|  |  |  |  |
| --- | --- | --- | --- |
| **프로젝트명 넣을것** | | | |
| **문서제목** | 화면설계... | **팀명** |  |
| **문서번호** | 3 | **작성일자** | 2012..xx.xx |
| **프로젝트단계** | 분석설계단계... | **작성자** | 홍길동 |

|  |
| --- |
|  |
| **C:\Users\cmj\Desktop\al\word.PNG**  beginDisplay()  기능 : 초기화면을 찍어준다  전달인자 : void  리턴값 : void  wordLoad()  기능 : 단어.txt에서 단어를 읽어와서 word구조체 배열에 저장하고 ok를 0, 단어의 길이를 초기화  전달인자 : word(구조체배열)  리턴값 : void  floorFrame()  기능 : 바닥의 틀을 찍어준다(딱 한번)  전달인자 : void  리턴값 : void  Stringcheck()  기능 : enter키 입력시 호출되어 입력한 단어를 화면에 출력중인(ok = 1) 단어와 비교하여 일치하면 ok를 0으로 변경하고 잔상을 지워주고 점수를 올려준다  전달인자 : word(구조체배열), &score(점수), inputw(입력한단어)  리턴값 : void  wordOk()  기능 : 랜덤으로 하나의 단어를 화면에 출력하기 위해 ok를 1로 변경, x축은 랜덤으로 y축은 1로  전달인자 : word(구조체배열)  리턴값 : void  yPlus()  기능 : 구조체배열(100개의단어)에서 출력중인 단어(ok=1)의 y축을 1증가  전달인자 : word(구조체배열)  리턴값 : void  floorUp()  기능 : 화면에 출력중인 단어(ok=1)중에 바닥에 닿은 단어를 검사해서 잔상을 지우고 바닥값을 1감소  전달인자 : &floor(바닥값), word(구조체배열)  리턴값 : 1이면 게임오버  floorDisplay()  기능 : 바닥값을 받아서 바닥의 틀을 찍어주고 메시지를 출력  전달인자 : &floor(바닥값)  리턴값 : void  wordDisplay()  기능 : 구조체배열(100개의 단어)에서 ok가 1인 단어를 화면에 출력하고 잔상을 지워준다  전달인자 : word(구조체배열)  리턴값: void  C:\Users\cmj\Desktop\al\push.PNG  PushGame()  #define UP 1  #define DOWN 2  #define LEFT 3  #define RIGHT 4  #define PPP "▧￠E"  #define SPECIAL\_KEY 0  #define COMMON\_KEY 1  #define UP\_ARROW 72  #define DOWN\_ARROW 80  #define LEFT\_ARROW 75  #define RIGHT\_ARROW 77  #define ENTER 13  #define ESC 27  #define SPACE\_BAR 32  struct BOX{  char MAP[25][40];//맵퇔모모양푖  int object[2];//객객체A의퍪C좌AA표C￥  int stage[3];//게캯임AO탄A수볼o,행Ca,열열  int goal[10][2];//골컉인AI지Ao점A? 좌AA표C￥  };  필요한구조체 및 define  loadMap()  기능: 파일에 있는 맵 정보를 2차원 Map[][]배열에 저장하고 goal배열에 Goal지점을 저장.  전달인자 : Box \*box,int \*gCnt  리턴값 :void  blockDisplay()  기능: 벽을 출력해주는 함수  전달인자 : Box \*box  리턴값 : void  moveDisplay()  기능 : 객체, 골, 박스, 빈공간, 완전체 출력.  전달인자 :Box \*box  리턴값: int \*  goalcheck()  기능 : goal지점에 좌표에 박스의 존재 유무확인  전달인자 : Box \*box,int gCnt  리턴값 : int  ~~>>리턴값이 1일때는  goal의 배열값이 모두 T일경우 . return 1.  //리턴값이 1일경우를 검사할때는 stage값을올려주어야한다  ~~>>리턴값이 0일경우는  goal의 배열값이 모두 T가 아닐경우    leftkey()  기능 : 왼쪽 방향키를 입력했을때 검사를 하고 이동가능여부를 검사하고 가능한 경우 move함수 호출  검사 가능한 경우의 수가 두가지  case1: 객체 다음칸이 빈칸이거나 goal 일경우  case2: 객체 다음칸에 박스 이면서 그 다음칸이 goal이거나 빈칸일경우  전달인자 : Box \*box  리턴값 void  rightkey()  기능 : 오른쪽 방향키를 입력했을때 검사를 하고 이동가능여부를 검사하고 가능한 경우 move함수 호출  검사 가능한 경우의 수가 두가지  case1: 객체 다음칸이 빈칸이거나 goal 일경우  case2. 객체 다음칸에 박스 이면서 그 다음칸이 goal이거나 빈칸일경우  전달인자 : Box \*box  리턴값 void  upkey()  기능 : 위쪽 방향키를 입력했을때 검사를 하고 이동가능여부를 검사하고 가능한 경우 move함수 호출  검사 가능한 경우의 수가 두가지  case1: 객체 다음칸이 빈칸이거나 goal 일경우  case2. 객체 다음칸에 박스 이면서 그 다음칸이 goal이거나 빈칸일경우    전달인자 : Box \*box  리턴값 void  downkey()  기능 : 아래쪽 방향키를 입력했을때 검사를 하고 이동가능여부를 검사하고 가능한 경우 move함수 호출  검사 가능한 경우의 수가 두가지  case1: 객체 다음칸이 빈칸이거나 goal 일경우  case2. 객체 다음칸에 박스 이면서 그 다음칸이 goal이거나 빈칸일경우  전달인자 : Box \*box  리턴값 void  move()  기능: 객체를 움직이게 해주는 함수  case1:객체만움직임  case2:객체,상자같이 움직임  전달인자 : Box \*box,int direction,int caseNum  리턴값 : void  caseNum:위쪽과 왼쪽은 -1 ,아래쪽과오른쪽은 +1  direction: define된 방향값을 사용  C:\Users\cmj\Desktop\al\hang.PNG  HangGame()  FileLoad()  기능: 더블링크드에 파일에서 읽어온 단어들을 저장시킴  전달인자: Linkedlist \*list  리턴값 : void  Display()  기능: 알파벳을 출력해준다  전달인자: char\* ary,hWord\* choiceWord  리턴값 : void  hangDisplay()  기능 : 사람을 보여주는데 메인에서 선언된 hangMan=7 에서 틀린 숫자만큼 위험수위를 보여준다.  리턴값 : void  전달인자 : int \*hangMan  wordDisplay()  기능 : 입력값이 choiceWord랑 같은게 있으면 alphabet모음에서 O로 대체하고 빈칸을 적어준다. 그리고 같은게 없으면 X로 대체, 입력받은 알파벳이 골라진 단어 철자중에 있으면 flag값을 1로 넣어주고 flag를 리턴해준다.  리턴값 :flag 리턴  전달인자 : hWord\* choiceWord , char ch,char \*ary, int \*sCnt  choiceword()  기능: 링크드리스트로연결된 워드 구조체를 중에서 하나를 랜덤으로 골라서 리턴해준다  전달인자 : LinkedList\* list  리턴값 : hword;  Input()  기능 : a~z까지만 키를 입력받아서 반환  F1일 경우도 (특수키)반환  전달인자 : x  리턴값 : ch(입력받은 문자)  hintDisplay()  기능 : F1키를 입력받으면 실행되는데 choiceWord에서 골라진 단어의 힌트를 보여주고 힌트를 사용하면 hint = 20 에서 1씩 감소시킨다  전달인자 : hWord choiceWord, int \*hint  리턴값 : void  C:\Users\cmj\Desktop\al\mazee.PNG  **mazeGame()**  #define UP 1  #define DOWN 2  #define LEFT 3  #define RIGHT 4  #define PPP "▧￠E"  #define SPECIAL\_KEY 0  #define COMMON\_KEY 1  #define UP\_ARROW 72  #define DOWN\_ARROW 80  #define LEFT\_ARROW 75  #define RIGHT\_ARROW 77  #define ENTER 13  #define ESC 27  #define SPACE\_BAR 32  struct BOX2{  char MAP[25][40];//맵퇔모모양푖  int object[2];//객객체A의퍪C좌AA표C￥  int stage[3];//게캯임AO탄A수볼o,행Ca,열열  int goal[10][2];//골컉인AI지Ao점A? 좌AA표C￥  };  필요한구조체 및 define  loadMap()  기능: 파일에 있는 맵 정보를 2차원 Map[][]배열에 저장하고 goal배열에 Goal지점을 저장.  전달인자 : Box \*box,int \*gCnt  리턴값 :void  blockDisplay()  기능: 벽과 골인지점 출력  전달인자 : Box \*box  리턴값 : void  moveDisplay()  기능 : 객체, 골, 박스, 빈공간, 완전체 출력.  전달인자 :Box \*box  리턴값: int \*  goalcheck()  기능 : goal지점에 좌표에 객체 유무확인  전달인자 : Box \*box,int gCnt  리턴값 : int  leftkey()  기능 : 왼쪽 방향키를 입력했을때 검사를 하고 이동가능여부를 검사하고 가능한 경우 move함수 호출  검사 가능한 경우의 수가 두가지  case1: 객체 다음칸이 빈칸이거나 goal 일경우  전달인자 : Box \*box  리턴값 void  rightkey()  기능 : 오른쪽 방향키를 입력했을때 검사를 하고 이동가능여부를 검사하고 가능한 경우 move함수 호출  검사 가능한 경우의 수가 두가지  case1: 객체 다음칸이 빈칸이거나 goal 일경우  전달인자 : Box \*box  리턴값 void  upkey()  기능 : 위쪽 방향키를 입력했을때 검사를 하고 이동가능여부를 검사하고 가능한 경우 move함수 호출  검사 가능한 경우의 수가 두가지  case1: 객체 다음칸이 빈칸이거나 goal 일경우    전달인자 : Box \*box  리턴값 void  downkey()  기능 : 아래쪽 방향키를 입력했을때 검사를 하고 이동가능여부를 검사하고 가능한 경우 move함수 호출  검사 가능한 경우의 수가 두가지  case1: 객체 다음칸이 빈칸이거나 goal 일경우  전달인자 : Box \*box  리턴값 void  move()  기능: 객체를 움직이게 해주는 함수  case1:객체만움직임  전달인자 : Box \*box,int direction,int caseNum  리턴값 : void  caseNum:위쪽과 왼쪽은 -1 ,아래쪽과오른쪽은 +1  direction: define된 방향값을 사용 |