65010065 กันตินันท์

#include <stdio.h>

#include <windows.h>

#include <time.h>

#define screen\_x 80

#define screen\_y 25

#define scount 80

using namespace std;

HANDLE rHnd;

HANDLE wHnd;

DWORD fdwMode;

CHAR\_INFO consoleBuffer[screen\_x \* screen\_y];

COORD characterPos = { 0,0 };

COORD bufferSize = { screen\_x,screen\_y };

SMALL\_RECT windowSize = { 0,0,screen\_x-1,screen\_y-1 };

struct Ship{

int X\_middle=0;

int Y\_middle=0;

int X\_Lwing = 0;

int X\_Rwing = 0;

int X\_Rd = 0;

int X\_Ld = 0;

int color = 7;

int life = 10;

};

Ship ship;

COORD star[scount];

int setMode(){

rHnd = GetStdHandle(STD\_INPUT\_HANDLE);

fdwMode = ENABLE\_EXTENDED\_FLAGS | ENABLE\_WINDOW\_INPUT | ENABLE\_MOUSE\_INPUT;

SetConsoleMode(rHnd, fdwMode);

return 0;

}

int setConsole(int x, int y){

wHnd = GetStdHandle(STD\_OUTPUT\_HANDLE);

SetConsoleWindowInfo(wHnd, TRUE, &windowSize);

SetConsoleScreenBufferSize(wHnd, bufferSize);

return 0;

}

void clear\_buffer(){

for(int y = 0; y < screen\_y; ++y){

for(int x = 0; x < screen\_x; ++x){

consoleBuffer[x + screen\_x \* y].Char.AsciiChar = ' ';

consoleBuffer[x + screen\_x \* y].Attributes = 0;

}

}

}

void fill\_buffer\_to\_console(){

WriteConsoleOutputA(wHnd, consoleBuffer, bufferSize, characterPos,&windowSize);

}

void ship\_to\_buffer(){

consoleBuffer[ship.X\_middle + screen\_x \* ship.Y\_middle].Char.AsciiChar ='0';

consoleBuffer[ship.X\_Lwing + screen\_x \* ship.Y\_middle].Char.AsciiChar ='<';

consoleBuffer[ship.X\_Rwing + screen\_x \* ship.Y\_middle].Char.AsciiChar ='>';

consoleBuffer[ship.X\_Rd + screen\_x \* ship.Y\_middle].Char.AsciiChar ='-';

consoleBuffer[ship.X\_Ld + screen\_x \* ship.Y\_middle].Char.AsciiChar ='-';

consoleBuffer[ship.X\_middle + screen\_x \* ship.Y\_middle].Attributes =ship.color;

consoleBuffer[ship.X\_Lwing + screen\_x \* ship.Y\_middle].Attributes =ship.color;

consoleBuffer[ship.X\_Rwing + screen\_x \* ship.Y\_middle].Attributes =ship.color;

consoleBuffer[ship.X\_Rd + screen\_x \* ship.Y\_middle].Attributes =ship.color;

consoleBuffer[ship.X\_Ld + screen\_x \* ship.Y\_middle].Attributes =ship.color;

}

void init\_star(){

for(int i = 0; i < scount; i++){

star[i].Y = rand() % 25;

star[i].X = rand() % 80;

}

}

void star\_fall(){

int i;

for (i = 0; i < scount; i++) {

if (star[i].Y >= screen\_y-1) {

star[i] = { (rand() % screen\_x),1 };

}

else {

star[i] = { star[i].X,star[i].Y+1 };

}

}

}

void fill\_star\_to\_buffer(){

for(int i = 0; i < scount; ++i){

consoleBuffer[star[i].X + screen\_x \* star[i].Y].Char.AsciiChar = '\*';

consoleBuffer[star[i].X + screen\_x \* star[i].Y].Attributes = 15;

}

}

void checkcol(){

for(int i = 0;i<scount;i++){

if(ship.X\_middle == star[i].X || ship.X\_Rwing == star[i].X || ship.X\_Lwing == star[i].X || ship.X\_Rd == star[i].X || ship.X\_Ld == star[i].X){

if(ship.Y\_middle == star[i].Y){

star[i].X = rand() % screen\_x;

star[i].Y = rand() % screen\_y;

ship.life -=1;

}

}

}

}

int main(){

srand(time(NULL));

bool play = true;

DWORD numEvents = 0;

DWORD numEventsRead = 0;

setConsole(screen\_x, screen\_y);

setMode();

init\_star();

while (play){

GetNumberOfConsoleInputEvents(rHnd, &numEvents);

if (numEvents != 0) {

INPUT\_RECORD\* eventBuffer = new INPUT\_RECORD[numEvents];

ReadConsoleInput(rHnd, eventBuffer, numEvents, &numEventsRead);

for (DWORD i = 0; i < numEventsRead; ++i) {

if (eventBuffer[i].EventType == KEY\_EVENT &&

eventBuffer[i].Event.KeyEvent.bKeyDown == true ) {

if (eventBuffer[i].Event.KeyEvent.wVirtualKeyCode == VK\_ESCAPE) {

play = false;

}

if(eventBuffer[i].Event.KeyEvent.uChar.AsciiChar == 'c'){

ship.color = rand() % 7;

}

//printf("press : %c\n", eventBuffer[i].Event.KeyEvent.uChar.AsciiChar);

}

else if (eventBuffer[i].EventType == MOUSE\_EVENT) {

int posx = eventBuffer[i].Event.MouseEvent.dwMousePosition.X;

int posy = eventBuffer[i].Event.MouseEvent.dwMousePosition.Y;

if (eventBuffer[i].Event.MouseEvent.dwButtonState &

FROM\_LEFT\_1ST\_BUTTON\_PRESSED) {

//printf("left click\n");

ship.color = rand() % 7;

}

else if (eventBuffer[i].Event.MouseEvent.dwButtonState &

RIGHTMOST\_BUTTON\_PRESSED) {

//printf("right click\n");

}

else if (eventBuffer[i].Event.MouseEvent.dwEventFlags & MOUSE\_MOVED) {

//printf("mouse position : (%d,%d)\n",posx, posy);

ship.X\_middle = posx;

ship.Y\_middle = posy;

ship.X\_Lwing = posx-2;

ship.X\_Rwing = posx+2;

ship.X\_Ld = posx-1;

ship.X\_Rd = posx+1;

}

}

}

delete[] eventBuffer;

}

clear\_buffer();

ship\_to\_buffer();

star\_fall();

fill\_star\_to\_buffer();

checkcol();

fill\_buffer\_to\_console();

if(ship.life < 1) play = false;

Sleep(100);

}

return 0;

}

