|  |
| --- |
| #include <stdio.h>  #include <windows.h>  #include <conio.h>  #include <time.h>    void gotoxy(int x, int y)  {  COORD c = { x, y };  SetConsoleCursorPosition(GetStdHandle(STD\_OUTPUT\_HANDLE), c);  }    void draw\_ship(int x, int y)  {  COORD c = { x, y };  SetConsoleCursorPosition(GetStdHandle(STD\_OUTPUT\_HANDLE), c);  printf(" <-0-> ");  }    void setcursor(bool visible)  {  HANDLE console = GetStdHandle(STD\_OUTPUT\_HANDLE);  CONSOLE\_CURSOR\_INFO lpCursor;  lpCursor.bVisible = visible;  lpCursor.dwSize = 20;  SetConsoleCursorInfo(console, &lpCursor);  }    void setcolor(int fg, int bg)  {  HANDLE hConsole = GetStdHandle(STD\_OUTPUT\_HANDLE);  SetConsoleTextAttribute(hConsole, bg \* 16 + fg);  }    void erase\_ship(int x, int y)  {  COORD c = { x, y };  SetConsoleCursorPosition(GetStdHandle(STD\_OUTPUT\_HANDLE), c);  setcolor(0, 0);  printf("        ");  }    void erase\_bullet(int x, int y)  {  COORD a = { x, y };  SetConsoleCursorPosition(GetStdHandle(STD\_OUTPUT\_HANDLE), a);  setcolor(0, 0);  printf(" ");  }    void erase\_star(int x, int y)  {  COORD b = { x, y };  SetConsoleCursorPosition(GetStdHandle(STD\_OUTPUT\_HANDLE), b);  setcolor(0, 0);  printf(" ");  }    char cursor(int x, int y)  {  HANDLE hStd = GetStdHandle(STD\_OUTPUT\_HANDLE);  char buf[2]; COORD c = { x,y }; DWORD num\_read;  if (!ReadConsoleOutputCharacter(hStd, (LPTSTR)buf, 1, c, (LPDWORD)&num\_read))  return '\0';  else  return buf[0];  }    int main()  {  srand(time(NULL));  char ch = '.';  int x = 38, y = 20;  int xbullet[100], ybullet[100];  int num = 1, point = 0;    setcolor(2, 4);  setcursor(0);  draw\_ship(x, y);    int direction = 0;  int shoot[100], bullet = 0, ammo = 50;      do {  if (\_kbhit()) {  ch = \_getch();  if (ch == 'a') { direction = 1; }  else if (ch == 'd') { direction = 2; }  else if (ch == 's') { direction = 0; }    if (ch == ' ') {  Beep(700, 400);  if (bullet < ammo) {  bullet += 1;  ybullet[bullet] = 19;  xbullet[bullet] = x + 3;  shoot[bullet] = 1;  }  }  fflush(stdin);  }    //move  if (direction == 1 && x != 0) {  erase\_ship(x, y);  setcolor(2, 4);  draw\_ship(--x, y);  }  else if (direction == 2 && x != 73) {  erase\_ship(x, y);  setcolor(2, 4);  draw\_ship(++x, y);  }    //star  if (num <= 20)  {  int stayrand = rand() % 4 + 2;  int staxyrand = rand() % 61 + 10;  setcolor(2, 0);  gotoxy(staxyrand, stayrand);  printf("\*");  Sleep(50);  num++;  }    //shoot  for (int i = 1; i <= bullet; i++)  {  if (shoot[i] == 1) {  erase\_bullet(xbullet[i], ybullet[i]);  if (ybullet[i] > 0) {  if (cursor(xbullet[i], ybullet[i] - 1) == '\*')  {  Beep(1000, 400);  num = num - 1;  erase\_bullet(xbullet[i], ybullet[i]);  erase\_bullet(xbullet[i], ybullet[i] - 1);  shoot[i] = 0;  point += 1;  }  else  {  gotoxy(xbullet[i], --ybullet[i]);  setcolor(2, 4);  printf("|");  }    }  else {  shoot[i] = 0;  }  }  }  setcolor(9, 0);  gotoxy(0, 0);  printf("%d", point);    Sleep(100);  } while (ch != 'x');    return 0;  } |

Graphical user interface, application

Description automatically generated