8주차 예비보고서

전공: 컴퓨터공학과 학년: 2학년 학번: 20211522 이름: 김정환

**1.**

.................

start

menu()

select mode

MENU\_EXIT

MENU\_PLAY

InitTetris()

exit = 1

GetCommand()

end

ProcessCommand(command)

!= QUIT

ProcessCommand

0

QUIT

ARROW\_KEY

check to move and move block as commanded

1

gameover

end

move block part

BlockDown()

CheckToMove()

0

yes

blockY == -1

gameOver = 1

no

blockY++

AddBlockToField()

DeleteLine()

nextBlock[0] = nextBlock[1]

nextBlock[1] = rand()%7

Initialize Location of currentBlock

end

void InitTetris() : 테트리스의 모든 global 변수를 초기화 한다.

void DrawOutline() : 테트리스의 모든 interface를 그려준다.

int GetCommand() : 테트리스 관련 키입력을 받는다.

int ProcessCommand(int command) : GetCommand()에서 받은 command에 대한 동작을 수행한다.

void BlockDown(int sig) : 블록이 1초마다 내려가도록 호출되는 함수(더 이상 내려가지 못할 때 처리할 부분도 수행)

int CheckToMove(char f[HEIGHT][WIDTH], int currentBlock, int blockRotate, int blockY, int blockX) : 입력된 움직임이 가능한지 판단한다.

void DrawChange(char f[HEIGHT][WIDTH], int command, int currentBlock, int blockRotate, int blockY, int blockX) : command에 의해 바뀐 부분만 다시 그린다.

void DrawField() : 테트리스 블록이 쌓이는 field를 그린다.

void AddBlockToField(char f[HEIGHT][WIDTH], int currentBlock, int blockRotate, int blockY, int blockX) : 떨어지는 블록을 field에 더해준다.

int DeleteLine(char f[HEIGHT][WIDTH]) : 완전히 채워진 Line을 지우고 점수를 return한다.

void gotoyx(int y, int x) : 커서의 위치를 입력된 x, y의 위치로 옮겨준다.

void DrawNextBlock(int\* nextBlock) : 화면 오른쪽 상단에 다음 블록을 그려준다.

void PrintScore(int score) : 화면 오른쪽 하단에 score를 출력한다.

void DrawBox(int y, int x, int height, int width) : (y, x)에 height \* width 크기의 box를 그린다.

void DrawBlock(int y, int x, int blockID, int blockRotate, char tile) : (y, x)에 원하는 모양의 블록을 그린다.

void DrawShadow(int y, int x, int blockID, int blockRotate) : 블록이 떨어질 위치를 미리 보여준다.

void play() : 게임을 시작한다.

char menu() : 메뉴를 보여준다.

void rank() : 화면에 랭킹 기록들을 보여준다.

void writeRankFile() : rank file을 생성한다.

void newRank(int score) : 새로운 랭킹을 추가한다.

int recommend(RecNode\* root) : 추천 블록 배치를 구한다.

void recommendedPlay() : 추천 기능에 따라 배치하는 게임을 시작한다.

................

**2.**

.......................

int CheckToMove(char f[HEIGHT][WIDTH], int currentBlock, int blockRotate, int blockY, int blockX)

for i = 0 to 3

for j = 0 to 3

if(block[currentBlock][blockRotate][i][j] == 1)

if(blockY + i >= HEIGHT or blockX + j >= WIDTH or blockX + j < 0)

return FALSE

if f[blockY + i][blockX + j] == 1

return FALSE

return TRUE

void DrawChange(char f[HEIGHT][WIDTH], int command, int currentBlock, int blockRotate, int blockY, int blockX)

int x = blockX, y = blockY, z = blockRotate

switch(command)

case KEY\_UP: z = (z + 3) % 4

case KEY\_DOWN: y -= 1

case KEY\_LEFT: x += 1

case KEY\_RIGHT: x -= 1

for i = 0 to 3

for j = 0 to 3

if( block[currentBlock][z][i][j] == 1)

print “ “ to field (y + i, x + j)

DrawBlock(blockY, blockX, currentBlock, blockRotate, ‘ ‘)

void AddBlockToField(char f[HEIGHT][WIDTH], int currentBlock, int blockRotate, int blockY, int blockX)

for i = 0 to 3

for j = 0 to 3

if(block[currentBlock][blockRotate][i][j] == 1)

f[blockY + i][blockX + j] = 1

int DeleteLine(char f[HEIGHT][WIDTH])

cnt = 0

for i = 0 to HEIGHT – 1

flag = TRUE

for j = 0 to WIDTH – 1

if f[i][j] == 0

flag = FALSE

break

if flag == TRUE

cnt += 1

for p = i to 1(downward)

for q = 0 to WIDTH – 1

f[p][q] = f[p – 1][q]

for q = 0 to WIDTH – 1

f[0][q] = 0

i -= 1

return cnt \* cnt \* 100

void BlockDown(int sig)

if CheckToMove(f, currentBlock, blockRotate, blockY + 1, blockX)

blockY += 1

DrawChange(f, command, currentBlock, blockRotate, blockY, blockX)

else

if blockY == -1

gameOver = TRUE

else

AddBlockToField(f, currentBlock, blockRotate, blockY, blockX)

score += DeleteLine(f)

nextBlock[0] = nextBlock[1]

nextBlock[1] = random integer(0 to 6)

blockY = -1

blockX = WIDTH/2 – 2

blockRotate = 0

DrawNextBlock(nextBlock)

PrintScore(score)

DrawField()

.........................