```
//Denzel P, Dariusz D
//gooseEscapeMain.cpp

#include <BearLibTerminal.h>
#include <cmath>
#include <iostream>  // Debugging tip:  You can still use cout to put debug messages on the
regular console window
using namespace std;
#include "gooseEscapeUtil.hpp"
#include "gooseEscapeActor.hpp"
#include "gooseEscapeConsole.hpp"
#include "gooseEscapeGamePlay.hpp"

//set up the console.   Don't modify this line!
Console out;

int main()
{
        //Set up the window.  Don't edit these two lines
    terminal_open();
        terminal_set(SETUP_MESSAGE);

/*
    The code below provides a skeleton of the game play.  You will need to
    write code for setting up the game board, and playing the game itself.
    You can modify the code given as needed.

    Call the functions that you have written in the game play file.
*/
/*
    The player and goose are initialized to the same locations
    (10,10) and (70,20) each time.  You likely want to change this somehow.
*/

    // make the player
        Actor player(PLAYER_CHAR, 5,5,100);

        // make the goose
        Actor monster(MONSTER_CHAR, 70,20,100);


    // declare the game board "map"
    int addBackMines[NUM_SCREEN_Y][NUM_SCREEN_X] = {0};
        int gameBoard[NUM_SCREEN_Y][NUM_SCREEN_X] = {0};
        int xSafe = 0;
        int ySafe = 0;
/*
    Initialize locations in the game board to have game features.  This
    would include anything that is static and doesn't move like a wall.  Hard
    coding them like you see below is a poor way to code this.  What if you
```

```
    have many features to add to the game board?  Should you use a loop?  Does
    it make sense to store this information in a file?  Should code be a
    function as well?
*/
        randomSetup(gameBoard, xSafe, ySafe);

        // Call the function to print the game board
        printWorld(gameBoard);

        // Printing the instructions in the console window
    out.writeLine("Escape the Goose! o:Wall,#:Landmine,F:Power-up for 10 moves");
        out.writeLine("Use the arrow keys to move");
        out.writeLine("If the goose catches you, you lose! Don't hit the landmines!");
        out.writeLine("Remember! Both you and the goose can't pass through walls!");
        //The functionality of letting goose jump walls is not implemented

/*
    This is the main game loop.  It continues to let the player give input
    as long as they do not press escape or close, they are not captured by
    the goose, and they didn't reach the win tile
*/
/*
    All key presses start with "TK_" then the character.  So "TK_A" is the "a"
    key being pressed.
*/
    int keyEntered = TK_A;  // can be any valid value that is not ESCAPE or CLOSE
    int powerCount = 0;

    while(keyEntered != TK_ESCAPE && keyEntered != TK_CLOSE
                    && !captured(player,monster)&& !safe(player,ySafe,xSafe) &&!isDead(player))
        {
            // get player key press, the function will wait until the user has pressed a key
            keyEntered = terminal_read();

        if (keyEntered != TK_ESCAPE && keyEntered != TK_CLOSE)
        {
                //move player based on tile type and next move
                powerCount = whichMove(keyEntered,player,gameBoard, powerCount, addBackMines);
            // move the player, you can modify this function
            terminal_refresh();
            //checks if player hit land mine
            injure(player, gameBoard);
            //check if player has died to landmine
                        isDead(player);

            //call the goose's chase function
            moveGoose(player, monster, gameBoard);
            terminal_refresh();

            // call other functions to do stuff?
```

```
        }
        }

    if (keyEntered != TK_CLOSE)
    {
        //once we're out of the loop, the game is over
        out.writeLine("Game has ended");

        // Output why:  did the goose get you?  Or did you win?
        if(safe(player,ySafe,xSafe)==true)
        {
                out.writeLine("Congratulations, You Win!");
                }
                else if(captured(player,monster)==true)
                {
                        out.writeLine("Unlucky, You Were Caught!");
                }
                else if(isDead(player)==true)
                {
                        out.writeLine("Unlucky, You Died!");
                }

        // Wait until user closes the window
        while (terminal_read() != TK_CLOSE);
    }
        //game is done, close it
    terminal_close();
}
```