8주차 결과보고서

전공: 컴퓨터공학과 학년: 4학년 학번: 20181688 이름: 조태연

**1.**

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

for i from 0 to BLOCK\_HEIGHT-1

for j from 0 to BLOCK\_WIDTH-1

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

if (f[i + blockY][j + blockX] == 1) return 0;

if (i + blockY >= HEIGHT || i + blockY < 0 || j + blockX >= WIDTH || j + blockX <) return 0;

}

}

}

return 1;

}텍스트, 스크린샷이(가) 표시된 사진

자동 생성된 설명

상수인 BLOCK\_HEIGHT와 BLOCK\_WIDTH만큼 반복하기 때문에 시간복잡도는 O(1)이고 공간복잡도는 추가로 입력이 없기 때문에 마찬가지로 O(1)이다. 달라진 점은 없다.

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

x=blockX y=blockY

switch(command)

case KEY\_RIGHT

x-=1

case KEY\_LEFT

x+=1

case KEY\_UP

blockrotate=(blockrotate+1)%4

case KEY\_DOWN

y-=1

for i from 0 to BLOCK\_HEIGHT-1

for j from 0 to BLOCK\_WIDTH-1

if(block[currentblock][blockRotate][i][j]==1 && i+y>=0)

move(i+y+1,j+x+1);

print(“.”)

DrawBlock();

Move();

}

텍스트, 스크린샷, 소프트웨어, 멀티미디어 소프트웨어이(가) 표시된 사진

자동 생성된 설명

Switch와 for문 모두 상수만큼의 시간복잡도를 가지고 DrawBlock()과 move()또한 상수만큼의 시간복잡도를 가지기 때문에 전체시간복잡도는 O(1)이다. 공간복잡도 역시 동적 배열이나 추가 배열없이 주어진 배열과 변수 내에서 진행되기 때문에 O(1)이다. 달라진 점은 함수에 인자를 넣은 부분이다.

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

for i from 0 to BLOCK\_HEIGHT-1

for j from 0 to BLOCK\_WIDTH-1

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

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

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

}

텍스트, 스크린샷이(가) 표시된 사진

자동 생성된 설명

상수인 BLOCK\_HEIGHT와 BLOCK\_WIDTH만큼 반복하기 때문에 시간복잡도는 O(1)이고 공간복잡도는 추가로 입력이 없기 때문에 마찬가지로 O(1)이다. 달라진 점은 없다.

int DeleteLine(char field[HEIGHT][WIDTH]){

ret=0

check=FAlSE

for i from 0 to HEIGHT-1

check=TRUE

for j from 0 to WIDTH-1

if(field[i][j]==0)

check=FALSE

if(check==TRUE)

ret++

for x from i-1 to 0

for y from 0 to WIDTH-1

field[x+1][y]=field[x][y]

return ret\*ret\*100;

}

텍스트, 스크린샷이(가) 표시된 사진

자동 생성된 설명

HEIGHT와 WIDTH만큼 반복하기 때문에 시간복잡도는 O(HEIGHT\*WIDTH)이고 공간복잡도는 추가로 입력이 없기 때문에 마찬가지로 O(1)이다. 달라진 점은 없다.

void BlockDown(int sig){

if(CheckToMove()==1)

blockY++

Drawchange()

Else

If(blockY==-1)

gameOver=1

AddBlockToField()

score+=DeleteLine()

nextBlock[0]=nextBlock[1]

nextBlock[1]=rand()%7

DrawNextBlock()

PrintScore()

DrawField()

}

텍스트, 스크린샷, 소프트웨어이(가) 표시된 사진

자동 생성된 설명

호출하는 함수들 중 Deleteline()은 O(HEIGHT\*WIDTH)의 시간 복잡도를 가지고 있고 나머지 함수는 모두 상수의 시간복잡도를 가지고 있기 때문에 시간복잡도는 O(HEIGHT\*WIDTH)이고 공간복잡도는 마찬가지로 기존의 전역변수들을 사용하기 때문에 O(1)이다. 달라진 점은 함수의 인자를 넣은 부분이다.

2.

Void InitTetris(){

----------기존 코드와 동일----------

nextBlock[0]=rand()%7;

nextBlock[1]=rand()%7;

nextBlock[2]=rand()%7;

blockRotate=0;

blockY=-1;

blockX=WIDTH/2-2;

----------기존 코드와 동일----------

}

시간 복잡도 O(1) 공간복잡도 O(1)

Void DrawOutline(){

----------기존 코드와 동일----------

DrawBox(3,WIDTH+10,4,8);

DrawBox(10,WIDTH+10,4,8);

move(16,WIDTH+10);

printw("SCORE");

DrawBox(17,WIDTH+10,1,8);

}

시간 복잡도 O(1) 공간복잡도 O(1)

Void DrawChange(){

----------기존 코드와 동일----------

while(CheckToMove(field,currentBlock,rot,y,x)) y++;

y--;

for(i=0;i<BLOCK\_HEIGHT;i++){

for(j=0;j<BLOCK\_WIDTH;j++){

if(block[currentBlock][rot][i][j]==1 && i+y>=0){

move(i+y+1,j+x+1);

printw(".");

}

}

}

DrawBlockWithFeatures(blockY, blockX, currentBlock, blockRotate);

----------기존 코드와 동일----------

}

시간 복잡도 O(1) 공간 복잡도 O(1)

Void blockdown(){

----------기존 코드와 동일----------

score+=DeleteLine(field);

nextBlock[0]=nextBlock[1];

nextBlock[1]=nextBlock[2];

nextBlock[2]=rand()%7;

DrawNextBlock(nextBlock);

blockY=-1;

blockX=WIDTH/2-2;

blockRotate=0;

----------기존 코드와 동일----------

}

시간복잡도 O(HEIGHT\*WIDTH) 공간복잡도 O(1)

void DrawShadow(int y, int x, int blockID,int blockRotate){

while(CheckToMove(field,blockID,blockRotate,y,x)) y++;

y--;

DrawBlock(y,x,blockID,blockRotate,'/');

}

시간복잡도 O(1) 공간복잡도 O(1)

void DrawBlockWithFeatures(int blockY, int blockX, int blockID, int blockRotate){

DrawShadow(blockY, blockX, blockID, blockRotate);

DrawBlock(blockY, blockX, blockID, blockRotate, ' ');

}

시간복잡도 O(1) 공간복잡도 O(1)