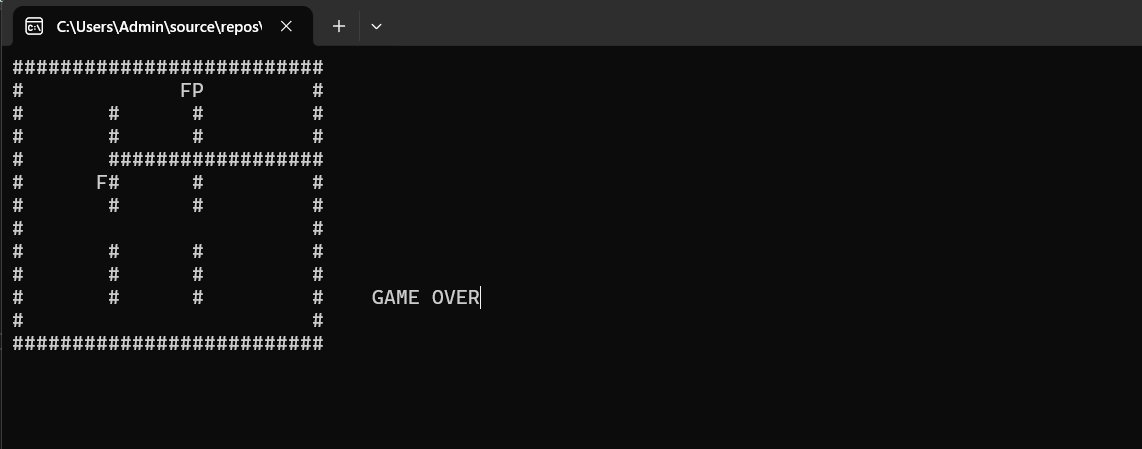
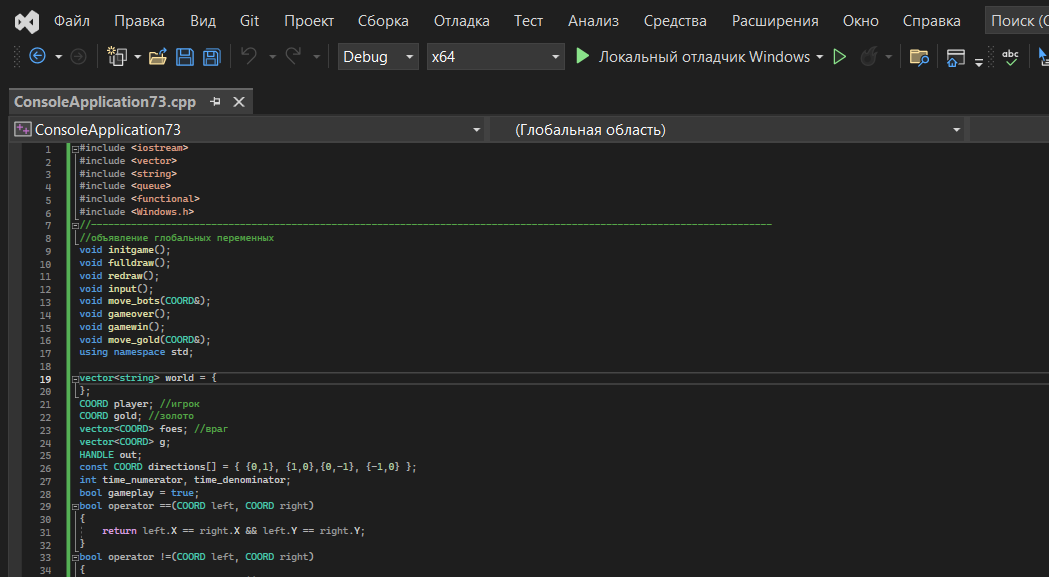
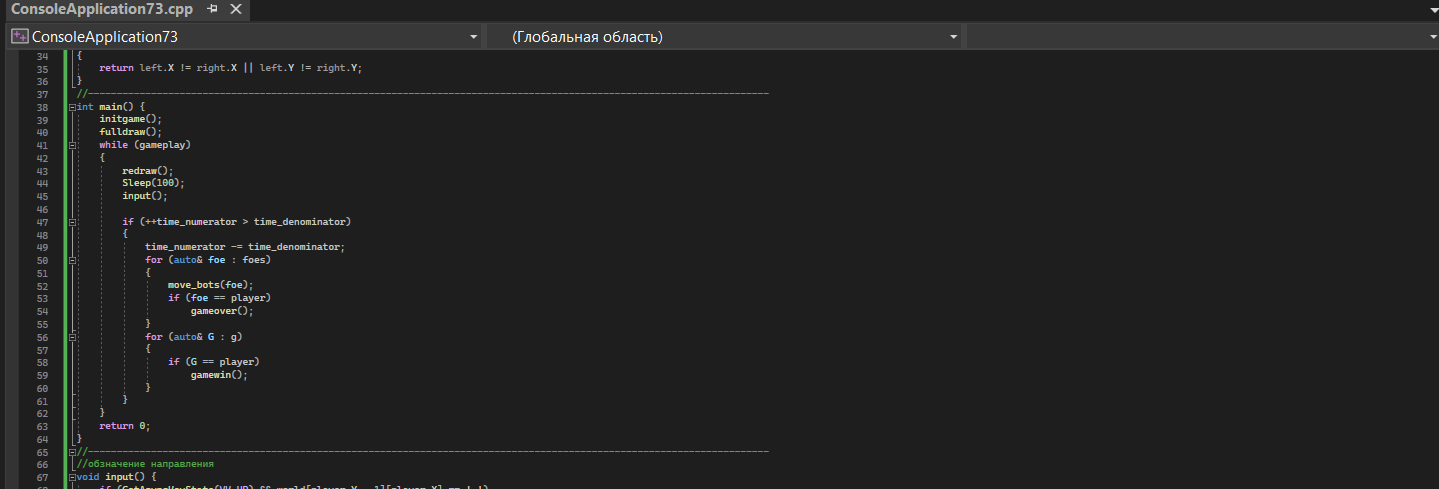
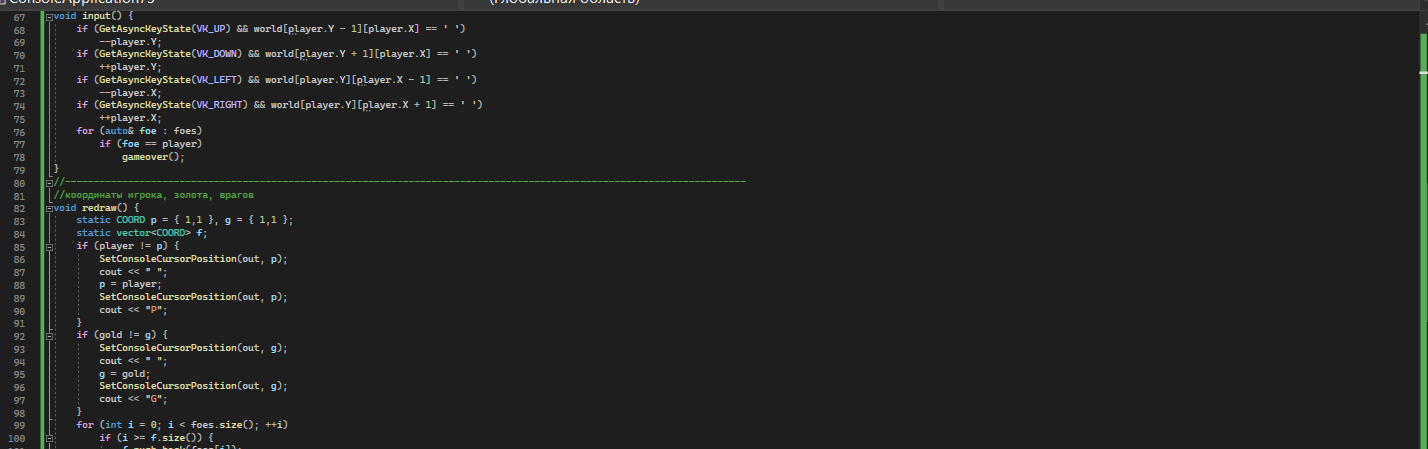
1. Результат игры

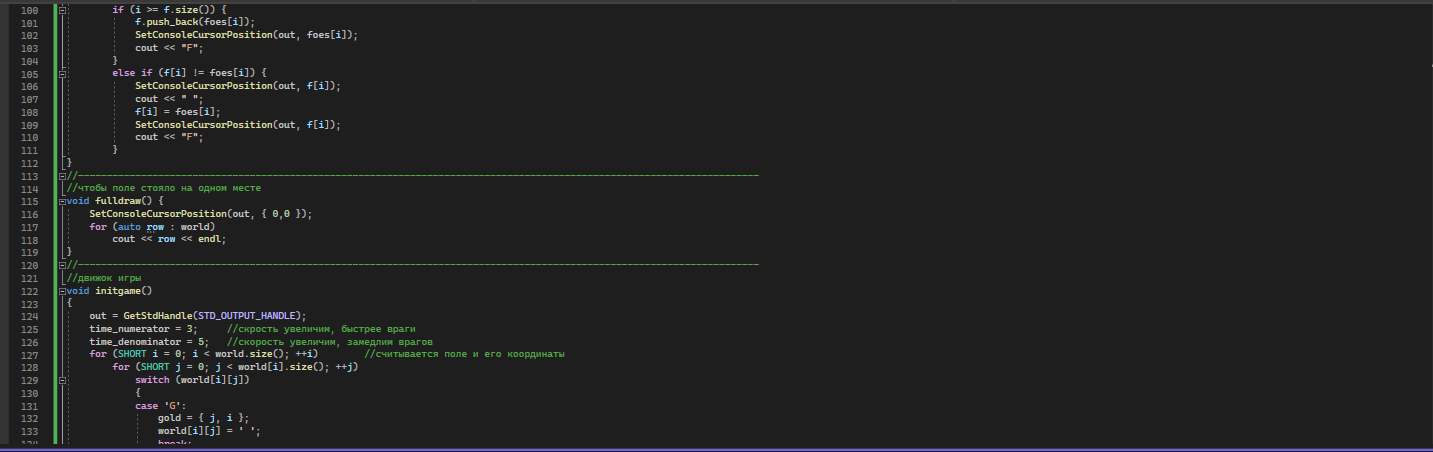


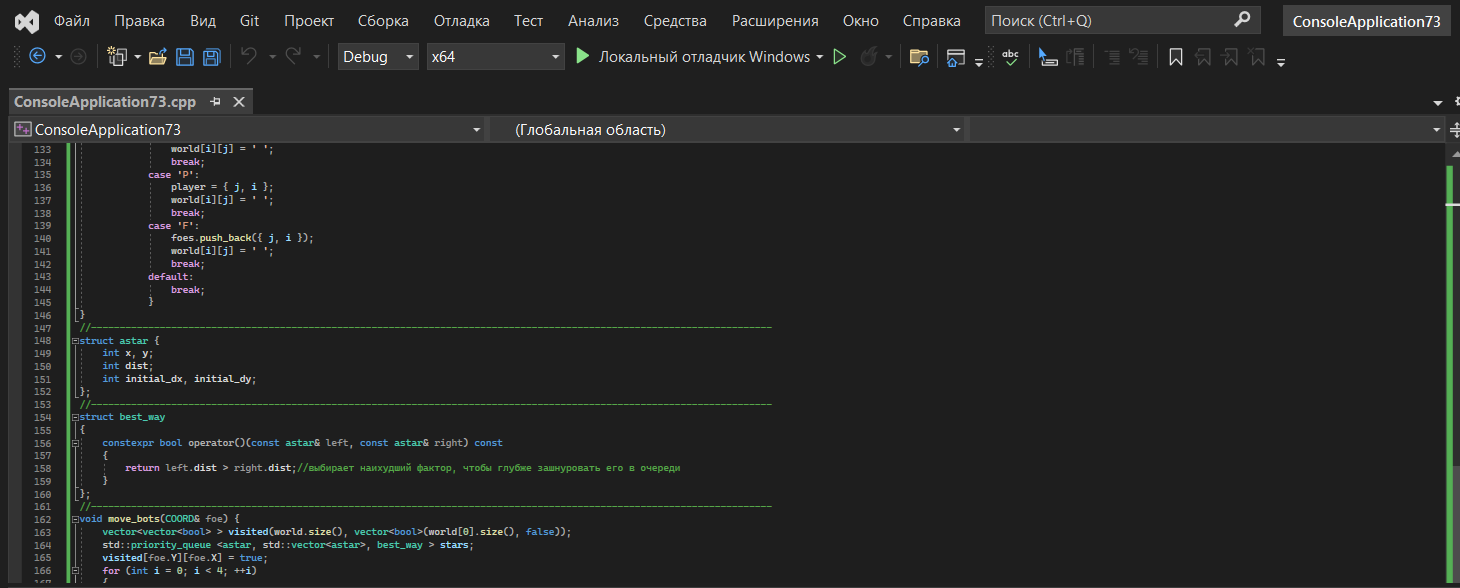
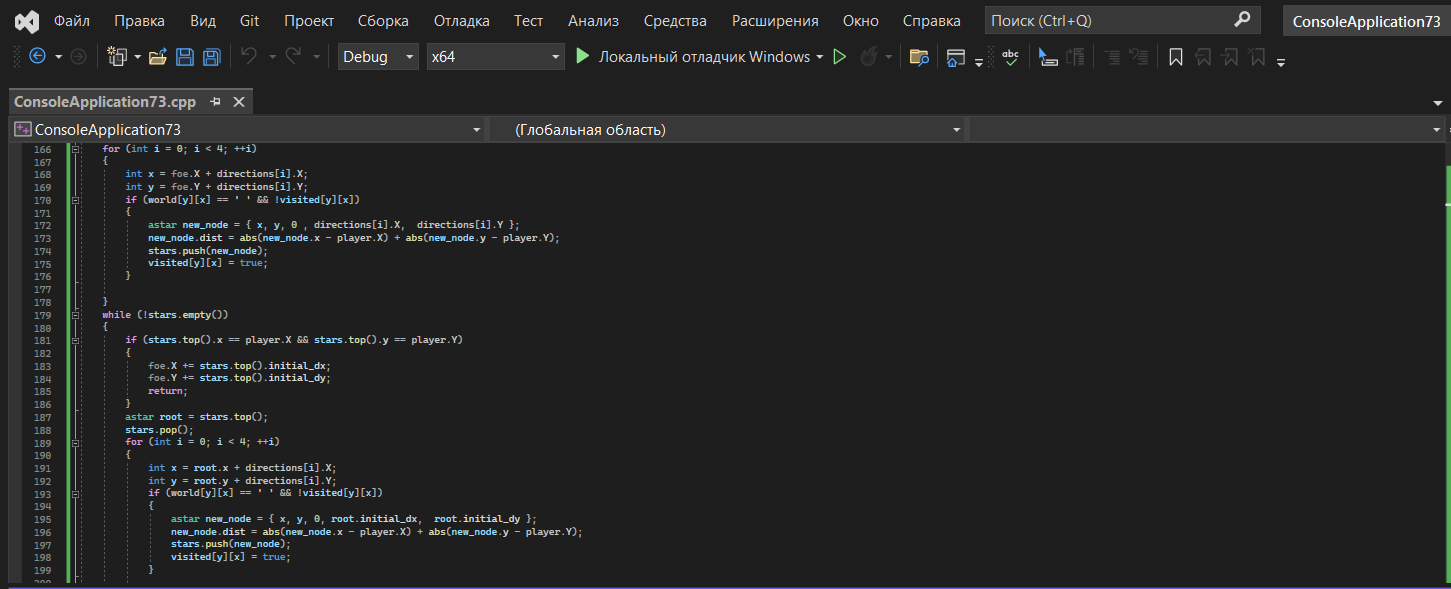
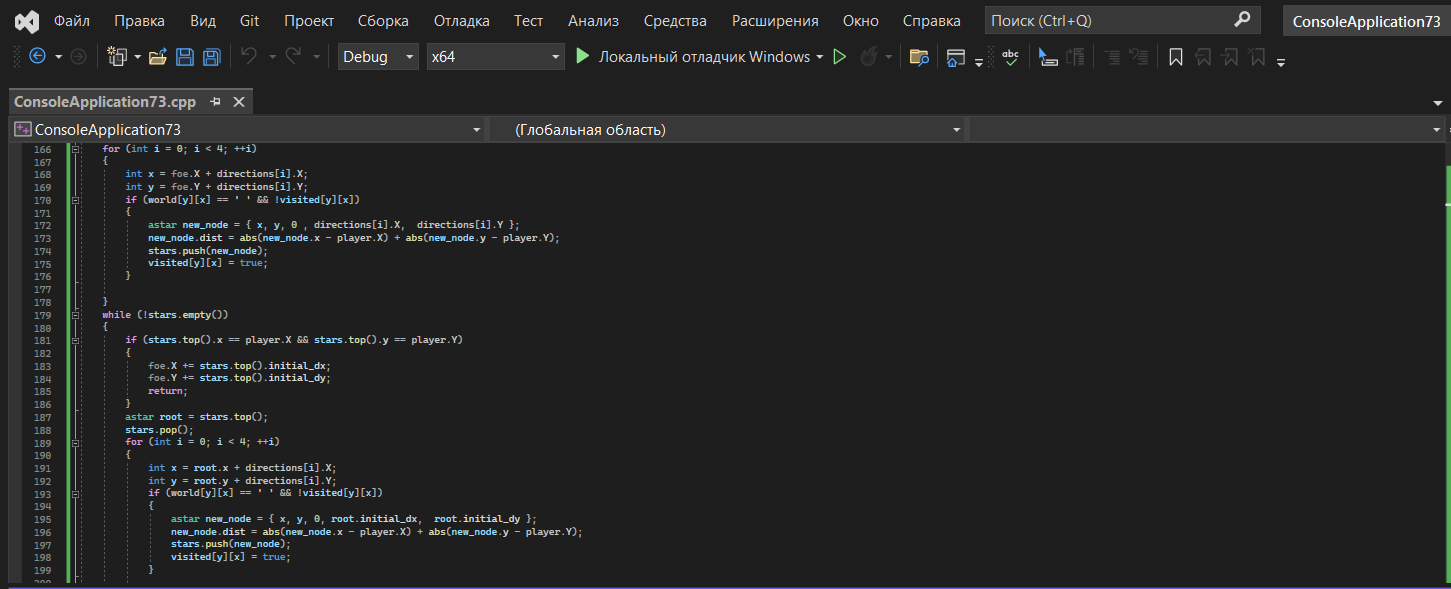
2. Код для самой игры

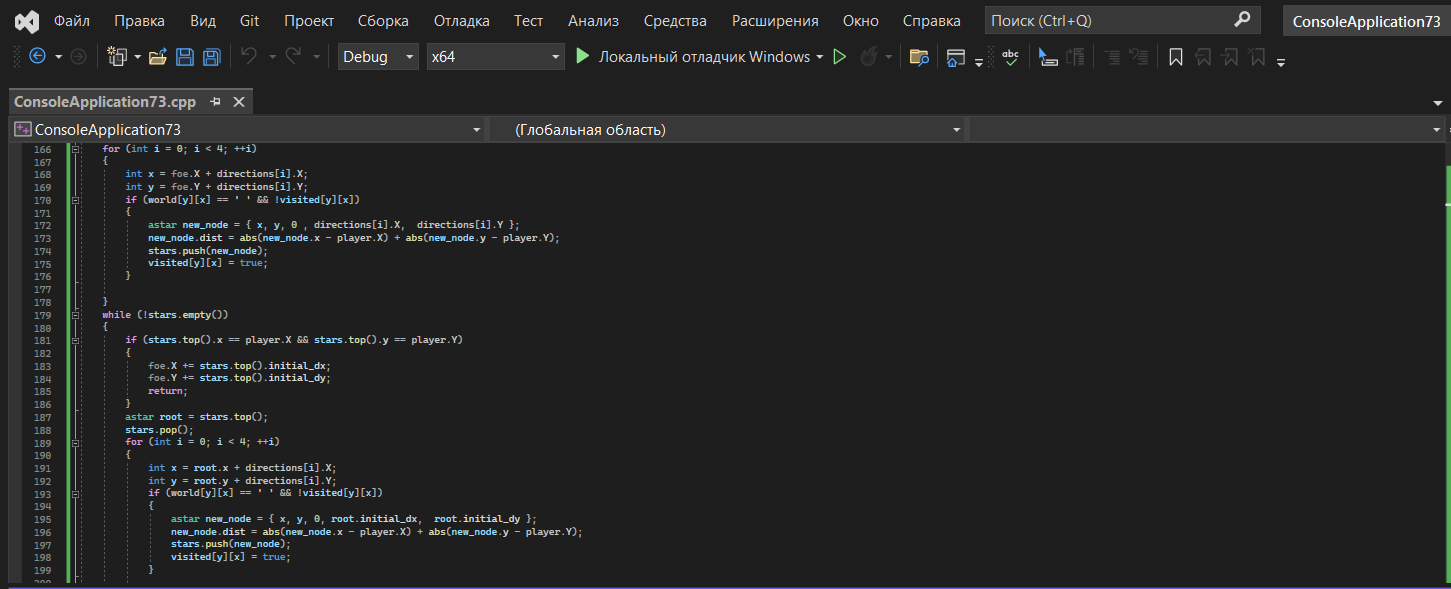












}

}

}

//-----------------------------------------------------------------------------------------------------------------------

void move\_win(COORD& G)

{

vector<vector<bool> > visited(world.size(), vector<bool>(world[0].size(), false));

std::priority\_queue <astar, std::vector<astar>, best\_way > stars;

visited[G.Y][G.X] = true;

for (int i = 0; i < 4; ++i)

{

int x = G.X + directions[i].X;

int y = G.Y + directions[i].Y;

if (world[y][x] == ' ' && !visited[y][x])

{

astar new\_node = { x, y, 0 , directions[i].X, directions[i].Y };

new\_node.dist = abs(new\_node.x - player.X) + abs(new\_node.y - player.Y);

stars.push(new\_node);

visited[y][x] = true;

}

}

while (!stars.empty())

{

if (stars.top().x == player.X && stars.top().y == player.Y)

{

G.X += stars.top().initial\_dx;

G.Y += stars.top().initial\_dy;

return;

}

astar root = stars.top();

stars.pop();

for (int i = 0; i < 4; ++i)

{

int x = root.x + directions[i].X;

int y = root.y + directions[i].Y;

if (world[y][x] == ' ' && !visited[y][x])

{

astar new\_node = { x, y, 0, root.initial\_dx, root.initial\_dy };

new\_node.dist = abs(new\_node.x - player.X) + abs(new\_node.y - player.Y);

stars.push(new\_node);

visited[y][x] = true;

}

}

}

}

//-----------------------------------------------------------------------------------------------------------------------

//проигрыш

void gameover() {

gameplay = false;

SetConsoleCursorPosition(out, { 30, 10 });

cout << "GAME OVER";

while (!GetAsyncKeyState(VK\_ESCAPE)) {} //определяет нажимание кнопки выхода

}

//-----------------------------------------------------------------------------------------------------------------------

//выигрыш

void gamewin() {

gameplay = false;

SetConsoleCursorPosition(out, { 30, 10 });

cout << "GAME Win";

while (!GetAsyncKeyState(VK\_ESCAPE)) {}

}