

REPORT

광운대학교
KwangWoon University



담당교수: 신영주 교수님

학과: 컴퓨터정보공학과

학번: 2017202037

이름: 오 민 혁

Introduction

이번 프로젝트는 MFC를 이용하여 2048 게임을 구현하는 것이다.

MFC란 Microsoft Foundation Class Library의 약자로 C++용 프로그램 라이브러리이다. MFC는 윈도우의 Object를 포함하는 시각적 객체를 사용하며, 사용자 또는 시스템이 윈도우에 어떠한 동작을 했을 경우 그에 대응하여 움직이는 사건의 중심의 프로그램을 수행하는 이벤트 처리 방식을 이용한다. 그리고 윈도우 이벤트가 발생했을 경우 해당 이벤트를 메시지를 통해 알려주고 그 메시지를 받아서 그에 따른 출력을 수행하는 메시지 처리 방식을 이용한다.

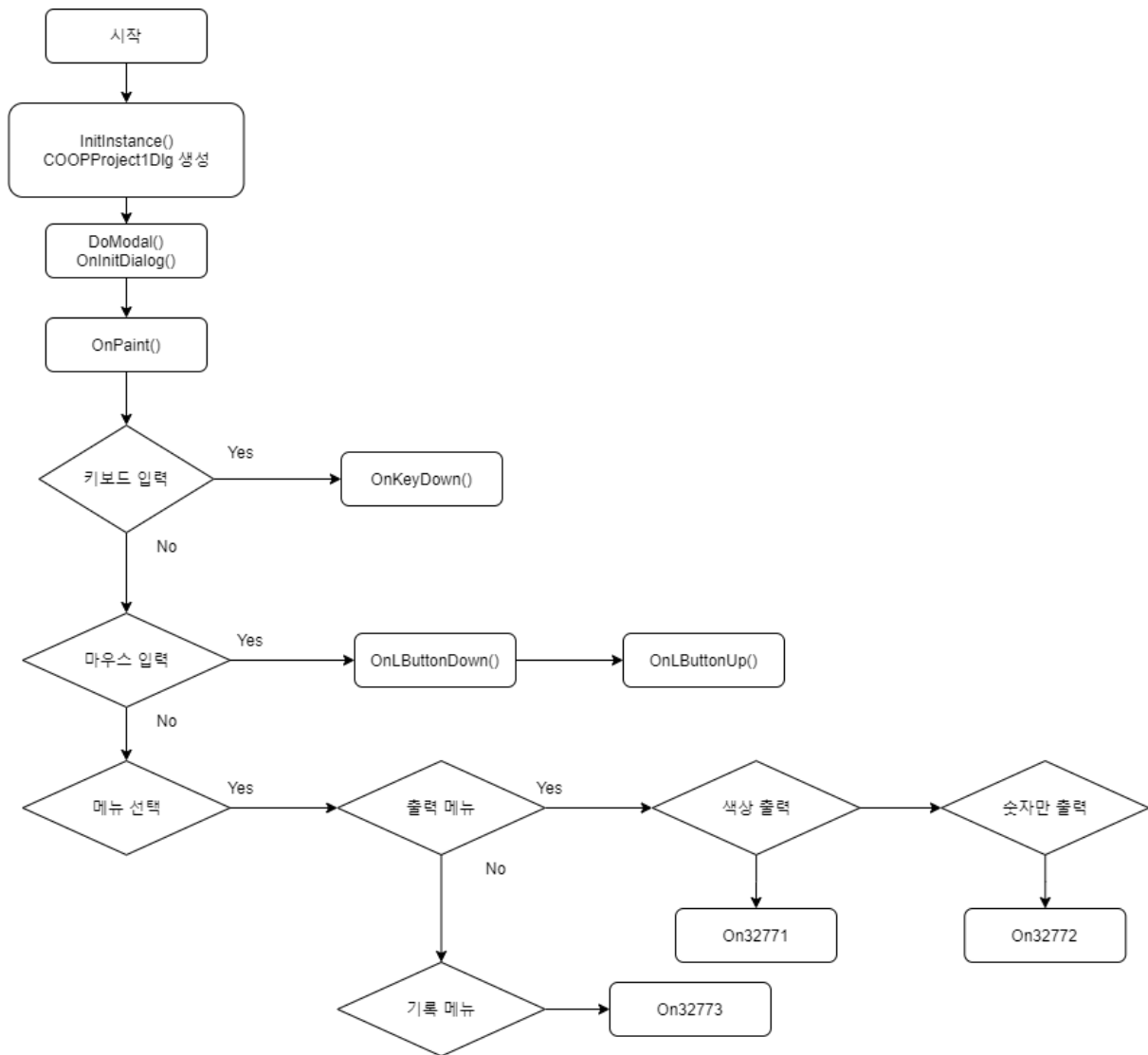
2048게임이란, 4 X 4 board에서 이루어지는 게임으로 방향키를 눌러 조작하여 2048의 숫자를 가진 Block이 만들어지면 게임을 승리하게 된다.

규칙은 다음과 같다.

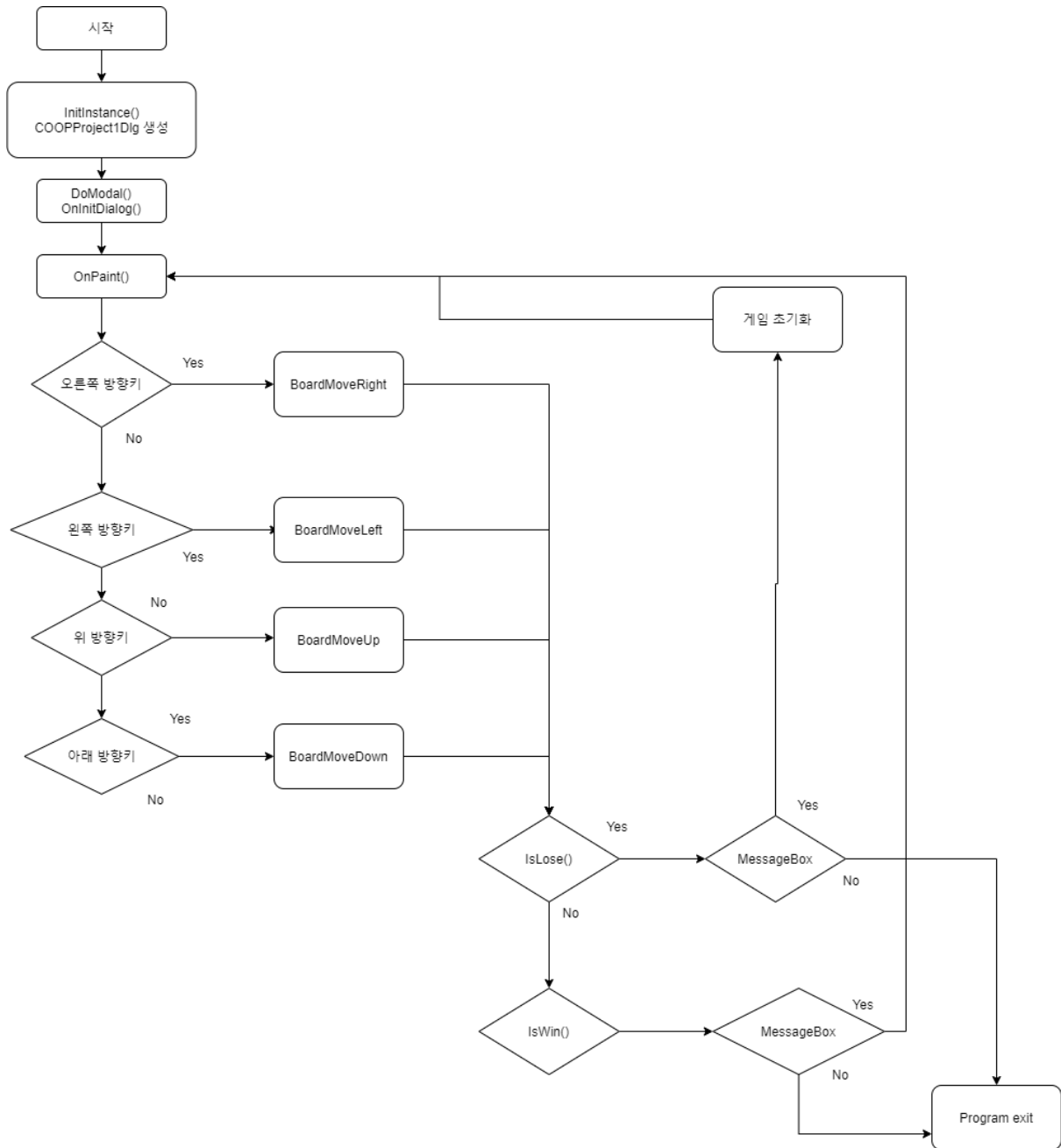
입력 받은 방향으로 Block들이 모두 비어 있는 블록 중 가장 가장자리로 움직이게 되며, 움직이는 방향으로 같은 숫자를 가진 Block이 있을 시에는 두 Block이 합쳐지며 Block이 가진 숫자는 2배가 된다. 이러한 규칙을 따를 시 Block이 가지는 숫자는 2의 제곱수 들로만 이루어진다.

우선 이 프로젝트를 하기 위해서는 Board Class와 Block Class를 정의해야 한다. Block Class는 위, 아래, 오른쪽, 왼쪽의 Member를 가지고 있으며, Board Class는 이러한 Block들이 2차원 4X4 형식으로 이중 Linked List가 되도록 연결한다. 2048 게임이 실행 되었을 때, 임의의 위치에 2를 가진 Block이 2개 생성된다. 그리고 2부터 2048 까지 2의 제곱수로 구성되는 Block의 숫자에 따른 각 Block 색상들이 따로 정해 져 있다. 그리고 방향키만 입력 받는 것이 아니라, 마우스에 대한 입력도 받는데, 만약 마우스를 이용하여 보드 내에서 왼쪽에서 오른쪽으로 드래그 할 시, Clock-wise 즉 시계방향으로 보드가 회전하며, 오른쪽에서 왼쪽으로 드래그 할 시 Counter-Clock-Wise 즉 반시계방향으로 보드가 회전하게 된다. 마지막으로 출력 메뉴와 기록 메뉴 2개의 메뉴가 있다. 출력 메뉴는 다시 하위 메뉴인 숫자/숫자, 색상 두 가지로 나뉘는데 이는 숫자만 출력하거나 숫자와 색상 모두 출력하는 기능을 수행한다. 그리고 기록 메뉴는 턴 기록, 즉 보드의 상태를 키보드나 마우스의 이벤트가 작동할 때 마다 Log.txt에 기록한다.

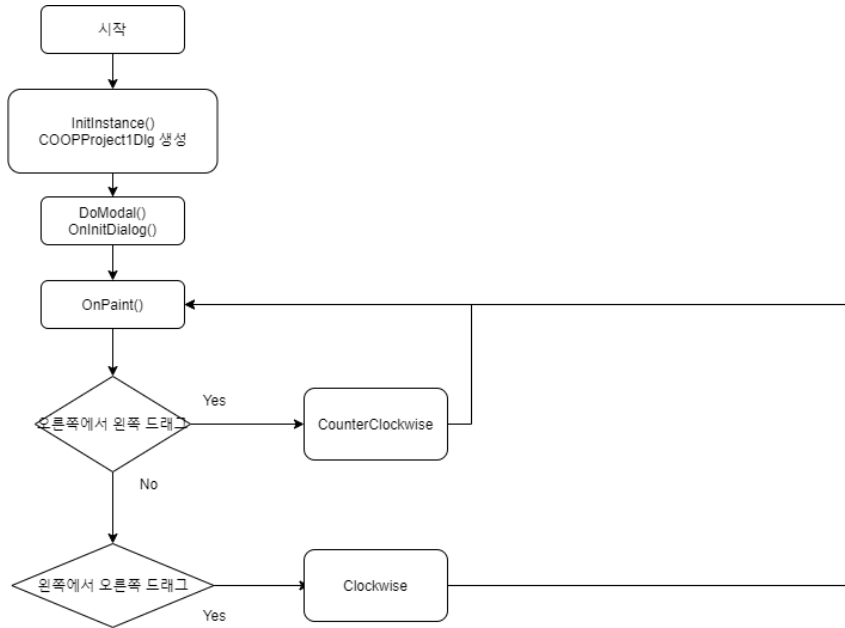
Flowchart



<키보드 입력 시>



<마우스 입력시>



Algorithm

<Board()>

I = 0 부터 16까지 반복{

새로운 Block 생성, Number = 0, $x=i\%4$, $y=i/4$, 이중 링크드 리스트 생성}

I = 0부터 4까지 반복

Block을 오른쪽으로 4번 이동

I=0부터 3까지 반복

J=0부터 4까지 반복

4X4 board를 만들기위해 4개씩 끊어서 위 아래로 연결

I=0부터 16까지 반복

가장자리에 있는 Block의 Right 모두 해제, 처음 자리에 있는

Block의 Left 모두 해제

두 개의 랜덤 Block 생성

<Move()>

각 4가지 방향에 대해서 비슷한 동작이기 때문에 Up만 설명

- Up Operation

모든 Block을 위쪽의 빈 Block중 가장자리로 이동

만약 같은 Number를 가진 Block이 위쪽에 있을 경우 두 Block
합침

다시 모든 Block을 위쪽의 빈 Block중 가장자리로 이동

<OnPaint()>

CPen, CBursh 및 font 설정 , 각 숫자에 맞는 색상을 설정하기 위해
CBrush 객체 생성, Block 크기 128로 설정, ColorSignal에 따라 색상
출력 및 숫자 Text 출력

<BlockSearch(x,y)>

x만큼 오른쪽 이동, y만큼 밑으로 이동 한 후 그 위치의 Block 반환

<On32771>

ColorSignal=1
OnPaint()
<On32772>
ColorSignal=0
OnPaint()

<On32773>
Flag=1
OnPaint()

<LocationSwapping(A,B)>
x=0, y=0
x=A->GetX(), y=A->GetY()
A->SetX(B->GetX()), B->SetY(B->GetY())
B->SetX(x), B->SetY(y)

<Clockwise()>
시작점을 잡고 바깥쪽부터 일일이 시계방향으로 Swapping해줌

<Counterclockwise()>
Clockwise()의 반대로 작동

<IsWin()>
Board의 모든 좌표를 탐색하여 2048을 가진 Block이 있을 시 True
반환

<IsLose()>
Board의 모든 좌표를 탐색하여 주변에 같은 Number를 가진 Block이
있을 시 false
아닐 시 true

<ResetGame()>
Board의 모든 Block의 Number 0으로 설정

결과화면

2048 Game				×
출력 메뉴 기록 메뉴				
2	16	2	8	
	2	16	16	
			4	
			2	

2048 Game				×
출력 메뉴 기록 메뉴				
	2		4	
8		8	16	
2	32	8	4	

2048 Game				×
출력 메뉴 기록 메뉴				
2	8			
32		2		
8	8			
4	16	4		

시계방향 회전

2048 Game				×
출력 메뉴 기록 메뉴				
	4	16	4	
		8	8	
	2		32	
		8	2	

반시계 방향 회전



실패 했을 때 메시지 박스 출력

고찰

이번 프로젝트를 진행하면서 MFC에 대한 많은 공부를 하였다. 사실 아직도 MFC의 동작 원리나 Dlg Class의 모든 함수들의 역할에 대해서는 잘 모르겠다. OnPaint() 그리고 메뉴나, 마우스 키보드 입력 시 이벤트 처리기에 대한 부분만 다루었던 것 같다. 사실 Board와 Block을 구현하는게 핵심이었고, Dialog 창을 디자인하는 부분은 부가적인 과제였던 것 같다. 맨 처음 OnKeyDown()에서 VK_UP을 설정해도 방향키가 입력이 안되어서 애를 먹었다. 열심히 해결법을 찾은 결과 Pretranslatemessage 함수를 추가해야 된 다는 것을 알았다. MFC는 이러한 사소한 오류들에 대해 처리하는 것이 너무 힘든 것 같다. 이번 프로젝트를 하면서 Linked List에 대해 다시 한 번 완벽하게 이해할 수 있었다.

