8주차 예비보고서

전공: 컴퓨터공학과 학년: 4학년 학번: 20212022 이름: 이예준

**1.**

텍스트, 도표, 평면도, 라인이(가) 표시된 사진

자동 생성된 설명

\*Flow\_Chart

**void InitTetris()**

: field를 0으로 초기화, nextBlock[0]과 nextBlock[1]을 랜덤으로 결정하고

block의 회전, 위치 변수를 초기화한다. 그리고 점수 및 시간을 초기화한다.

함수를 이용해서 outline, field, block, next block, score를 그린다.

**void DrawOutline()**

: block이 떨어지는 공간의 테두리, next block을 보여주는 공간의 테두리,

score를 보여주는 공간의 테두리를 그린다.

**int GetCommand()**

: switch문을 통해 방향키를 누르면 break되고, 스페이스바를 누르면 block을 떨어뜨린다.

q, Q를 누르면 종료되고, 그 외의 경우에는 아무 작업도 하지 않고, break된다.

**int ProcessCommand(int command)**

: command값을 이용해서 switch문을 실행하며, command값이 QUIT라면 ret값을

QUIT으로 설정하고, KEY\_UP이라면 block을 회전시키고, KEY\_DOWN이라면 block을 아래로

한 칸 내리고, KEY\_RIGHT이라면 block을 오른쪽으로 한 칸 이동하고, KEY\_LEFT이라면

block을 왼쪽으로 한 칸 이동하고, 그 외의 경우에는 break한다.

마지막으로 command값에 따라 block의 위치 및 모양이 변경되었다면,

DrawChange 함수를 통해 그림을 업데이트하여 그린다.

**void DrawField()**

: field를 그리는 함수이며, 2차원 배열 field값이 0이면 공백 “ “,

field 배열 값이 1이면 “.”로 출력한다.

**void PrintScore(int score)**

: 점수 판 위치에 score값을 출력한다.

**void DrawNextBlock(int \*nextBlock)**

: next block을 확인하는 위치에 이동하여 next block의 정보를 이용해서 그린다.

**void DrawBlock(int y, int x, int blockID,int blockRotate,char tile)**

: x, y를 이용해 block을 그려야 하는 위치를 지정해서 block의 모양을 나타내는 tile을 그린다.

blockRotate는 block의 회전 수를 나타낸다.

**void DrawBox(int y,int x, int height, int width)**

: Box를 그리는 함수로 x, y를 이용해서 이동을 하면서 그 위치에 Box를 그린다.

**void play()**

: 게임을 시작하기 위한 함수로 InitTetris 함수로 초기화하고

GetCommand,ProcessCommand 함수로 command에 대한 처리를 한다.

만약 Q를 누르면 “Good-bye!!” 메세지를 출력하고 종료한다.

**char menu()**

: 4가지의 각 명령어(1. play, 2. rank, 3. recommended play, 4. exit)를 선택할 수 있도록

출력이 되며, 번호를 입력하면 해당되는 명령어를 실행한다.

**2.**

**1. int CheckToMove()**

for(i= 0 ~ HEIGHT-1){

for(j= 0 ~ WIDTH-1){

if 회전 및 이동한 블록이 필드를 벗어나면 return 0 ;

else if 회전 및 이동한 블록 블록이 다른 블록과 충돌하면 return 0 ;

else return 1 ;

}

}

**2. void DrawChange()**

switch(command){

case KEY\_UP: 이동 전 block 정보 얻기 ;

case KEY\_DOWN: 이동 전 block 정보 얻기 ;

case KEY\_LEFT: 이동 전 block 정보 얻기 ;

case KEY\_RIGHT: 이동 전 block 정보 얻기 ;

}

for(i= 0 ~ HEIGHT-1){

for(j= 0 ~ WIDTH-1){

if 이동 전 block 위치이면 block 제거 ;

}

}

DrawBlock(현재 block 정보);

move(커서 이동)

**3. void BlockDown()**

If block y좌표가 -1이면Gameover = 1 ;

AddBlockToField() ;

DeleteLine() ;

nextBlock[0] = nextBlock[1] ;

nextBlock[1] = rand()%7 ;

blockRotate = 0 ;

blockY = -1 ;

blockX = WIDTH/2-2 ;

DrawNextBlock() ;

PrintScore() ;

Drawfield() ;

**4. void AddBlockToField()**

for(i= 0 ~ HEIGHT-1){

for(j= 0 ~ WIDTH-1){

if(Block[currentBlock][blockRotate][i][j] == 1) field[blockY+i][blockX+j] = 1 ;

}

}

**5. int DeleteLine()**

int flag = 0 ;

for(i= 0 ~ HEIGHT-1){

for(j= 0 ~ WIDTH-1){

if 한 줄의 element가 모두 1이면 해당 줄 삭제 & flag = 1 ;

}

if flag가 1이면 삭제한 바로 위에서부터 block을 한 줄씩 내림

}

return (지운 line개수)2 × 100