNum

typedef struct ClientState

{

SOCKET ClientSocket

bool ReadyState = 0

}

typedef struct ClientInfo

{

bool ReadyState

}

void CreatGameThread

.

DWORD WINAPI GameThread(LPVOID arg

DWORD WINAPI ConnectThread(LPVOID arg.

class SceneData

{

Private:

PlayerStatus m\_playerStatus[MAX\_PLAYER];

MapData m\_mapData[MAP\_SIZE][MAP\_SIZE];

Public:

update();

keyInput(SOCKET socket, KeyInput key)

}

struct PlayerStatus

{

SOCKET playerSocket;

char speed;

char power;

bool isAlive;

};

struct MapData

{

Bool isRock

Bool isBubble;

};

* 1. **함수**

**클라이언트**

|  |  |  |
| --- | --- | --- |
| 구분 | 함수 | 설명 |
| 송신 | Recv\_Con\_dat() | 서버 로부터 데이터 소켓을 전달받는 함수 |
| Recv\_draw\_data() | 서버에서 받은 Update 데이터를 그리는 함수 |
| Recv\_check() | Player가 모두 준비되었는지 데이터를 받는 함수 |
| 수신 | Send\_Con\_dat() | 서버에 데이터 소켓을 전달하는 함수 |
| Send\_input\_data() | 입력 받은 데이터를 서버에 전달하는 함수 |
| 처리 | Change\_status() | 게임의 상태를 play, wait, end 로 바꿔주는 함수 |
| Update() | 입력 받은 데이터를 확인해 최신화 하는 함수 |
| Check\_state() | 게임의 진행 상황을 서버가 확인하도록 요청하는 함수 |
| Timeout() | 일정시간 안에 입력 데이터의 전송이 되지 않고 지연 시 False로 처리하는 함수 |

서버

|  |  |  |
| --- | --- | --- |
| 구분 | 함수 | 설명 |
| 송신 | Recv\_Con\_dat() | 클라이언트 로부터 데이터 소켓을 전달받는 함수 |
| Recv\_input\_data() | 클라이언트 로부터 입력 데이터를 받는 함수 |
| Recv\_check\_state() | 클라이언트로부터 실시간으로 게임 상태를 확인하도록 요청 받아 확인하는 함수 |
| 수신 | Send\_Con\_dat() | 클라이언트 에 데이터 소켓을 전달하는 함수 |
| Send\_draw\_data() | 서버에서 Update 처리한 데이터를 클라이언트에 보내는 함수 |
| send\_check() | 2명의 player의 상태를 보내는 함수 |
| 처리 | Change\_status() | 게임의 상태를 play, wait, end 로 바꿔주는 함수 |
| Check\_player() | 2명의 player가 서버와 연결되어 준비상태인지 확인하는 함수 |
| Update() | 입력 받은 데이터를 확인해 최신화 하는 함수 |
| Check\_state() | 게임의 진행 상황을 서버가 확인하도록 요청하는 함수 |
| Timeout() | 일정시간 안에 입력 데이터의 전송이 되지 않고 지연 시 False로 처리하는 함수 |

1. **팀원 역할**

|  |  |
| --- | --- |
| 이름 | 역할 |
| 윤승규 | 서버 송/수신 함수 및 클라이언트 송신 함수 구현 |
| 신동원 | 서버 프레임 워크 및 송신 함수 구현 |
| 김석현 | 클라이언트 프레임워크 및 수신함수 구현, 문서 작성 |

1. **개발 일정**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **11월** | | | | | | | |
| **구분** | **일** | **월** | **화** | **수** | **목** | **금** | **토** |
| **6** | **7** | **8** | **9** | **10** | **11** | **12** |
| **신동원** | **계획서 작성** | | | |  | **서버 프레임 워크 구현** | |
| **김석현** |  | **클라이언트 프레임 워크** | |
| **윤승규** | **함수1** | | **함수1보완** |
|  | **13** |  |  |  |  |  |  |
| **신동원** | **정기 회의** | **status (),Timeout() 구현** | | | **Recv() 구현** | | |
| **김석현** | **클라이언트 프레임워크 (2)** | | **함수 1** | | | **함수 2** |
| **윤승규** | **함수2** | | | **함수2보완** |
|  | **20** |  |  |  |  |  |  |
| **신동원** | **정기 회의** | **Send() 구현** | | | | | **병합** |
| **김석현** | **접속 테스트** | | **문제점 보완(1)** | | |
| **윤승규** | **함수1,2 테스트 보완** | | |
|  | **27** |  |  |  |  |  |  |
| **신동원** | **정기 회의** | **Update() 구현** | | | | | |
| **김석현** | **함수 3 / 데이터 송수신 보조** | | | | | |
| **윤승규** | **함수3** | | | | | |
|  |  |  |  |  |  |  |  |
| **신동원** | **정기 회의** |  |  | | | | |
| **김석현** |  |
| **윤승규** |  |