

표지

1. 제목 (게임 이름)
2. 팀원

목차

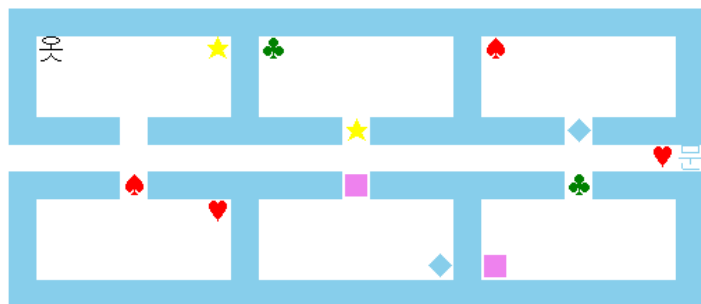
- 개요
- UI설계
- 프로그램 구조
- 알고리즘
- Sequence Diagram
- 소스 코드
- 사용성 평가 결과
- 일정표
- 마치며
- 참고자료

개요

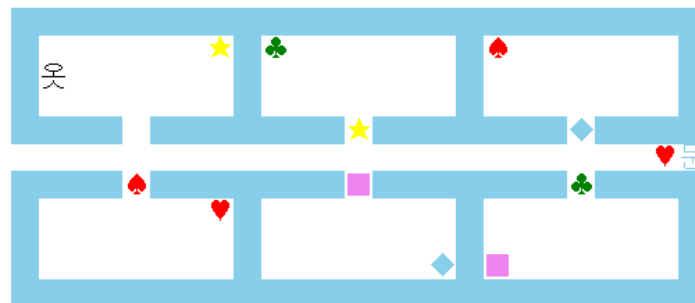
프로그램 소개
(동기, 목표, 특징 등)

UI 설계

설계도, 캡처화면, 동영상
모두 가능

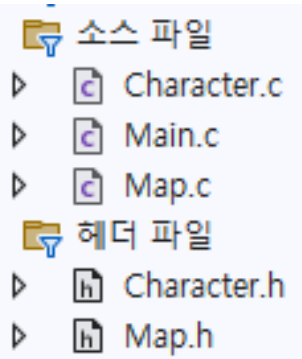


WASD와 [Enter]를 입력하세요: **s**



WASD와 [Enter]를 입력하세요: **q**

프로그램 구조



Main.c

```
int main()
```

Character.h

```
Character.c
```

```
void move( char *map, int row, int column );  
int finish( int row, int column );
```

소스 파일 목록
주요 함수 목록
주요 변수 목록

Map.h

```
Map.c
```

```
extern int map[100][100];  
extern int width, height;  
int readMap( char *filename );  
void writeMap();
```

알고리즘

Main.c

프로그램 시작
먼저 파일에서 맵
이전 위치의 캐시
게임 종료 확인
게임 종료 아니면
프로그램 종료

주요함수별 알고리즘

Character.h

Character.c

맵 이동 함수
W는 위로 한칸 이동
A는 왼쪽으로 한칸 이동
S는 아래로 한칸 이동
D는 오른쪽으로 한칸 이동

게임 종료 확인 함수
목적지에 도착하면 종료
아니면 게임 계속 진행

Map.h

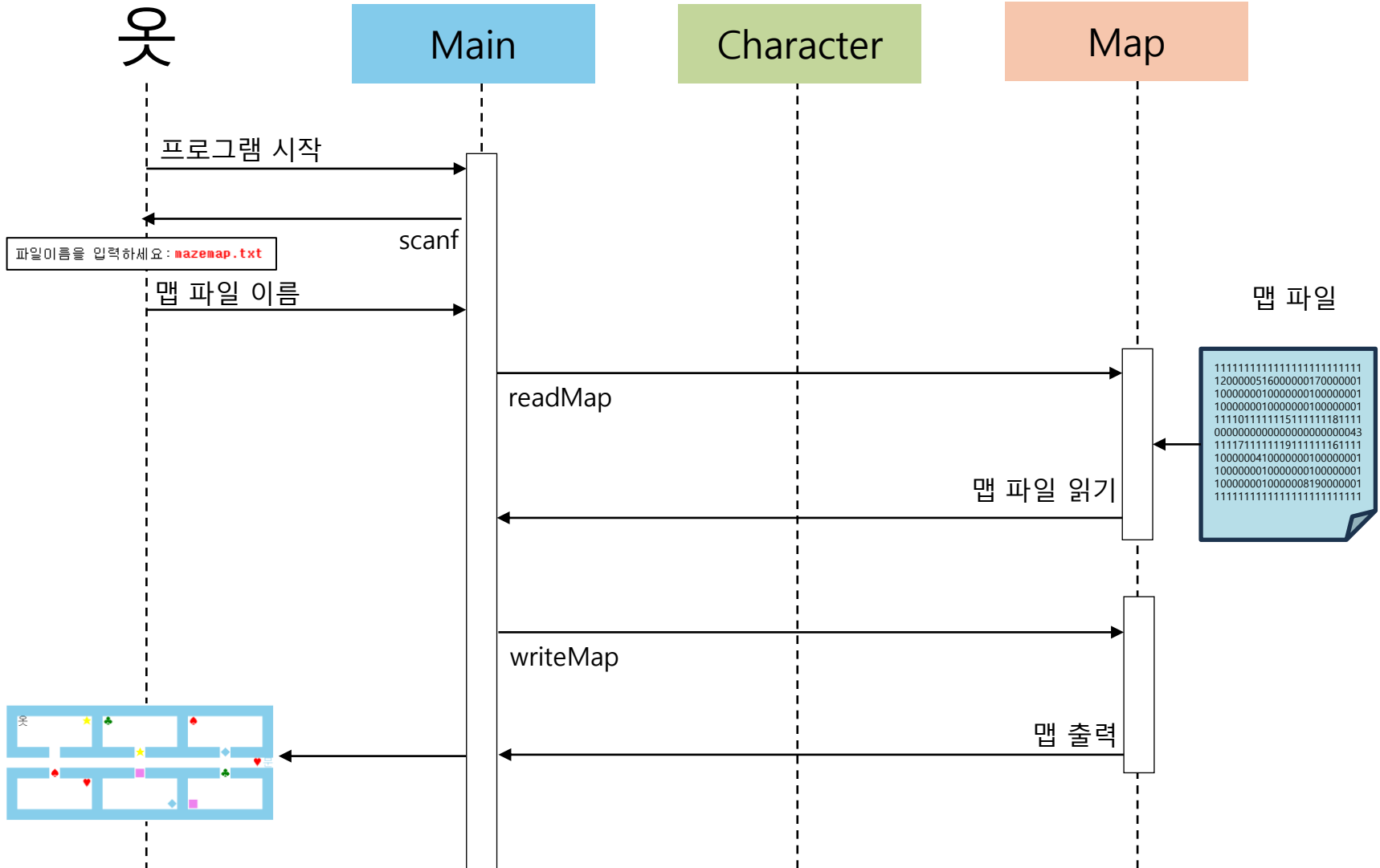
Map.c

맵 읽기 함수
맵 파일 열기
맵 파일에서 한 줄씩 읽기
맵 파일 닫기

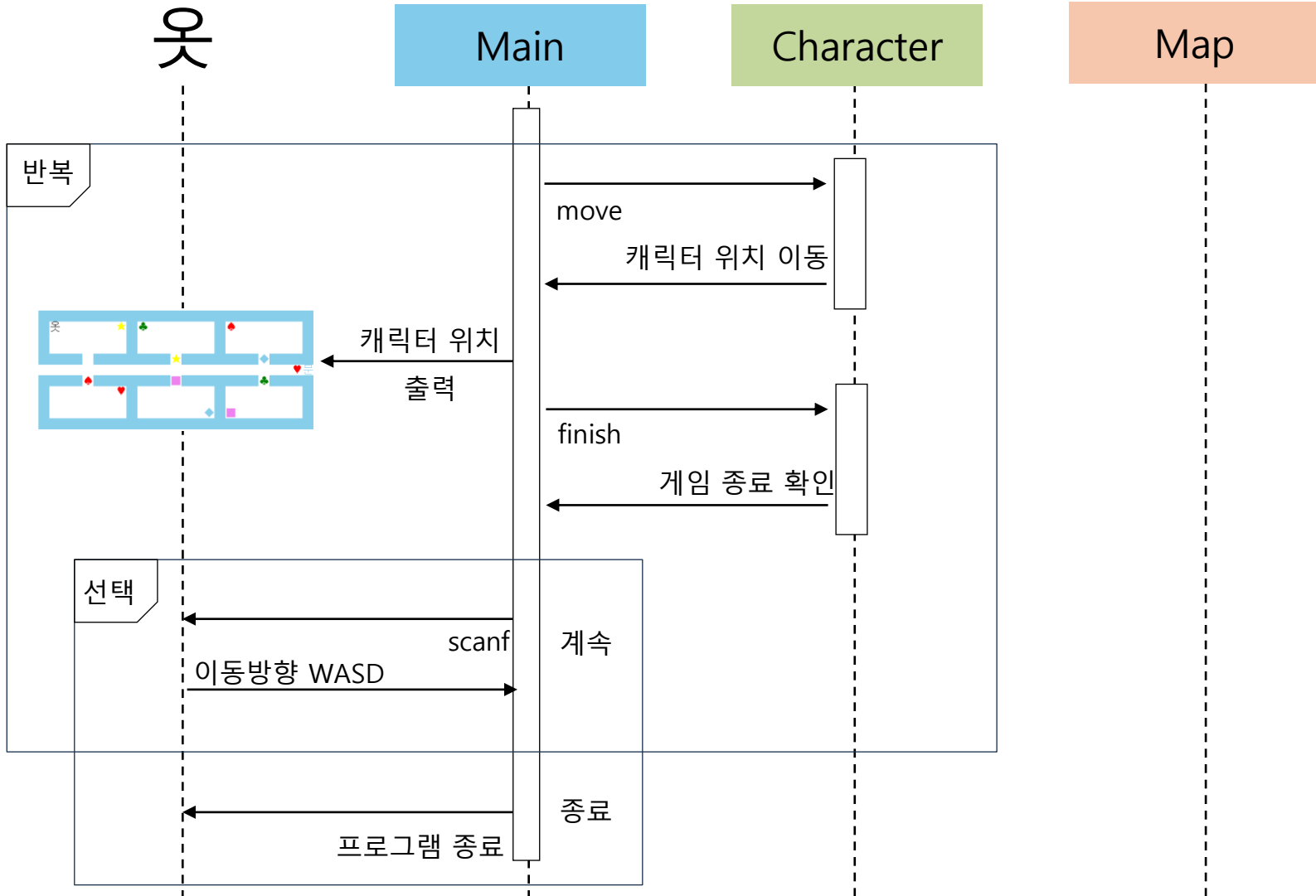
맵 쓰기 함수

Sequence Diagram

주요 함수의
호출 관계를
표현



Sequence Diagram



소스 코드

■ Main.c

```
#define _CRT_SECURE_NO_WARNINGS
#include <stdio.h>
#include "Character.h"
#include "Map.h"

// 프로그램시작
int main() {
    // 먼저 파일에서 맵을 읽고 출력
    printf( "파일이름을 입력하세요: " );
    char filename[1024] = "";
    scanf( "%s", filename );
    if ( readMap( filename ) == FALSE )
        return FALSE;
    writeMap();
}
```

소스 코드

■ Main.c

```
char direction = 'w';
int row = 1, column = 1;
do {
    // 이전 위치의 캐릭터를 지우고 이동하려는 새 위치에 캐릭터를 출력
    printf( "W033[%d;%df  ", row+1, column*2+1 );
    move( direction, &row, &column );
    printf( "W033[%d;%df웃", row+1, column*2+1 );
    // 게임 종료 확인
    if( finish( row, column ) == TRUE )
        break;
    // 게임 종료 아니면 새 이동 방향 입력 받아서 게임 계속 진행
    printf( "W033[%d;1fW033[2KWASD와[Enter]를 입력하세요: ", height + 2 );
    fflush( stdin );
    scanf( "Wn%c", &direction );
} while ((direction=='W')||(direction=='A')||(direction=='S')||(direction=='D')
        ||(direction=='w')||(direction=='a')||(direction=='s')||(direction=='d'));
// 프로그램 종료
return 0;
}
```

소스 코드

■ Map.h

```
#ifndef _MAP_H_
#define _MAP_H_

#define FALSE 0
#define TRUE 1

extern int map[100][100];
extern int width, height;

int readMap( char *filename );
void writeMap();

#endif
```

소스 코드

■ Map.c

```
#define _CRT_SECURE_NO_WARNINGS
#include <stdio.h>
#include <string.h>
#include "Map.h"

int map[100][100];
int width, height;

// 맵 쓰기 함수
void writeMap() {
    char symbol[10][1024] = { " ", "W033[44m W033[0m", "웃", "W033[34m문W033[0m", "W033[31m♥W033[0m", "W033[33m★W033[0m", "W033[32m♣W033[0m", "W033[31m♠W033[0m", "W033[36m♦W033[0m", "W033[35m■W033[0m" };
    printf("W033[2KW033[1;1f");
    for ( int row = 0; row < height; row++ ) {
        for ( int column = 0; column < width; column++ )
            printf("%s", symbol[map[row][column]]);
        printf("Wn");
    }
}
```

소스 코드

■ Map.c

```
// 맵 읽기 함수
int readMap( char *filename ) {
    // 맵 파일 열기
    FILE* inFile = NULL;
    if ((inFile = fopen(filename, "rt")) == NULL) {
        printf("[오류] 파일을 열 수 없습니다!");
        return FALSE;
    }
    // 맵 파일에서 한 줄씩 읽기
    for (char line[1024] = ""; fgets(line, 100, inFile) != NULL; height++) {
        if (strlen(line) < 3)
            break;
        else {
            for (width = 0; (width < strlen(line)) && ('0' <= line[width]) && (line[width] <= '9'); width++)
                map[height][width] = line[width] - '0';
        }
    }
    // 맵 파일 닫기
    fclose(inFile);
    return TRUE;
}
```

소스 코드

■ Character.h

```
#ifndef _CHARACTER_H_
#define _CHARACTER_H_

void move( char direction, int *row, int *column );
int finish( int row, int column );

#endif
```

소스 코드

■ Character.c

```
#include <stdio.h>
#include "Character.h"
#include "Map.h"

#define WALL 1
#define EXIT 3

// 게임 종료 확인 함수
int finish(int row, int column) {
    // 목적지에 도착하면 종료
    if ( map[row][column] == EXIT ) {
        printf("W033[2JW033[4;15fSUCCESS!W033[7;15f");
        return TRUE;
    }
    // 아니면 게임 계속 진행
    else
        return FALSE;
}
```

소스 코드

■ Character.c

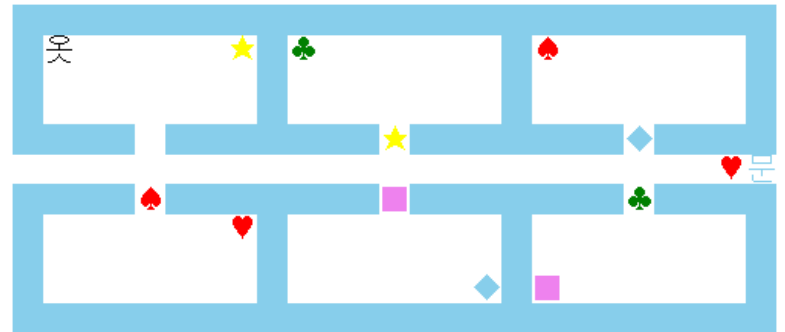
```
// 맵 이동 함수
void move( char direction, int *row, int *column ) {
    // W는 위로 한칸 이동
    if (((direction=='w')||(direction=='W'))&&(*row>0)&&(map[*row-1][*column]!=WALL))
        *row = *row-1;
    // A는 왼쪽으로 한칸 이동
    else if (((direction=='a')||(direction=='A'))&&(*column>0)&&(map[*row][*column-1]!=WALL))
        *column = *column-1;
    // S는 아래로 한칸 이동
    else if (((direction=='s')||(direction=='S'))&&(*row<(height-1))&&(map[*row+1][*column]!=WALL))
        *row = *row + 1;
    // D는 오른쪽으로 한칸 이동
    else if (((direction=='d')||(direction=='D'))&&(*column<(width-1))&&(map[*row][*column+1]!=WALL))
        *column = *column + 1;
}
```


소스 코드

■ mazemap.txt

11111111111111111111111111111111
1200000051600000001700000001
1000000001000000001000000001
1000000001000000001000000001
11110111111115111111181111
0000000000000000000000000043
1111711111119111111161111
1000000041000000001000000001
1000000001000000001000000001
1000000001000000081900000001
1111111111111111111111111111

```
char symbol[10][1024] = {
    "",
    "₩033[44m ₩033[0m",
    "옷",
    "₩033[34m문₩033[0m",
    "₩033[31m♥₩033[0m",
    "₩033[33m★₩033[0m",
    "₩033[32m♣₩033[0m",
    "₩033[31m♠₩033[0m",
    "₩033[36m♦₩033[0m",
    "₩033[35m■₩033[0m"
};
```

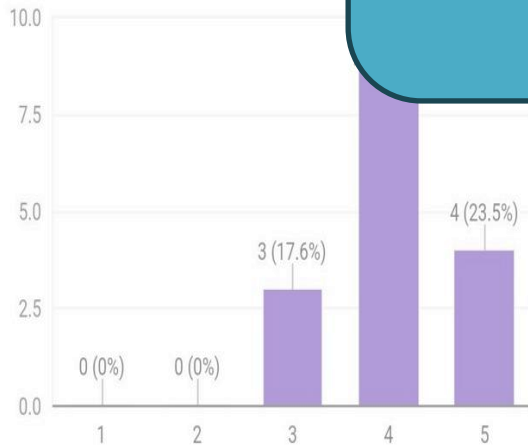


사용성 평가 결과

객관식 설문결과

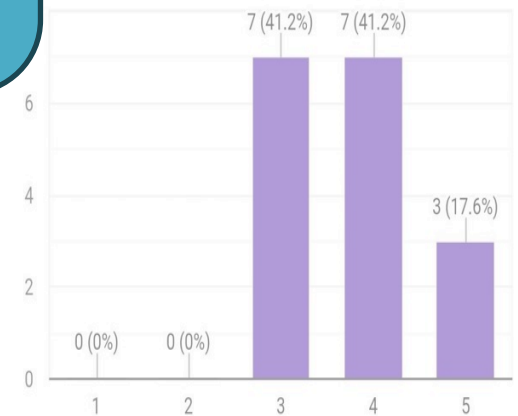
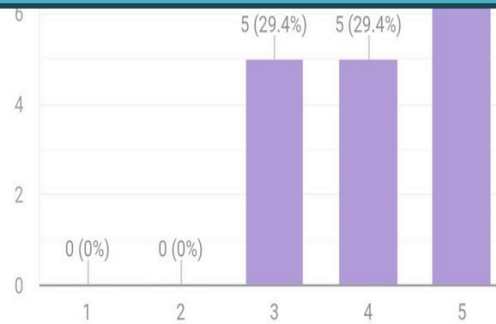
참신한가?

응답 17개



레이하고 싶은가?

7개



사용성 평가 결과

마음에
들었던 점

주관식 설문결과
(마음에 들었던 점,
개선되었으면 하는 점)

개선되었으면
하는 점

일정표

이름		웹서버구축		프로토타입			웹사이트 개발						평가결과분석 및 개선		
		3/11	3/18	3/25	4/1	4/8	4/15	4/22	4/29	5/6	5/13	5/20	5/27	6/3	6/10
기획 및 평가 (김OO)	웹서버구축														
	UI 벤치마킹-SWOT 분석														
	프로젝트 진행 계획														
	요구사항분석(UML)														
	태스크 시나리오														
	사용성 평가 문항														
	로그데이터 종류 및 추출방법														
	평가결과 분석 및 개선														
데이터베이스 (이OO)	웹서버구축														
	DB 요구사항 분석														
	개념적설계(ER-다이아그램)														
	논리적설계(테이블)														
	물리적설계(SOL)														
	데이터 구축														
	평가결과 분석 및 개선														
백엔드개발 (박OO)	웹서버구축														
	시스템 설계(UML 다이어그램-클래스, 시퀀스)														
	검색/삽입/삭제(JSP)														
	검색/삽입/삭제(PHP)														
	회원가입/로그인														
	대여/반납														
	로그데이터 추출														
	평가결과 분석 및 개선														
프론트엔드개발 (홍OO)	웹서버구축														
	UI설계-스케치, XD, 피그마														
	검색/삽입/삭제(HTML, CSS)														
	회원가입/로그인/대여/반납(HTML, CSS)														
	JavaScript 구현														
	지도 API 연동														
	평가결과 분석 및 개선														

팀 전체의 목표 제시

개인별& 작업별로
자세히 기술

1주일 단위로 계획 &
진행상황 표기

매주 개인별 작업내용
표기

마치며

	어려웠던 점	극복한 방법
1.1	<p>기획하면서 나름 제작이 가능한 정도의 부랴만 기획했다고 생각했는데 생</p> <p>또, 개발을 다같이 다른 사람들의</p>	<p>파일 저장해두고 새 파일로 열어서 음거하고 안되면 다시 고치면서 부으며 극복했습니다.</p>
1.2	<p>정말 0부터 10 프로</p> <p>답이 없는 문제</p> <p>팀원들이 각자 코드를 합치면 원인</p> <p>부분도 어려웠습니다.</p>	<p>...</p>
1.3	<p>처음 해보는 게임 제작에 코딩을 하는 시간이 오래 걸리고 오류 및 버그를 찾아내는 데 힘들었습니다.</p>	<p>게임을 코딩 할때 게임의 진행 방향과 같은 방향으로 코딩을 하도록 노력해야겠습니다.</p>
1.4	<p>한명이 코드를 몰아서 하는 것이 아닌 여러 명이 코드를 만들어서 합치는 구조다 보니, 그 과정에서 변수가 겹치거나, 혹은 함수의 타입이 안 맞는 등의 오류와 버그가 발생하여 그 점을 해결하는 점이 쉽지는 않았던 것 같습니다.</p> <p>또한 코드가 길어지다 보니, 오류나 버그를 찾는 것이 쉽게 이루어지지도 않았고 버그를 찾아서 해결하면 다른 버그가 보이기도 하였습니다.</p>	<p>초반에는 회의를 통해서 자주 만나서 대화를 했고 후반부에는 함수나 변수 근처에 주석을 달아서 코드를 제공함으로써 문제를 해결했습니다.</p> <p>코드가 길어져도 메인함수를 시작으로 모든 함수는 함수의 호출과 리턴값으로 연결되어 있으므로 함수의 위치를 조정하거나, 호출할 함수의 위치를 대략적으로 파악하여 문제를 해결 할 수 있었으며, 버그의 경우 함수들의 작동 메커니즘을 파악하여 문제점을 알 수만 있다면 버그를 해결할 수 있었습니다.</p>

팀원별 의견 (어려웠던 점, 극복한 방법)

참고자료

소스코드 작성시
참고한 내용 및
출처(URL 등)