#include <iostream>

#include <windows.h>

#include <conio.h>

using namespace std;

HANDLE h = GetStdHandle(STD\_OUTPUT\_HANDLE);

COORD smile1{};

COORD smile2{};

int code;

void lrletter()

{

smile2.X--;

SetConsoleCursorPosition(h, smile2);

cout << " ";

if (code == 100)

{

SetConsoleCursorPosition(h, smile2);

SetConsoleTextAttribute(h, 13);

cout << (char)1;

}

SetConsoleTextAttribute(h, 4);

cout << (char)3;

SetConsoleCursorPosition(h, smile1);

SetConsoleTextAttribute(h, 14);

cout << (char)1;

if (code == 97)

{

smile2.X += 2;

SetConsoleCursorPosition(h, smile2);

SetConsoleTextAttribute(h, 13);

cout << (char)1;

}

}

void lrarrow()

{

smile1.X--;

SetConsoleCursorPosition(h, smile1);

cout << " ";

if (code == 77)

{

SetConsoleCursorPosition(h, smile1);

SetConsoleTextAttribute(h, 14);

cout << (char)1;

}

SetConsoleTextAttribute(h, 4);

cout << (char)3;

SetConsoleCursorPosition(h, smile2);

SetConsoleTextAttribute(h, 13);

cout << (char)1;

if (code == 75)

{

smile1.X += 2;

SetConsoleCursorPosition(h, smile1);

SetConsoleTextAttribute(h, 14);

cout << (char)1;

}

}

void udletter()

{

SetConsoleCursorPosition(h, smile2);

cout << " ";

smile2.X--;

SetConsoleCursorPosition(h, smile2);

cout << " ";

smile2.X++;

SetConsoleTextAttribute(h, 4);

cout << (char)3;

if (code == 115)

{

smile2.Y--;

SetConsoleCursorPosition(h, smile2);

SetConsoleTextAttribute(h, 13);

cout << (char)1;

SetConsoleCursorPosition(h, smile1);

SetConsoleTextAttribute(h, 14);

cout << (char)1;

}

else if (code == 119)

{

SetConsoleCursorPosition(h, smile1);

SetConsoleTextAttribute(h, 14);

cout << (char)1;

smile2.Y++;

SetConsoleCursorPosition(h, smile2);

SetConsoleTextAttribute(h, 13);

cout << (char)1;

}

}

void udarrow()

{

SetConsoleCursorPosition(h, smile1);

cout << " ";

smile1.X--;

SetConsoleCursorPosition(h, smile1);

cout << " ";

smile1.X++;

SetConsoleTextAttribute(h, 4);

cout << (char)3;

if (code == 80)

{

smile1.Y--;

SetConsoleCursorPosition(h, smile1);

SetConsoleTextAttribute(h, 14);

cout << (char)1;

SetConsoleCursorPosition(h, smile2);

SetConsoleTextAttribute(h, 13);

cout << (char)1;

}

else if (code == 72)

{

smile1.Y++;

SetConsoleCursorPosition(h, smile2);

SetConsoleTextAttribute(h, 13);

cout << (char)1;

SetConsoleCursorPosition(h, smile1);

SetConsoleTextAttribute(h, 14);

cout << (char)1;

}

}

int main()

{

int height = 23;

int width = 79;

for (int y = 0; y < height; y++)

{

for (int x = 0; x < width; x++)

{

if (y == 0 || x == 0 || y == height - 1 || x == width - 1)

{

SetConsoleTextAttribute(h, 1);

cout << "\*";

//Sleep(20);

}

else cout << " ";

}

cout << "\n";

}

smile1.X = 2;

smile1.Y = 2;

SetConsoleCursorPosition(h, smile1);

SetConsoleTextAttribute(h, 14);

cout << (char) 1;

smile2.X = 5;

smile2.Y = 5;

SetConsoleCursorPosition(h, smile2);

SetConsoleTextAttribute(h, 13);

cout << (char)1;

while (true)

{

code = \_getch();

if (code == 0 || code == 224) code = \_getch();

////////////FIRST YELLOW SMILE/////////////

SetConsoleCursorPosition(h, smile1); // стирание смайлика в старой позиции

cout << " ";

if (code == 77 && smile1.X != width - 2) // right

{

smile1.X++;

}

else if (code == 75 && smile1.X != 1) // left

{

smile1.X--;

}

else if (code == 72 && smile1.Y != 1) // up

{

smile1.Y--;

}

else if (code == 80 && smile1.Y != height - 2) // down

{

smile1.Y++;

}

SetConsoleCursorPosition(h, smile1);

SetConsoleTextAttribute(h, 14);

cout << (char)1;

////////////SECOND PURPLE SMILE/////////////

SetConsoleCursorPosition(h, smile2); // стирание смайлика в старой позиции

cout << " ";

if (code == 100 && smile2.X != width - 2) // right D

{

smile2.X++;

}

else if (code == 97 && smile2.X != 1) // left A

{

smile2.X--;

}

else if (code == 119 && smile2.Y != 1) // up W

{

smile2.Y--;

}

else if (code == 115 && smile2.Y != height - 2) // down S

{

smile2.Y++;

}

SetConsoleCursorPosition(h, smile2);

SetConsoleTextAttribute(h, 13);

cout << (char)1;

/////////////////////////////////////////////////////////////////////////// точки соприкосновения

if (smile1.X == smile2.X - 1 && smile1.Y == smile2.Y && code == 97) // left letter

{

lrletter();

break;

}

else if (smile2.X == smile1.X - 1 && smile2.Y == smile1.Y && code == 75) // left arrow

{

lrarrow();

break;

}

else if (smile1.X == smile2.X + 1 && smile1.Y == smile2.Y && code == 100) // right letter

{

lrletter();

break;

}

else if (smile2.X == smile1.X + 1 && smile2.Y == smile1.Y && code == 77) // right arrow

{

lrarrow();

break;

}

else if (smile1.Y == smile2.Y + 1 && smile1.X == smile2.X && code == 115) // down letter

{

udletter();

break;

}

else if (smile2.Y == smile1.Y + 1 && smile2.X == smile1.X && code == 80) // down arrow

{

udarrow();

break;

}

else if (smile1.Y == smile2.Y - 1 && smile1.X == smile2.X && code == 119) // up letter

{

udletter();

break;

}

else if (smile2.Y == smile1.Y - 1 && smile2.X == smile1.X && code == 72) // up arrow

{

udarrow();

break;

}

else if (smile2.X == smile1.X && smile2.Y == smile1.Y)

{

SetConsoleCursorPosition(h, smile2);

cout << " ";

SetConsoleCursorPosition(h, smile1);

cout << " ";

SetConsoleTextAttribute(h, 4);

cout << (char)3 << " Happy end! " << (char)3 << "\n";

break;

}

}

}