|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **MINI-GAME-CENTER**  **PORTFOLIE**    팀장 : 함영식  팀원 : 이로운  김상아  박지수  2012.09.06 | | | | |
| **목 차**  **1. 프로그램 개요**  ① 팀 소개  ② 개발 배경 및 목표  **2. 개발 환경**  ① 시스템 설계  ② 개발일정  ③ 업무분담  **3. 분석 / 설계**  ① 벤치마킹  ② 화면설계  ③ 데이터설계  ④ 파일설계  ⑤ DFD(Data-Flow-Diagram)  ⑥ 함수설계  **4. 결론** | | | | |
| **1. 프로그램 개요**  ① 팀 소개  ② 개발 배경 및 목표 | | | | |
| **MINI-GAME-CENTER** | | | |
| **문서제목** | 팀소개 | **팀명** | GameOver |
| **문서번호** | 1 | **작성일자** |  |
| **프로젝트단계** | 분석설계단계 | **작성자** | GameOver |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **[ 팀명 ] GameOver**  **[의미]**    1.박스 데이터설계  struct BOX{  char MAP[25][40];//맵의 최대크기  int object[2];//객체의 좌표저장  int stage[3];//0번방-stage,1번방-맵의행,2번방-맵의열  int goal[10][2];//골인지점의 좌표  };  **2.박스 파일설계**    **N-빈공간,B-벽,T-박스,G-골지점**  **3.박스 DFD**    **4.박스 함수설계**   |  |  | | --- | --- | | **boxGame** | | | int | **boxGame main()**  사용자에게 원하는 stage를 입력받고 키를 입력받으며 게임에 필요한 함수 호출,사용자의 stage를 리턴해줌 | | void | **loadMap(Box \*box,int \*gCnt)**  텍스트 파일을 읽어와 box구조체에 map배열의 문자,goal배열에 좌표를 저장,gCnt골의 개수를 세준다. | | void | **blockDisplay(Box \*box)**  box구조체의 map배열의 문자중 B의 문자의 위치에 벽모양 출력 | | void | **moveDisplay(Box \*box)**  box구조체의 map배열의 문자중 C,T,F,N에 해당되는 define된 문자열출력 | | void | **Move(Box \*box,it direction,int caseNum)**  direction은 캐릭터가 움직이는 방향을 나타내며 box구조체의  object배열(객체의좌표)의 값을 이용해서 객체가 움직일수 있는 2가지종류를 1(객체혼자이동),2(객체와박스함께이동) caseNum과 비교하여 움직인다. | | int | **goalCheck(Box \*box,int gCnt)**  box의 구조체의 goal배열에 좌표에 T(박스)가 위치하고 있는지 확인하여  골인모양을 출력하고 변수 count를 이용하여 gCnt와 같은지 확인하여  같을시 stage[0]번방의 숫자를 늘려주고 리턴1 아닐시 리턴 0 | | void | **upKey(Box \*box)**  box구조체의 객체의 좌표 위가 비어있거나 골인지점일 경우(case 1)  move(box,up,1)호출  객체의 위에 박스가 있고 그위가 비어있거나 골인지점일 경우(case 2)  move(box,up,2)호출 | | void | **downKey(Box \*box)**  box구조체의 객체의 좌표 아래가 비어있거나 골인지점일 경우(case 1)  move(box,down,1)호출  객체의 아래에 박스가 있고 그아래가 비어있거나 골인지점일 경우(case 2)  move(box,down,2)호출 | | void | **leftKey(Box \*box)**  box구조체의 객체의 좌표 왼쪽이 비어있거나 골인지점일 경우(case 1)  move(box,left,1)호출  객체의 왼쪽에 박스가 있고 그왼쪽이 비어있거나 골인지점일 경우(case 2)  move(box,left,2)호출 | | void | **rigthKey(Box \*box)**  box구조체의 객체의 좌표 오른쪽이 비어있거나 골인지점일 경우(case 1)  move(box,right,1)호출  객체의 오른쪽에 박스가 있고 그오른쪽이 비어있거나 골인지점일 경우(case 2)  move(box,right,2)호출 |   **5.박스 NS**   |  |  |  |  |  |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | | boxGame main() 시작 | | | | | | | | | | | | | | | BOX Box={0},int step=0,char ch  int keyFlag, res=0, gCnt=0 i=0; | | | | | | | | | | | | | | | printf("몇 탄부터 시작하시겠습니까?“) 출력 | | | | | | | | | | | | | | | scanf("%d",&box->stage[0]) // 사용자에게 탄수를 입력받는다 | | | | | | | | | | | | | | | while(1) | | | | | | | | | | | | | | |  | loadMap(&Box,&gCnt); // 파일에서 함수 호출 | | | | | | | | | | | | | | blockDisplay(&Box) //벽 출력해주는 함수 호출 | | | | | | | | | | | | | | moveDisplay(&Box) | | | | | | | | | | | | | | goalcheck(&Box,gCnt) | | | | | | | | | | | | | | help(&Box) | | | | | | | | | | | | | | while(1) | | | | | | | | | | | | | |  | ch=inkey() //특수키 입력받기 | | | | | | | | | | | | | switch(ch) | | | | | | | | | | | | | case: | Up | | | upkey(&Box); //위쪽 방향키upkey함수 호출  break; | | | | | | | | | Down | | | downkey(&Box); //아래쪽 방향키downkey함수 호출  break; | | | | | | | | | Left | | | leftkey(&Box); //왼쪽 방향키 leftkey함수 호출  break; | | | | | | | | | Right | | | rightkey(&Box); //오른쪽 방향키 rightkey함수 호출  break; | | | | | | | | | space | | | break;//space입력시 종료 | | | | | | | | | esc | | | break; | | | | | | | | | moveDisplay(&Box) | | | | | | | | | | | | | step=goalCheck(&Box,gCnt) | | | | | | | | | | | | | yes | | | | | | ch==space?||step==1? | | | | | no | | y | | Box.stage[0]==11? | | | | n | |  | | | | | res=1 | | |  | | | | |  | | | | | break; | | |  | | | | | | gCnt=0 | | | | | | | | | | | | | break | | | | | | | | | | | | | yes | | | | | | ch==ESC? | | | | | no | | res=1; //esc입력 확인 변수  break; //종료 while문 탈출 | | | | | | | | |  | | | | yes | | | | | | res==1? | | | | | no | | | break; //esc입력시 종료 while문 완전 탈출 | | | | | | | |  | | | | | | return (Box.stage[0])-1//사용자의 기록리턴 | | | | | | | | | | | | | |  |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | --- | | void loadMap(BOX \*box,int \*gCnt) 시작 | | | | | | | | | | int I,j선언 char str[100] 텍스트 파일 불러올 변수 선언 | | | | | | | | | | 맵을 읽어올 FILE \* ifp 선언 | | | | | | | | | | sprintf("str, "C:\\data\\Map\_%d.txt", sp->num); str문자열에 파일위치저장 | | | | | | | | | | ifp = fopen(str, "rt"); str배열에 저장된 맵 읽기모드로 열기 | | | | | | | | | | fscanf(ifp, "%d %d", box->stage[1],box->stage[2]) 맵의 행, 열수 읽어와서 저장 | | | | | | | | | | i=0부터 box->stage[1]보다 작을때까지 i++ | | | | | | | | | |  | j=0부터 box->stage[2]보다 작을때까지 j++ | | | | | | | | |  | fscanf(ifp, "%c ",box-> map[i][j]); | | | | | | | | yes | box->map[i][j]==’G’? | | | no | | | | box->goal[gCnt][0]=i | | yes | box->map[i][j]==’\n’? | | | no | | box->goal[gCnt][1]=j | | box->map[i][j]  =NULL | | |  | | | (\*gCnt)++ | | break | | |  | | | fclose(ifp) | | | | | | | | |  |  |  |  |  |  | | --- | --- | --- | --- | --- | | void blockDisplay(BOX \*box) 시작 | | | | | | int i,j 변수 선언 | | | | | | for(i=0; i<box->stage[1]; i++) | | | | | |  | for(j=0; j<box->stage[2]; j++) | | | | |  | yes | box->map[i][j]=='B'? | no | | printf("%s",10); | | |  |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | --- | | void moveDisplay(BOX \*box) 시작 | | | | | | | | | | int i, j,count=0선언 | | | | | | | | | | for(i=0; i<box->stage[1]; i++) | | | | | | | | | |  | for(j=0; j<=box->stage[2]; j++) | | | | | | | | |  | yes | box->map[i][j]=='C'? | | | | no | | | printf("%s",11)//11번으로 define된 문자열출력 | | |  | | | | | yes | box->map[i][j]=='T'? | | | | no | | | printf("%s",12)//12번으로 define 된 문자열출력 | | |  | | | | | yes | box->map[i][j]=='G'? | | | | no | | | printf("%s",13)//13번으로 define 된 문자열출력 | | | |  | | | | yes | | box->map[i][j]=='N'? | | | | no | | printf(" ") //공백 두칸 짜리 출력 | | | | | | |  |  |  |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | | move(BOX \*box,int direction, int caseNum)함수시작 | | | | | | | | | | | | low=box->object[0],col=box->object[1] | | | | | | | | | | | | int num; | | | | | | | | | | | | y | | direction==UP||direction==LEFT? | | | | | | | n | | | num=-1 | | | | | | num=1 | | | | | | y | | direction==UP||direction==DOWN? | | | | | | | n | | | y | caseNum==1? | | | n | y | | caseNum==1? | | | n | | Box->map  [low+num][col]=  box->map  [low][col] | | | box->map[low+num\*2][col]=box->map[low+num][col] | | box->map  [low][col+num]=  box->map[low][col] | | | box->map[low][col+num\*2]=  box->map[low][col+num] | | | |  | | | box->map  [low+num][col]=  box->map  [low][col] | |  | | | box->map[low][col+num]=  box->map[low][col] | | | | box->map[low][col]='N' | | | | | | | | | | |  |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | --- | | int goalcheck(BOX \*box,int gCnt) | | | | | | | | | | int I,count=0,char ch | | | | | | | | | | for(i=0;i<gCnt;i++) | | | | | | | | | |  | yes | box->map[box->goal[i][0]][box->goal[i][1]]==’T’? | | | | no | | | | gotoxy() | | | yes | box->map[box->goal[i][0]][box->goal[i][0]!=’C’? | | | no | | printf(“★”) | | | box->map[box->goal[i][0]][box->goal[i][1]]=’G’ | |  | | | | count++ | | |  | | | | | | yes | | | gCnt==count? | | | | no | | | box->stage[0]++ | | | | return 0 | | | | | | return 1 | | | |  | | | | |  |  |  |  |  |  | | --- | --- | --- | --- | --- | | void Upkey(BOX \*box)시작 | | | | | | int i,j, row, col 선언 //row와 col은 object에 저장된 행 길이와 열 크기를 집어 넣는다 | | | | | | row=box->object[0];, col=box->object[1]; //행열값 집어넣기 | | | | | | yes | box->map[row-1][col]=='N'|| box->map[row-1][col]=='G'? | | | no | | move(box,up,1) 호출 | |  | | | | yes | box->map[row-1][col]=='T'&&  (box->map[row-2][col]=='N'||box->map[row-2][col]=='G')? | | | no | | move(box,up,2) 호출 | | |  | |  |  |  |  |  |  | | --- | --- | --- | --- | --- | | void downkey(BOX \*box)시작 | | | | | | int i,j, row, col 선언 //row와 col은 object에 저장된 행 길이와 열 크기를 집어 넣는다 | | | | | | row=box->object[0];, col=box->object[1]; //행열값 집어넣기 | | | | | | yes | box->map[row+1][col]=='N'|| box->map[row+1][col]=='G'? | | | no | | move(box,down,1) 호출 | |  | | | | yes | box->map[row+1][col]=='T'&&  (box->map[row+2][col]=='N'||  box->map[row+2][col]=='G')? | | | no | | move(box,down,2) 호출 | | |  | |  |  |  |  |  | | --- | --- | --- | --- | | void leftkey(BOX \*box)시작 | | | | | int i,j, row, col 선언 //row와 col은 object에 저장된 행 길이와 열 크기를 집어 넣는다 | | | | | row=box->object[0];, col=box->object[1]; //행열값 집어넣기 | | | | | yes | box->map[row][col-1]=='N'|| box->map[row][col-1]=='G'? | | no | | move(box,left,1) 호출 | |  | | | yes | box->map[row][col-1]=='T'&&  (box->map[row][col-2]=='N'||box->map[row][col-2]=='G')? | | no | | move(box,left,2) 호출 | |  | |  |  |  |  |  | | --- | --- | --- | --- | | void rightkey(BOX \*box)시작 | | | | | int i,j, row, col 선언 //row와 col은 object에 저장된 행 길이와 열 크기를 집어 넣는다 | | | | | row=box->object[0];, col=box->object[1]; //행열값 집어넣기 | | | | | yes | box->map[row-1][col]=='N'|| box->map[row-1][col]=='G'? | | no | | move(box,right,1) 호출 | |  | | | yes | box->map[row-1][col]=='T'&&  (box->map[row-2][col]=='N'||box->map[row-2][col]=='G')? | | no | | move(box,right,2) 호출 | |  | |   **화면설계!!!!!!**    게임시작시-사용자이름입력받음    게임시작시-로고    F1->게임설명    미로-게임선택      위>타자게임시작,아래>실행화면        위>박스게임시작,아래>실행화면        위>행맨게임시작,아래>실행화면 |