浙江大學

# Object-Oriented Programming

(Project)

| | |
|---|---|
| Student Name | 刘佳润 |
| Title | The Castle |
| Mentor | 翁恺 |
| Student ID | 3180105640 |
| Class | 数媒 1801 |

## PART 1：The brief introduction of the project

The project implements a simple game of the castle. Just like many other similar games such as Dungeon Worrier or Undertale, the basic rule of the game is to control your character to move in a certain map. During the game, you may encounter some specific event such as fighting with monsters, collecting items and so on. When you reach a certain state, the game will give you the winning information.

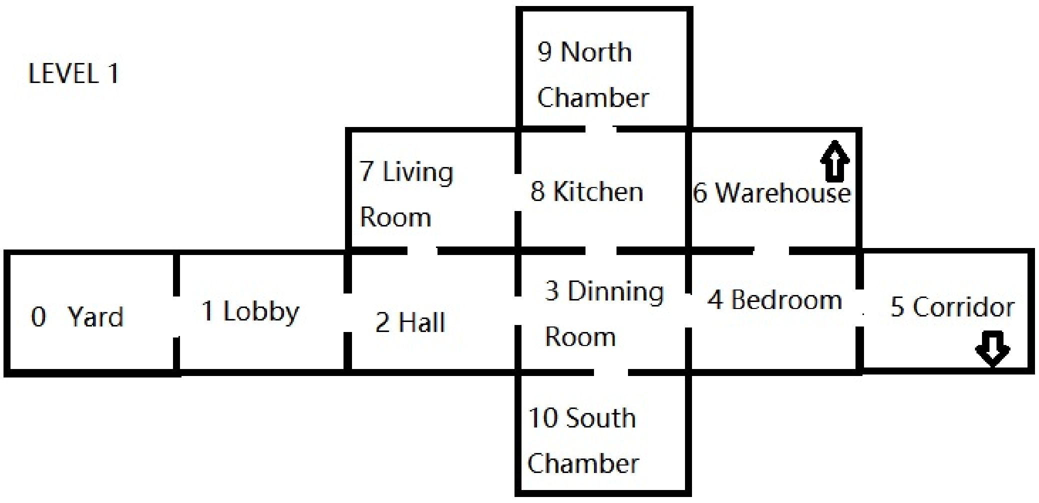
The detailed rule will be explained below.

## PART 2: The detailed information of the game

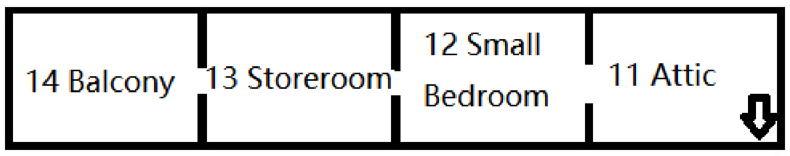
### The rule of the game:

You're a worrier. The princess of the kingdom is been caught by the monsters and locked up in a horrible castle. Your task is to explore the dangerous castle and rescue the princess. There're two monsters in the castle. The only way to defeat the monster is to kill them by the sword—which can be found in the castle. Moreover, the princess is locked up, and you must find the key to take her out. The key is also hidden in the castle. The winning condition is taking the princess out of the castle.

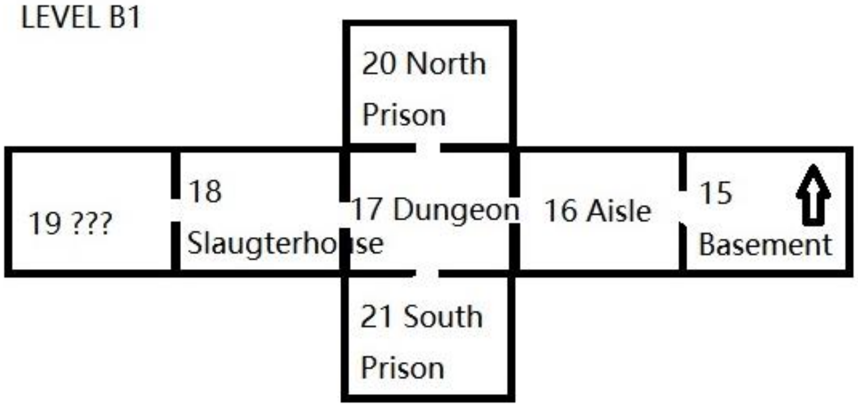
### The map of the castle:



LEVEL 2



LEVEL B1



- The princess is spawn in a random room of level B1.
- Two monsters are spawn respectively in range [5,11] and [11,21]. In other words, one in level 1, another one in level 2 or level B1.
- The sword is spawn in a random room of level 1.
- The key is spawn in a random room of level 2.

## <u>More detailed information of operation:</u>

- Moving control: input north/south/east/west/upstairs/downstairs to move. If your moving is legal, you will enter the next room. Otherwise, you are still in this room.
- Get help: input HELP to active the function *Help()* to get the hints.
- End the game: input BYE to leave the game immediately.
- Cheating: input the password—which is the birthday of the developer (20010308) to get the numbers of the room which contains items or princess or monster. However, this will lead to another ending of the game.

## PART 3: The general design of the project

### The File Structure:

-The Game of the Castle

- -Debug: The release file of the project
- -src: The source code of the project, several header files and cpp files included
- -THE CASTLE: The file constructed by Visual Studio 2017
- -THE CASTLE.exe: The executable file of the game
- -THE CASTLE.sln: The project solution
- -doc: The document file of the project
  - -Report.pdf: The project report
  - -Map: The map of the game, three jpeg files included
  - -README.txt: The instruction of the project

### The Project Structure:

main.cpp: The entrance of the game, including the main function

Game.h: Including the basic operation and checking functions of the game

Game.cpp: The body part of Game.h

Room.h: Declaration of the Room class

Room.cpp: The body part of Room.h

Princess.h: Declaration of the Princess class

Princess.cpp: The body part of Princess.h

Monster.h: Declaration of the Monster class

Monster.cpp: The body part of Monster.h

Sword.h: Declaration of the Sword class

Sword.cpp: The body part of Sword.h

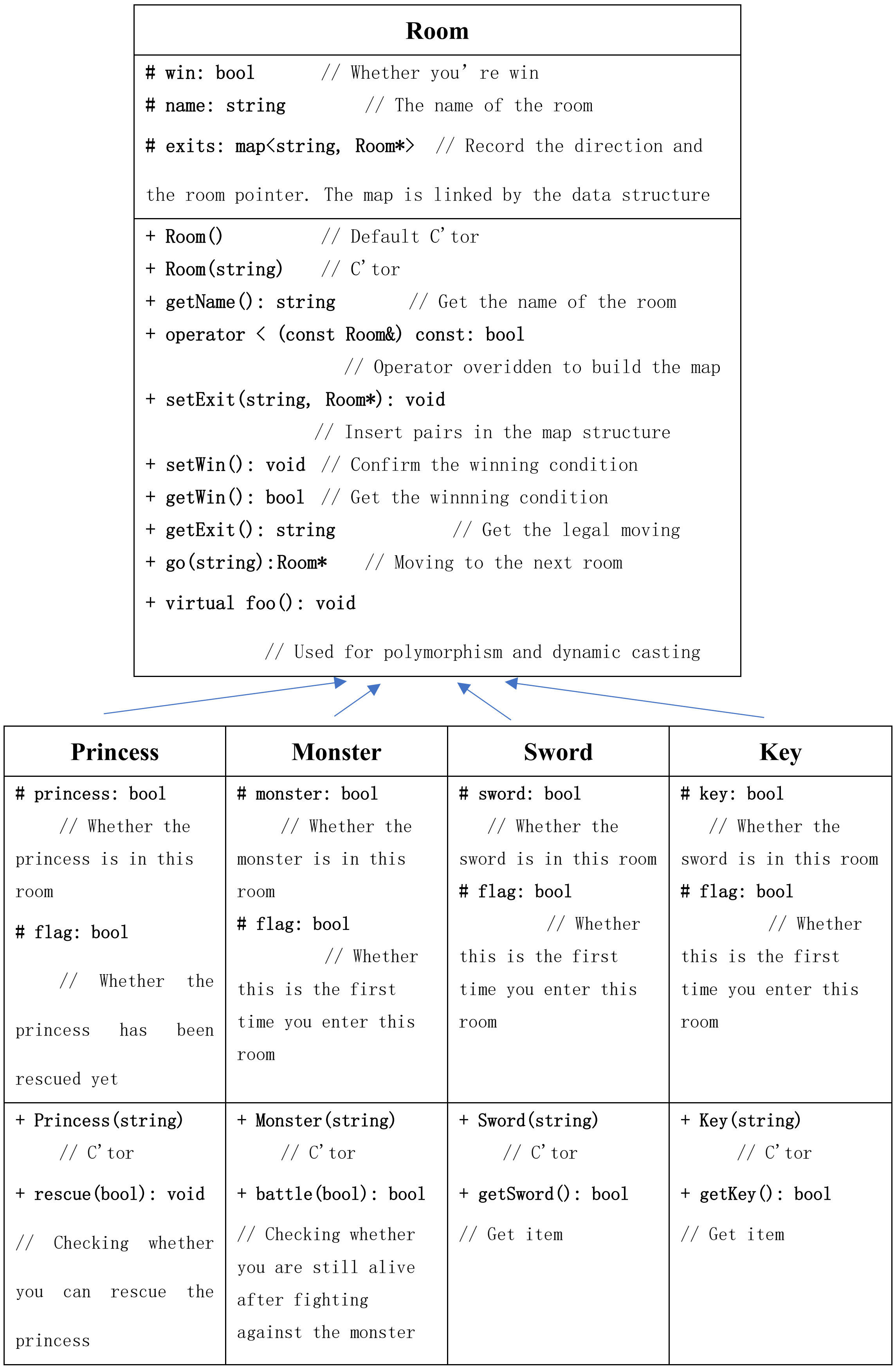
Key.h: Declaration of the Key class

Key.cpp: The body part of Key.h

## The Basic Data Structure:

| | |
|---|---|
| `struct Playerstate {`<br>`    bool sword = false;`<br>`    bool princess = false;`<br>`    bool key = false;`<br>`    bool cheat = false;`<br>`};` | structure Playerstate: Record the items that the player have and current situation of the player. |
| `map<int, string> Map;` | map Map: Store the room number and the name of the room. |
| `int no_prin, no_mon_1, no_mon_2, no_swo, no_key;` | global variable: The number of the special room. |
| `vector<Room*> r;` | vector r: During the initialization, store the room in order. |
| `map<string, Room*> exits;` | map exits: Record the direction and the room pointer. The rooms are linked by this data structure. |

The Class Design:



| **Room** |
|---|
| # win: bool // Whether you' re win<br># name: string // The name of the room<br># exits: map<string, Room*> // Record the direction and the room pointer. The map is linked by the data structure |
| + Room() // Default C' tor<br>+ Room(string) // C' tor<br>+ getName(): string // Get the name of the room<br>+ operator < (const Room&) const: bool<br>// Operator overidden to build the map<br>+ setExit(string, Room*): void<br>// Insert pairs in the map structure<br>+ setWin(): void // Confirm the winning condition<br>+ getWin(): bool // Get the winnning condition<br>+ getExit(): string // Get the legal moving<br>+ go(string):Room* // Moving to the next room<br>+ virtual foo(): void<br>// Used for polymorphism and dynamic casting |

| **Princess** | **Monster** | **Sword** | **Key** |
|---|---|---|---|
| # princess: bool<br>// Whether the princess is in this room<br># flag: bool<br>// Whether the princess has been rescued yet | # monster: bool<br>// Whether the monster is in this room<br># flag: bool<br>// Whether this is the first time you enter this room | # sword: bool<br>// Whether the sword is in this room<br># flag: bool<br>// Whether this is the first time you enter this room | # key: bool<br>// Whether the sword is in this room<br># flag: bool<br>// Whether this is the first time you enter this room |
| + Princess(string)<br>// C' tor<br>+ rescue(bool): void<br>// Checking whether you can rescue the princess | + Monster(string)<br>// C' tor<br>+ battle(bool): bool<br>// Checking whether you are still alive after fighting against the monster | + Sword(string)<br>// C' tor<br>+ getSword(): bool<br>// Get item | + Key(string)<br>// C' tor<br>+ getKey(): bool<br>// Get item |

The Function Interface Design:

| Function declaration | Function attribution | Algorithm description |
|---|---|---|
| *int main()* | main.cpp | Main function. |
| *void Initialization();* | Game.h | Intialize the map. Connect the name and the number of each room. Generate the random number of the special room. Set the exits of each room. |
| *void Help();* | Game.h | Print the help information. |
| *Room* Move(Room* cur, string dir);* | Game.h | According to the parameter *dir*, search the associated room of *cur*. Return the room pointer of the next room. |
| *bool Check(Room* p);* | Game.h | Checking the certain event that will happen when the player enter the room. |
| *void Cheat()* | Game.h | …… you cheater. |
| *Room::Room()* | Room.h (member function) | The default constructor of Room class. Create an empty room. |
| *Room::Room(string name)* | Room.h (member function) | The constructor of Room class. Create a room without exits whose name is the same of the |

| | | |
|---|---|---|
| | | parameter *name.* |
| *string Room::getName()* | Room.h (member function) | Return the name of the room. |
| *bool Room::operator < (const Room& cmp) const* | Room.h (member function) | Operator overidden to build the map |
| *void Room::setExit(string dir, Room* room)* | Room.h (member function) | Add an exit of the current room by inserting a pair of elements <*string, Room**> according to the parameter in the map *exits*. |
| *void Room::setWin()* | Room.h (member function) | Confirm the winning condition. Simply set the member variable *win* true. |
| *bool Room::getWin()* | Room.h (member function) | Get the winnning condition. |
| *string Room::getExit()* | Room.h (member function) | Get the legal moving possibilities by do a traversal of the member variable *exits.* |
| *Room* Room::go(string dir)* | Room.h (member function) | Moving to the next room. Find the room pointer which is associated with given parameter *dir*. Return the room pointer. If such a pointer doesn't exist, return the pointer of |

| | | |
|---|---|---|
| | | current room itself. |
| *void Room::foo()* | Room.h (member function) | An empty function. This function is designed for polymorphism and static casting of the pointer. |
| *Princess::Princess(string name)* | Princess.h (member function) | The constructor of Princess class. Create a room with a princess inside whose name is the same as the parameter *name.* |
| *void Princess::rescue(bool key)* | Princess.h (member function) | Checking whether you can rescue the princess. |
| *Monster::Monster(string name)* | Monster.h (member function) | The constructor of Monster class. Create a room with a monster inside whose name is the same as the parameter *name.* |
| *bool Monster::battle(bool sword)* | Monster.h (member function) | Checking whether you are still alive after fighting against the monster. |
| *Sword::Sword(string name)* | Sword.h (member function) | The constructor of Sword class. Create a room with a sword inside whose name is the same of the parameter *name.* |
| *bool Sword::getSword()* | Sword.h | Get the sword and take it |

| | | |
|---|---|---|
| | (member function) | by changing the player state. |
| *Key::Key(string name)* | Key.h (member function) | The constructor of Key class. Create a room with a key inside whose name is the same of the parameter *name*. |
| *bool Key::getKey()* | Key.h (member function) | Get the key and take it by changing the player state. |

## Some detailed description of the function:

The basic elements of the game—player and rooms, are implemented by a structure and several classes respectively. There're five different kinds of room: the simple room, the room with a monster, the room with a princess, the room with a sword and the room with a key. The latter four types are inherited from the original one.

As for the initialization part, the original map is static, but the special rooms are spawn randomly according to certain rules which have been mentioned above. The function *srand* and *rand* were used in this part. To simplify the code, a random number generator is defined by the sentence:

```
#define random(a,b) (rand()%(b-a)+a)
```

The exits of the room are added by inserting pairs into the map variable of the class:

```
void Room::setExit(string dir, Room* room)
{
    this->exits.insert(map<string, Room*>::value_type(dir, room));
}
```

The moving is implemented by the code below:

```
Room* Room::go(string dir)
{
    map<string, Room*>::iterator iter;
```

```
        iter = this->exits.find(dir);
        if (iter != this->exits.end())
            return iter->second;
        else
        {
            Sleep(500);
            cout << "---There's no door." << endl;
            return this;
        }
    }
    Room* Move(Room* cur, string dir)
    {
        Room *next = new Room();
        next = cur->go(dir);

        Sleep(500);
        cout << "*You're at the " << next->getName() << ".*" << endl;
        Sleep(500);
        cout << "*You can go " << next->getExit() << "*" << endl;

        return next;
    }
```

As for the checking part, the function made a judge by checking the parameter which refers to the state of the player and do the operation correspondingly. The method is very simple, so please check the codes if you want to know more details.

A very interesting method I've applied in the project is the static casting of the pointer to let a pointer of the basic class point to the derived class. A sample code is shown below:

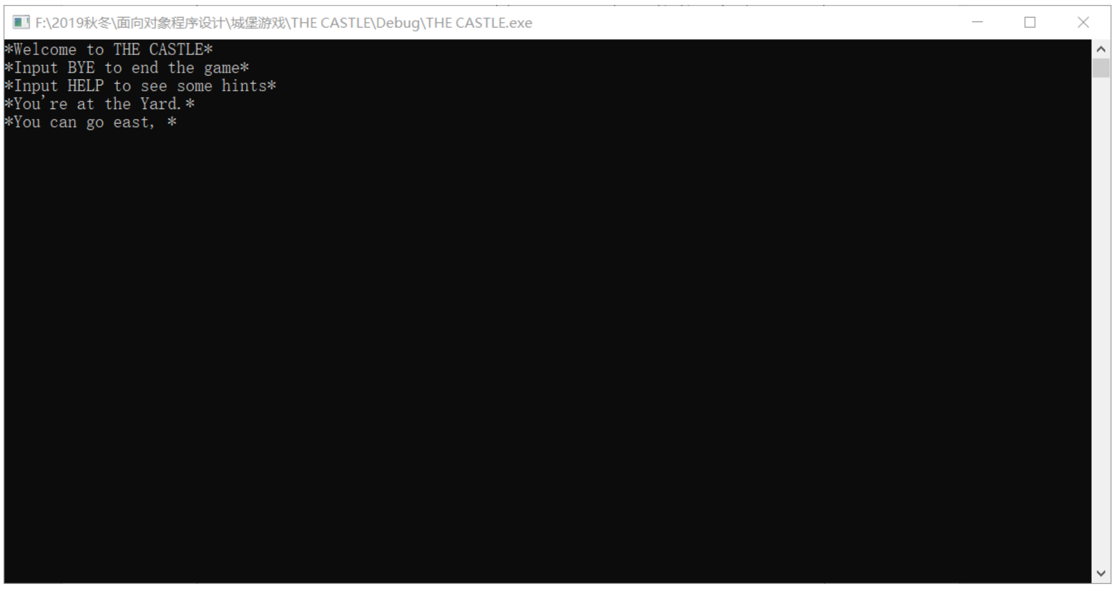
```
    if (p->getName() == Map[no_prin])
    {
        Princess* q = static_cast<Princess*>(p);
        if (q != NULL)
        {
            q->rescue(Player.key);
            Player.princess = true;
        }
    }
```

Notice that p is a pointer of the Room class. In order to call the function of its derived class Princess, we have to create a pointer of the Princess class. Since I've created an empty virtual function, I can use the static casting of the pointer to do the type conversion.

## PART 4: The deployment and the operation result of the game

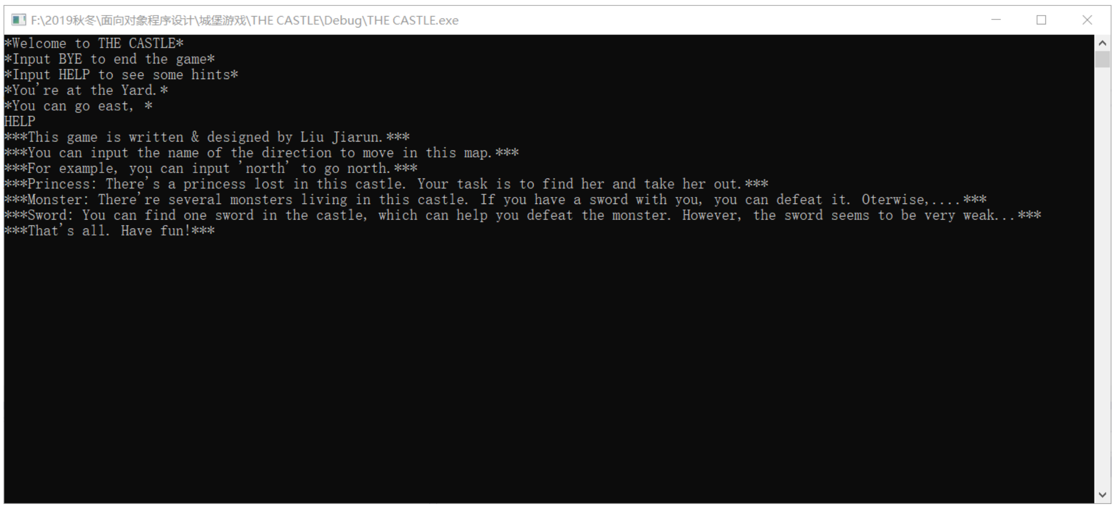
The compiler environment of the project is Visual Studio 2017. If your compiler is also VS2017, you can open the solution file *The Castle.sln* to compile and test the code. Otherwise, please copy all the header files and cpp files in src file and add them to your own project. Then you can test it on your own compiler.

The executable file has already been created. If you want to play the game in your console, you can simply run *The Castle.exe* file and enjoy yourself.


F:\2019秋冬\面向对象程序设计\城堡游戏\THE CASTLE\Debug\THE CASTLE.exe

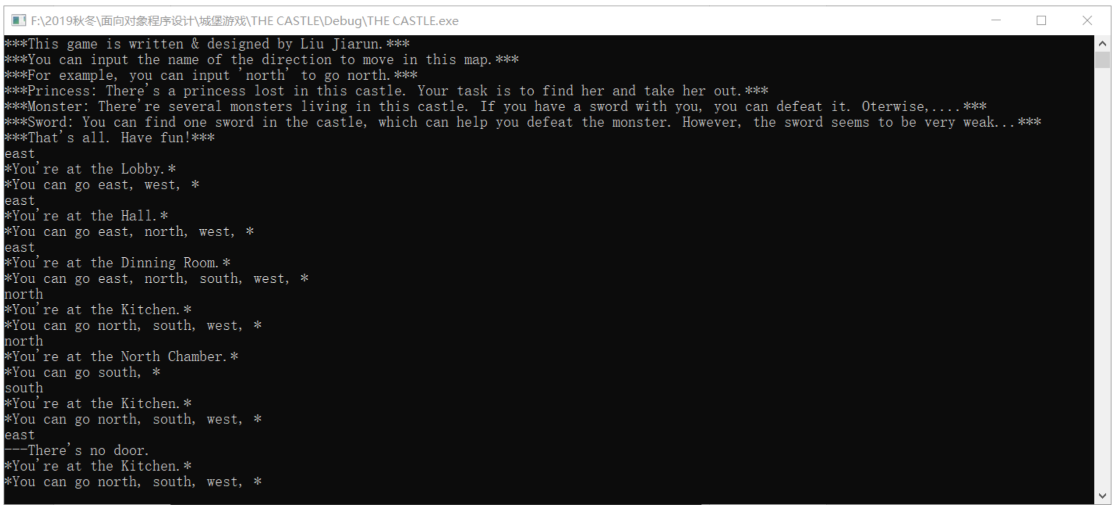
```
*Welcome to THE CASTLE*
*Input BYE to end the game*
*Input HELP to see some hints*
*You're at the Yard.*
*You can go east, *
```

The beginning of the game


F:\2019秋冬\面向对象程序设计\城堡游戏\THE CASTLE\Debug\THE CASTLE.exe

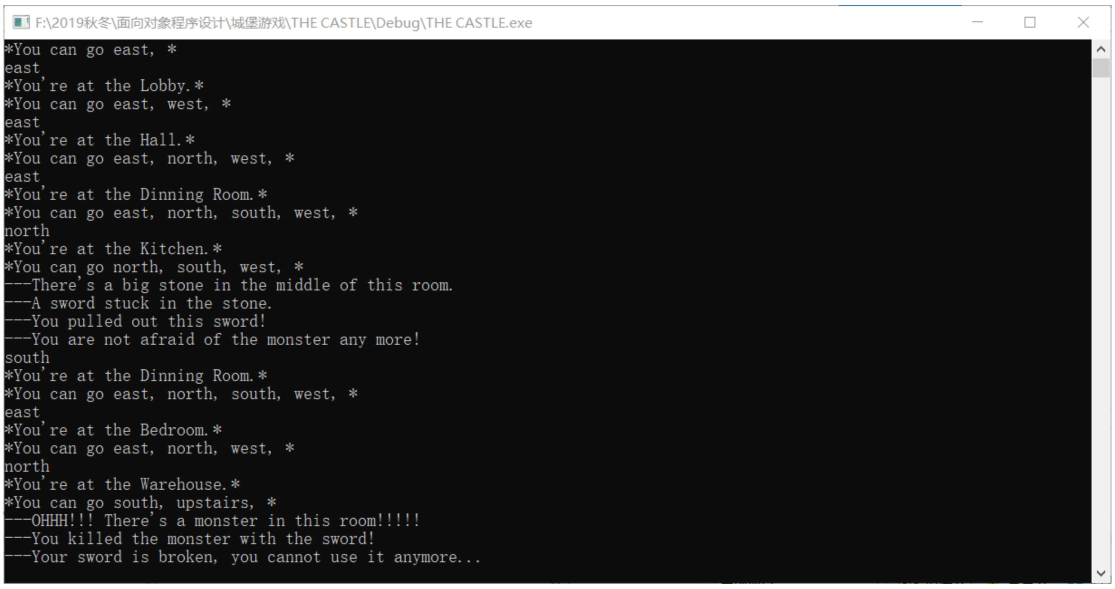
```
*Welcome to THE CASTLE*
*Input BYE to end the game*
*Input HELP to see some hints*
*You're at the Yard.*
*You can go east, *
HELP
***This game is written & designed by Liu Jiarun.***
***You can input the name of the direction to move in this map.***
***For example, you can input 'north' to go north.***
***Princess: There's a princess lost in this castle. Your task is to find her and take her out.***
***Monster: There're several monsters living in this castle. If you have a sword with you, you can defeat it. Oterwise,....***
***Sword: You can find one sword in the castle, which can help you defeat the monster. However, the sword seems to be very weak...***
***That's all. Have fun!***
```

The help menu of the game


F:\2019秋冬\面向对象程序设计\城堡游戏\THE CASTLE\Debug\THE CASTLE.exe

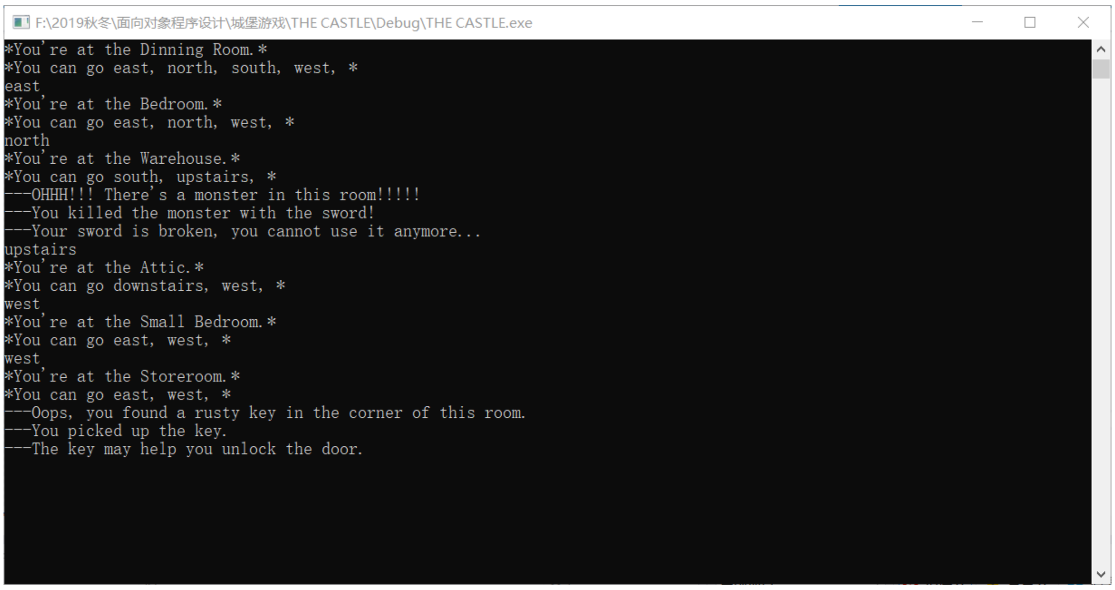
```
***This game is written & designed by Liu Jiarun.***
***You can input the name of the direction to move in this map.***
***For example, you can input 'north' to go north.***
***Princess: There's a princess lost in this castle. Your task is to find her and take her out.***
***Monster: There're several monsters living in this castle. If you have a sword with you, you can defeat it. Oterwise,....***
***Sword: You can find one sword in the castle, which can help you defeat the monster. However, the sword seems to be very weak...***
***That's all. Have fun!***
east
*You're at the Lobby.*
*You can go east, west, *
east
*You're at the Hall.*
*You can go east, north, west, *
east
*You're at the Dinning Room.*
*You can go east, north, south, west, *
north
*You're at the Kitchen.*
*You can go north, south, west, *
north
*You're at the North Chamber.*
*You can go south, *
south
*You're at the Kitchen.*
*You can go north, south, west, *
east
---There's no door.
*You're at the Kitchen.*
*You can go north, south, west, *
```

The basic operation and results


F:\2019秋冬\面向对象程序设计\城堡游戏\THE CASTLE\Debug\THE CASTLE.exe

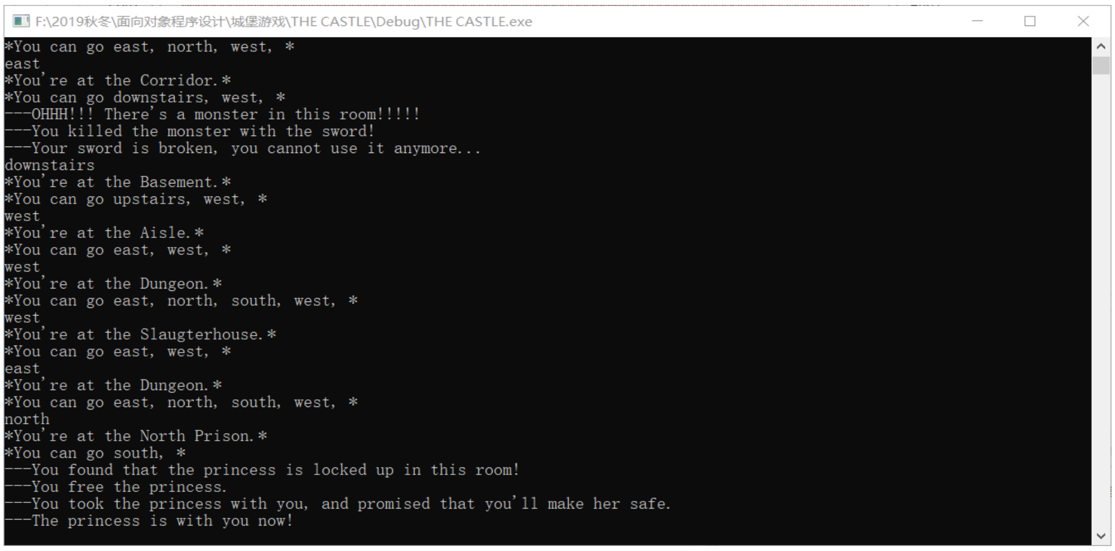
```
*You can go east, *
east
*You're at the Lobby.*
*You can go east, west, *
east
*You're at the Hall.*
*You can go east, north, west, *
east
*You're at the Dinning Room.*
*You can go east, north, south, west, *
north
*You're at the Kitchen.*
*You can go north, south, west, *
---There's a big stone in the middle of this room.
---A sword stuck in the stone.
---You pulled out this sword!
---You are not afraid of the monster any more!
south
*You're at the Dinning Room.*
*You can go east, north, south, west, *
east
*You're at the Bedroom.*
*You can go east, north, west, *
north
*You're at the Warehouse.*
*You can go south, upstairs, *
---OHHH!!! There's a monster in this room!!!!!
---You killed the monster with the sword!
---Your sword is broken, you cannot use it anymore...
```

Found a sword and defeated a monster


F:\2019秋冬\面向对象程序设计\城堡游戏\THE CASTLE\Debug\THE CASTLE.exe

```
*You're at the Dinning Room.*
*You can go east, north, south, west, *
east
*You're at the Bedroom.*
*You can go east, north, west, *
north
*You're at the Warehouse.*
*You can go south, upstairs, *
---OHHH!!! There's a monster in this room!!!!!
---You killed the monster with the sword!
---Your sword is broken, you cannot use it anymore...
upstairs
*You're at the Attic.*
*You can go downstairs, west, *
west
*You're at the Small Bedroom.*
*You can go east, west, *
west
*You're at the Storeroom.*
*You can go east, west, *
---Oops, you found a rusty key in the corner of this room.
---You picked up the key.
---The key may help you unlock the door.
```

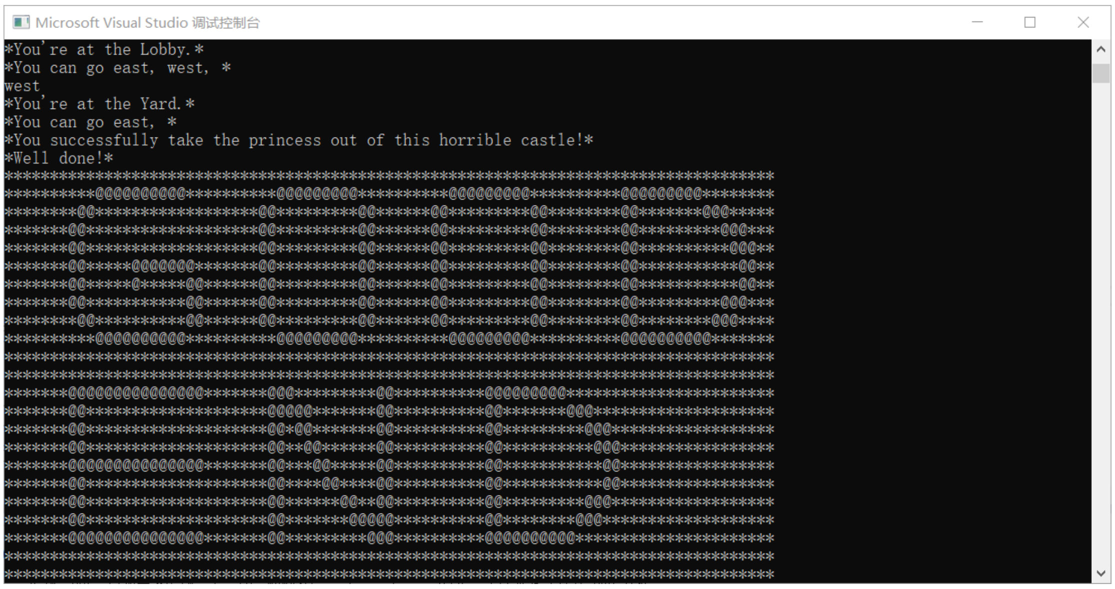
Found the key to the room of princess



F:\2019秋冬\面向对象程序设计\城堡游戏\THE CASTLE\Debug\THE CASTLE.exe

```
*You can go east, north, west, *
east
*You're at the Corridor.*
*You can go downstairs, west, *
---OHHH!!! There's a monster in this room!!!!!
---You killed the monster with the sword!
---Your sword is broken, you cannot use it anymore...
downstairs
*You're at the Basement.*
*You can go upstairs, west, *
west
*You're at the Aisle.*
*You can go east, west, *
west
*You're at the Dungeon.*
*You can go east, north, south, west, *
west
*You're at the Slaugterhouse.*
*You can go east, west, *
east
*You're at the Dungeon.*
*You can go east, north, south, west, *
north
*You're at the North Prison.*
*You can go south, *
---You found that the princess is locked up in this room!
---You free the princess.
---You took the princess with you, and promised that you'll make her safe.
---The princess is with you now!
```

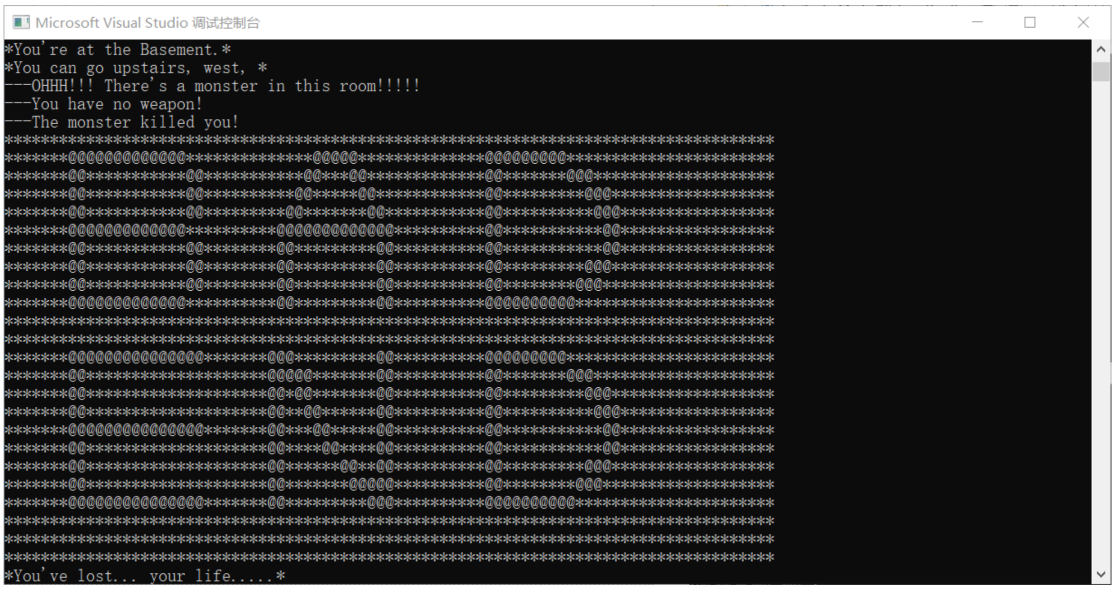
The princess was rescued



Microsoft Visual Studio 调试控制台

```
*You're at the Lobby.*
*You can go east, west, *
west
*You're at the Yard.*
*You can go east, *
*You successfully take the princess out of this horrible castle!*
*Well done!*
*************************************************************************************
**********@@@@@@@@@@**********@@@@@@@@@**********@@@@@@@@@**********@@@@@@@@@********
********@@******************@@*********@@******@@*********@@********@@*******@@@*****
*******@@*******************@@*********@@******@@*********@@********@@*********@@@***
*******@@*******************@@*********@@******@@*********@@********@@**********@@@**
*******@@*****@@@@@@@*******@@*********@@******@@*********@@********@@***********@@**
*******@@*****@*****@@******@@*********@@******@@*********@@********@@***********@@**
*******@@***********@@******@@*********@@******@@*********@@********@@*********@@@***
********@@**********@@******@@*********@@******@@*********@@********@@********@@@****
**********@@@@@@@@@@**********@@@@@@@@@**********@@@@@@@@@**********@@@@@@@@@@*******
*************************************************************************************
*************************************************************************************
*******@@@@@@@@@@@@@@@*******@@@*********@@**********@@@@@@@@@***********************
*******@@********************@@@@@*******@@**********@@*******@@@********************
*******@@********************@@*@@*******@@**********@@*********@@@******************
*******@@********************@@**@@******@@**********@@**********@@@*****************
*******@@@@@@@@@@@@@@@*******@@***@@*****@@**********@@***********@@*****************
*******@@********************@@****@@****@@**********@@***********@@*****************
*******@@********************@@******@@**@@**********@@*********@@@******************
*******@@********************@@*******@@@@@**********@@********@@@*******************
*******@@@@@@@@@@@@@@@*******@@*********@@@**********@@@@@@@@@@**********************
*************************************************************************************
*************************************************************************************
```

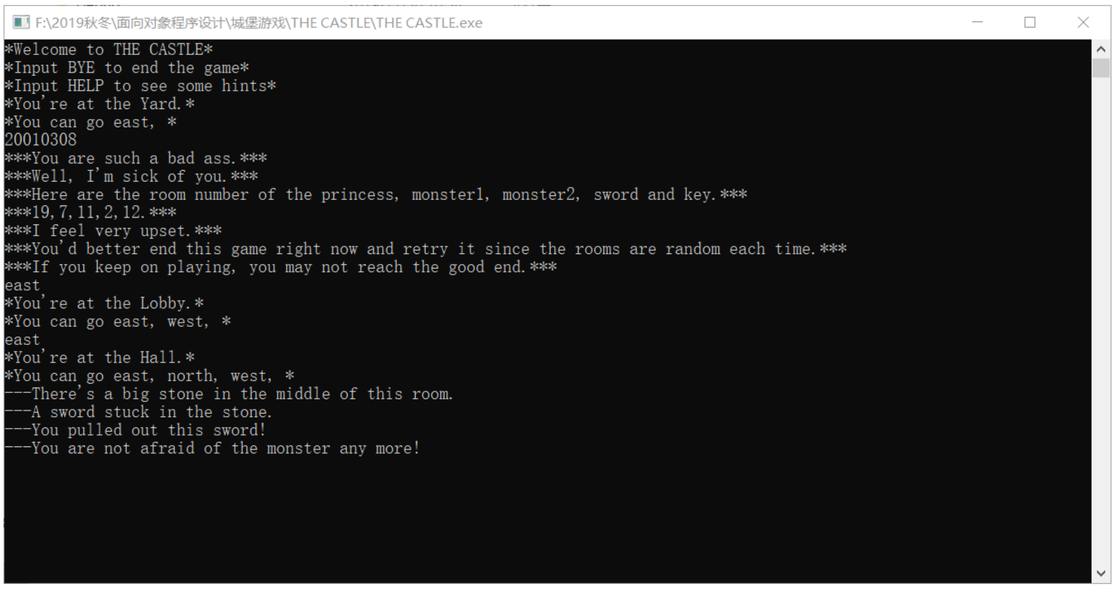
Good End: You took the princess out of the castle



Microsoft Visual Studio 调试控制台

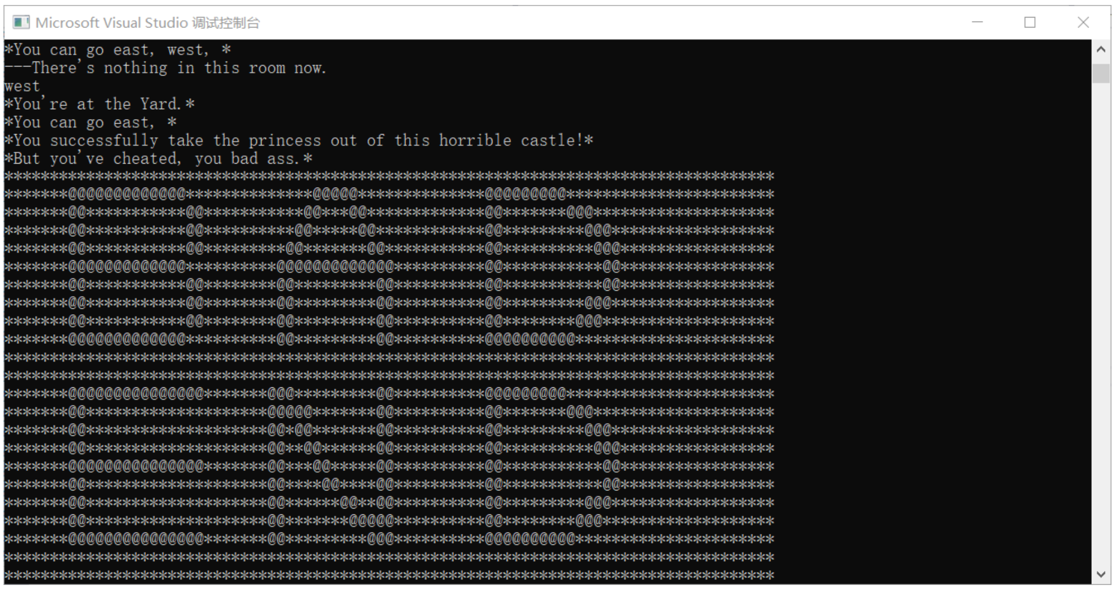
```
*You're at the Basement.*
*You can go upstairs, west, *
---OHHH!!! There's a monster in this room!!!!!
---You have no weapon!
---The monster killed you!
*************************************************************************************
*******@@@@@@@@@@@@@**************@@@@@**************@@@@@@@@@***********************
*******@@***********@@***********@@***@@*************@@*******@@@********************
*******@@***********@@**********@@*****@@************@@*********@@@******************
*******@@***********@@*********@@*******@@***********@@**********@@@*****************
*******@@@@@@@@@@@@@**********@@@@@@@@@@@@@**********@@***********@@*****************
*******@@***********@@********@@*********@@**********@@***********@@*****************
*******@@***********@@********@@*********@@**********@@*********@@@******************
*******@@***********@@********@@*********@@**********@@********@@@*******************
*******@@@@@@@@@@@@@**********@@*********@@**********@@@@@@@@@@**********************
*************************************************************************************
*************************************************************************************
*******@@@@@@@@@@@@@@@*******@@@*********@@**********@@@@@@@@@***********************
*******@@********************@@@@@*******@@**********@@*******@@@********************
*******@@********************@@*@@*******@@**********@@*********@@@******************
*******@@********************@@**@@******@@**********@@**********@@@*****************
*******@@@@@@@@@@@@@@@*******@@***@@*****@@**********@@***********@@*****************
*******@@********************@@****@@****@@**********@@***********@@*****************
*******@@********************@@******@@**@@**********@@*********@@@******************
*******@@********************@@*******@@@@@**********@@********@@@*******************
*******@@@@@@@@@@@@@@@*******@@*********@@@**********@@@@@@@@@@**********************
*************************************************************************************
*************************************************************************************
*************************************************************************************
*You've lost... your life.....*
```

Bad End 1: You are killed by the monster



```
F:\2019秋冬\面向对象程序设计\城堡游戏\THE CASTLE\THE CASTLE.exe
*Welcome to THE CASTLE*
*Input BYE to end the game*
*Input HELP to see some hints*
*You're at the Yard.*
*You can go east, *
20010308
***You are such a bad ass.***
***Well, I'm sick of you.***
***Here are the room number of the princess, monster1, monster2, sword and key.***
***19,7,11,2,12.***
***I feel very upset.***
***You'd better end this game right now and retry it since the rooms are random each time.***
***If you keep on playing, you may not reach the good end.***
east
*You're at the Lobby.*
*You can go east, west, *
east
*You're at the Hall.*
*You can go east, north, west, *
---There's a big stone in the middle of this room.
---A sword stuck in the stone.
---You pulled out this sword!
---You are not afraid of the monster any more!
```

Input the cheating code



```
Microsoft Visual Studio 调试控制台
*You can go east, west, *
---There's nothing in this room now.
west
*You're at the Yard.*
*You can go east, *
*You successfully take the princess out of this horrible castle!*
*But you've cheated, you bad ass.*
*************************************************************************************
*******@@@@@@@@@@@@@**************@@@@@**************@@@@@@@@@***********************
*******@@***********@@***********@@***@@*************@@*******@@@********************
*******@@***********@@**********@@*****@@************@@*********@@@******************
*******@@***********@@*********@@*******@@***********@@**********@@@*****************
*******@@@@@@@@@@@@@**********@@@@@@@@@@@@@**********@@***********@@*****************
*******@@***********@@********@@*********@@**********@@***********@@*****************
*******@@***********@@********@@*********@@**********@@*********@@@******************
*******@@***********@@********@@*********@@**********@@********@@@*******************
*******@@@@@@@@@@@@@**********@@*********@@**********@@@@@@@@@@**********************
*************************************************************************************
*************************************************************************************
*******@@@@@@@@@@@@@@@*******@@@*********@@**********@@@@@@@@@***********************
*******@@********************@@@@@*******@@**********@@*******@@@********************
*******@@********************@@*@@*******@@**********@@*********@@@******************
*******@@********************@@**@@******@@**********@@**********@@@*****************
*******@@@@@@@@@@@@@@@*******@@***@@*****@@**********@@***********@@*****************
*******@@********************@@****@@****@@**********@@***********@@*****************
*******@@********************@@******@@**@@**********@@*********@@@******************
*******@@********************@@*******@@@@@**********@@********@@@*******************
*******@@@@@@@@@@@@@@@*******@@*********@@@**********@@@@@@@@@@**********************
*************************************************************************************
*************************************************************************************
```

Bad End 2: You' ve cheated

## PART 5: Conclusion & Experience sharing

To be honest, I've learned a lot from this project. First of all, thanks to the design of the room type, I've tried a lot of methods to build the classes and implement the class inheritance. What's more, to construct a map is also a hard task for me. In this part, I've learned a lot about the STL and the usage of map. I think that I've got the point that STL is a great invention of C++. However, the memory allocation in C++ truly made me feel sick. I've struggled for a long time testing the legality of the pointer and temporary variable. Finally, I’ve got to know the importance of mastering the usage of “new” to create the pointer of a class.

Furthermore, I don’t want to design more complicated function for this simple game. Frankly speaking, I feel sick playing such a game in windows console. But I do think of several fresh features and implement them in the game, such as the sword and the key. Moreover, I’ve made some progress in spawning the special room. I’ve given the restriction of the random number generator to raise the playability of the game.

If I’m going to make a game called The Castle 2.0, I may create more types of monsters and upgrade the battle system. The player(worrier) will be described as a class, too. The generating of the map will be upgraded either. There’ll be a data base to store the possibility of the maps. More items may added to the game.

Well, I hope I won’t do that in the future.

## Reference books:

《Introduction to Programming with C++ (Third Edition)》 Y. Daniel Liang

《Thinking in C++ (Second Edition)》 Bruce Eckel

《C Programming FAQs: Frequently Asked Questions》 Steve Summit