12주차 예비보고서

전공: 기계공학과 학년: 3학년 학번: 20191820 이름: 김형준

**1.**

**사용되는 매크로, 전역 변수 및 구조체 :**

#define baseX 15.0

#define baseY 15.0

#define Length 30.0

Room\*\* Maze = NULL

int Row, Col, Init = 0

typedef struct \_room {

int visited

int r

int d

}Room

**bool readFile(const char\* filename)**

string fileName = filename

ofFile file(fileName)

if (!file.exists())

cout << "Target file does not exists." << endl

return false

else

cout << "We found the target file." << endl

isOpen = 1

ofBuffer buffer(file)

if (Init == 1)

freeMemory()

Init = 1

string templine = buffer.getFirstLine()

Col = (templine.size() - 1) / 2, Row = 0

while(1)

templine = buffer.getNextLine()

if (templine.size() < Col) break

Row++

Row /= 2

Maze = (Room\*\*)malloc(sizeof(Room\*) \* Row)

for i = 0 to Row-1

Maze[i] = (Room\*)malloc(sizeof(Room) \* Col

for i = 0 to Row-1

for j = 0 to Col-1

Maze[i][j].visited = 0

Maze[i][j].r = 1

Maze[i][j].d = 1

templine = buffer.getFirstLine()

for i = 0 to Row-1

string rline = buffer.getNextLine()

string dline = buffer.getNextLine()

for j = 0 to Col-1

if (j < Col - 1)

if (rline[2 \* j + 2] == ' ')

Maze[i][j].r = 0

if (i < Row - 1)

if (dline[2 \* j + 1] == ' ')

Maze[i][j].d = 0

**void freeMemory()**

for i = 0 to Row-1

free(Maze[i])

free(Maze)

**void draw ()**

ofSetColor(100)

ofSetLineWidth(5)

ofDrawLine(baseX, baseY, baseX + Length \* (Col), baseY)

ofDrawLine(baseX, baseY, baseX, baseY + Length \* (Row))

for i = 0 to Row-1

for j = 0 to Col-1

if (Maze[i][j].r == 1)

ofDrawLine(baseX + (j+1) \* Length, baseY + i \* Length, baseX + (j+1) \* Length, baseY + (i+1) \* Length)

if (Maze[i][j].d == 1)

ofDrawLine(baseX + j \* Length, baseY + (i+1) \* Length, baseX + (j+1) \* Length, baseY + (i+1) \* Length)

**2.**

HWND hWnd = WindowFromDC(wglGetCurrentDC());

: 메뉴 및 버튼을 윈도우 창에 추가시키기 위해 HWND 타입(윈도우 창 핸들러) 변수를 선언하고, WindowFromDC(wglGetCurrentDC())의 리턴값을 변수에 저장함으로써 현재 윈도우 창의 정보를 핸들러 변수에 넘겨준다.

ofxWinMenu\* menu = new ofxWinMenu(this, hWnd);

: 메뉴를 추가하기 위해 위에서 만들었던 핸들러를 사용하여 새로운 메뉴 오브젝트를 만들어 ofxWinMenu 타입 포인터 menu에 저장한다.

menu->CreateMenuFunction(&ofApp::appMenuFunction);

: 메뉴의 버튼을 클릭했을 때 호출할 함수를 CreateMenuFunction()을 사용해 메뉴 오브젝트에 등록한다. (위 코드에서는 메뉴의 버튼을 클릭할 경우 ofApp::appMenuFunction() 함수를 호출함)

HMENU hMenu = menu->CreateWindowMenu();

: HMENU 타입 (메뉴 핸들러) 변수를 선언하고, menu->CreateWindowMenu();으로 윈도우 창에 메뉴를 추가하고 함수의 리턴값을 저장해 핸들러 변수에 생성된 메뉴의 정보를 넘겨준다.

HMENU hPopup = menu->AddPopupMenu(hMenu, string);

: menu->AddPopupMenu(hMenu, string);으로 hMenu가 가리키는 메뉴에 버튼의 문자열이 string인 팝업 메뉴를 생성한다. (팝업 메뉴는 클릭시 해당 메뉴가 가진 모든 아이템들을 버튼 위에 직사각형 박스를 띄워서 그 안에 모두 표시함)

menu->AddPopupItem(hPopup, string, bChecked,bAutoCheck);

: 버튼의 문자열이 string인 팝업 아이템을 팝업 메뉴(hPopup이 가리키는 팝업 메뉴)에 추가한다. bChecked는 프로그램 시작시 해당 아이템의 체크 여부를 설정하는 값이고 (true면 체크, false면 체크 해제상태로 시작함, 기본값 false), bAutoCheck는 클릭마다 아이템의 체크 여부를 전환할지를 설정하는 값이다. (true면 체크할때마다 해당 아이템의 체크값이 전환됨, false면 전환안함, 기본값 true)

아이템의 위치는 해당 팝업 메뉴에 추가하는 순서대로 위에서 아래로 내려가며, 아이템 사이에 menu->AddPopupSeparator(hPopup); (hPopup이 가리키는 팝업 메뉴에 분리선을 추가하는 함수)를 사용해 분리선을 삽입할 수 있다.