

作者：董佳昕

一、背景

实现一款名为**Adventure**的CLI游戏。玩家必须探索一个城堡，其中包含多个关卡和多个房间。玩家的任务是找到囚禁公主的房间，并带她离开！

房间有很多种类型，每种类型都有不同的出口。注意，在其中一个房间里有一个怪物，它的确切位置是未知的。但一旦玩家遇到怪物，游戏就结束了。

程序总是显示玩家进入的房间的信息：房间的名称，有多少个出口，以及所有出口的名称。例如，当游戏开始时，玩家在城堡的大厅中，程序输出：

```
1 | welcome to the lobby. There are 3 exits: east, west and up.  
2 | Enter your command:
```

然后玩家可以输入go命令，然后输入他/她想要通过的出口名称，例如：

```
1 | go east
```

因此，玩家将进入东边的房间。该程序提供了该房间的信息，就像上面显示的大厅。这个过程重复。

在这个过程中，如果玩家进入一个有怪物的房间，程序就会显示一条信息，游戏就结束了。同样地，当玩家进入公主所在的密室时，程序便会显示角色与公主之间的对话。在那之后，她准备带着这个角色离开。然后玩家必须找到自己的出路。离开城堡的唯一途径是通过大厅。

为了简化实现，所有打印的消息和用户输入都以英文显示。

二、实现思路

2.1 地图的存储

通过一个二维int数组map，记录当前游戏的地图

每一个位置有5种不同的取值：

```
1 | //房间的种类  
2 | const int cannot_go = 0;    //该房间不能进入  
3 | const int lobby = 1;        //出生点(大厅)  
4 | const int princessRoom = 2; //房间有公主  
5 | const int monsterRoom = 3;  //房间有怪兽  
6 | const int emptyRoom = 4;    //空房间
```

2.2 地图的创建

根据输入的地图大小n*m，依次创建出：

(1)玩家的初始位置

(2)公主的初始位置

(3)玩家==>公主之间的路径（为了保证存在答案）

(4)怪兽的位置


```

the map is as bellow,size=10*10
H:the position of the player
#:there is no room
*:lobby
P:princess room
M:monster room
_:empty room
-----
? ? ? ? ? ? ? ? ? ?
? ? ? ? ? ? ? ? ? ?
? ? ? ? ? ? ? ? ? ?
? ? ? ? ? ? ? ? ? ?
? ? ? ? ? ? ? ? ? ?
? ? ? ? ? ? ? ? ? ?
? @ ? ? ? ? ? ? ? ?
? ? ? ? ? ? ? ? ? ?
* ? ? ? ? ? ? ? ? ?
_ _ _ _ _ _ _ _ _ ?

Welcome to the empty room. There are 4 exits:north, south, west, east,
Enter your command:
go west

```

(3)当输入正确的指令时，程序会调用**MovePlayer()**函数，修改玩家的位置

(4)当输入错误的指令时，程序会输出，并且不修改玩家的位置

```

Welcome to the empty room. There are 2 exits:south, west,
Enter your command:
go north
this move is illegal, please try another

```

(5)如果要退出程序，需要输入**exit**

```

Welcome to the empty room. There are 2 exits:south, west,
Enter your command:
exit

```

(5)当玩家遇到怪兽时，游戏结束

(6)当玩家遇到公主时，程序会输出**You meet the princess, please go back to the lobby**

```

_ # _ M
_ # _ 
* _ _ 
_ _ _ P
_ _ _

Welcome to the empty room. There are 3 exits:north, west, east,
Enter your command:
go east

You meet the princess, please go back to the lobby

```

并且在后续的地图中，可以看到，公主原来所在的房间里面已经没有公主了

```
the map is as bellow, size=5*5
H:the position of the player
#:there is no room
*:lobby
P:princess room
M:monster room
_:empty room
```

```
-----
?  ?  ?  ?  ?
?  ?  ?  ?  ?
*  ?  ?  ?  ?
_  ?  ?  ?  ?
_  _  _  @  _
```

(7)在找到公主的情况下，返回大厅，游戏结束

```
?  ?  ?  ?  ?
?  ?  ?  ?  ?
*  ?  ?  ?  ?
@  ?  ?  ?  ?
_  _  _  _  _
```

```
Welcome to the empty room. There are 3 exits:north, south, east,
Enter your command:
go north
```

```
You save the princess, congratulations!
```

2.4 类的封装

游戏的控制程序封装为一个类**Adventure**，通过调用**adventure.StartAdventure()**函数，进行游戏

三、测试样例

由于每一次的地图为系统随机生成，本程序并没有固定的测试样例

所以，通过一次运行，作为本程序的测试样例

```
please input the size of the map(<30*30)
row:5
column:5
now is init a map, its size is 5*5
create map succeed, cost 0ms
```

```
the map is as bellow,size=5*5
H:the position of the player
#:there is no room
*:lobby
P:princess room
M:monster room
_:empty room
```

```
-----
? ? ? ? @
? ? ? ? ?
? ? ? ? ?
? ? ? ? ?
? ? ? ? ?
```

```
Welcome to the lobby. There are 2 exits:south, west,
Enter your command:
go south
```

```
the map is as bellow,size=5*5
H:the position of the player
#:there is no room
*:lobby
P:princess room
M:monster room
_:empty room
```

```
-----
? ? ? ? *
? ? ? ? @
? ? ? ? ?
? ? ? ? ?
? ? ? ? ?
```

```
Welcome to the empty room. There are 3 exits:north, south, west,
Enter your command:
go south
```

the map is as bellow, size=5*5
H:the position of the player
#:there is no room
*:lobby
P:princess room
M:monster room
_:empty room

```
-----  
?  ?  ?  ?  *  
?  ?  ?  ?  _  
?  ?  ?  ?  @  
?  ?  ?  ?  ?  
?  ?  ?  ?  ?
```

Welcome to the empty room. There are 3 exits:north, south, west,
Enter your command:
go south

You meet the princess, please go back to the lobby

the map is as bellow, size=5*5
H:the position of the player
#:there is no room
*:lobby
P:princess room
M:monster room
_:empty room

```
-----  
?  ?  ?  ?  *  
?  ?  ?  ?  _  
?  ?  ?  ?  _  
?  ?  ?  ?  @  
?  ?  ?  ?  ?
```

Welcome to the empty room. There are 3 exits:north, south, west,
Enter your command:
go north

```
the map is as bellow,size=5*5
H:the position of the player
#:there is no room
*:lobby
P:princess room
M:monster room
_:empty room
```

```
-----
? ? ? ? *
? ? ? ? _
? ? ? ? @
? ? ? ? -
? ? ? ? _
```

```
Welcome to the empty room. There are 3 exits:north, south, west,
Enter your command:
go north
```

```
the map is as bellow,size=5*5
H:the position of the player
#:there is no room
*:lobby
P:princess room
M:monster room
_:empty room
```

```
-----
? ? ? ? *
? ? ? ? @
? ? ? ? -
? ? ? ? _
? ? ? ? _
```

```
Welcome to the empty room. There are 3 exits:north, south, west,
Enter your command:
go north
```

```
You save the princess, congratulations!
```

附录：源代码

```
1 #include <iostream>
2 #include <ctime>
3 #include <algorithm>
4 #include <string>
5 using namespace std;
6 const string rootName[] = {"cannot_go","lobby","princess room","monster
  room","empty room"};
7 class Adventure {
8     static const int maxn = 30+5;
```

```

9
10 //地图是一个n行m列的矩阵
11 int n, m;
12 int map[maxn][maxn];
13 bool vis[maxn][maxn];
14 bool NextStep[4];
15
16 //房间的种类
17 const int cannot_go = 0; //该房间不能进入
18 const int lobby = 1; //出生点(大厅)
19 const int princessRoom = 2; //房间有公主
20 const int monsterRoom = 3; //房间有怪兽
21 const int emptyRoom = 4; //空房间
22
23 //位置信息
24 struct Position {
25     int init_x, init_y; //初始的位置==大厅的位置
26     int monster_x, monster_y; //怪兽的位置
27     int princess_x, princess_y; //公主的位置
28     int now_x, now_y; //勇者现在的位置
29     void Print() {
30         printf("初始位置 = (%d,%d)\n", init_x, init_y);
31         printf("怪物位置 = (%d,%d)\n", monster_x, monster_y);
32         printf("公主位置 = (%d,%d)\n", princess_x, princess_y);
33         printf("勇者位置 = (%d,%d)\n", now_x, now_y);
34     }
35 }position;
36
37 //获得随机数
38 int random(int L, int R) {
39     int len = R - L + 1;
40     return rand() * rand() % len + 1;
41 }
42
43 //高亮输出,0-黑,1-红,2-绿,3-黄,4-蓝,5-紫,6-深绿,7-白
44 void HighLightPrint(const char* s, int color) {
45     switch (color) {
46     case 0://黑
47         printf("\033[30m"); break;
48     case 1://红
49         printf("\033[31m"); break;
50     case 2://绿
51         printf("\033[32m"); break;
52     case 3://黄
53         printf("\033[33m"); break;
54     case 4://蓝
55         printf("\033[34m"); break;
56     case 5://紫
57         printf("\033[35m"); break;
58     case 6://深绿
59         printf("\033[36m"); break;
60     case 7://白
61         printf("\033[37m"); break;
62     }
63     printf("%s", s);
64     printf("\033[0m");
65 }
66

```



```

67 //得到房间(x,y)的类型
68 int GetRoom(int x, int y) {
69     if (x <= 0 || x > n)return 0;
70     if (y <= 0 || y > m)return 0;
71     return map[x][y];
72 }
73
74 //得到当前房间的类型
75 int GetNowRoom() {
76     vis[position.now_x][position.now_y] = true;
77     return GetRoom(position.now_x, position.now_y);
78 }
79
80 //创建出生点->公主的路径
81 void CreateSucceedPath() {
82     int x1 = position.init_x;
83     int y1 = position.init_y;
84     int x2 = position.princess_x;
85     int y2 = position.princess_y;
86     if (x1 <= x2 && y1 <= y2) { //出生点在公主的左下角
87         for (int i = x1 + 1; i < x2; i++)
88             map[i][y1] = emptyRoom;
89         for (int i = y1 + 1; i < y2; i++)
90             map[x2][i] = emptyRoom;
91         if (y1 != y2)map[x2][y1] = emptyRoom;
92     }
93     else if (x1 <= x2 && y1 >= y2) { //出生点在公主的左上角
94         for (int i = x1 + 1; i < x2; i++)
95             map[i][y1] = emptyRoom;
96         for (int i = y1 - 1; i > y2; i--)
97             map[x2][i] = emptyRoom;
98         if (y1 != y2)map[x2][y1] = emptyRoom;
99     }
100     else if (x1 >= x2 && y1 <= y2) { //出生点在公主的右下角
101         for (int i = x1 - 1; i > x2; i--)
102             map[i][y1] = emptyRoom;
103         for (int i = y1 + 1; i < y2; i++)
104             map[x2][i] = emptyRoom;
105         if (y1 != y2)map[x2][y1] = emptyRoom;
106     }
107     else if (x1 >= x2 && y1 >= y2) { //出生点在公主的右上角
108         for (int i = x1 - 1; i > x2; i--)
109             map[i][y1] = emptyRoom;
110         for (int i = y1 - 1; i > y2; i--)
111             map[x2][i] = emptyRoom;
112         if (y1 != y2)map[x2][y1] = emptyRoom;
113     }
114 }
115
116 //创建大厅、公主、怪物的位置
117 void CreateSpecialPosition() {
118     //大厅位置
119     position.init_x = random(1, n);
120     position.init_y = random(1, m);
121     map[position.init_x][position.init_y] = lobby;
122     //出生点位置==大厅位置
123     position.now_x = position.init_x;

```

```

125     position.now_y = position.init_y;
126     //公主位置
127     do {
128         position.princess_x = random(1, n);
129         position.princess_y = random(1, m);
130     } while (map[position.princess_x][position.princess_y] != -1);
131     map[position.princess_x][position.princess_y] = princessRoom;
132     CreateSucceedPath();
133     //怪物位置
134     do {
135         position.monster_x = random(1, n);
136         position.monster_y = random(1, m);
137     } while (map[position.monster_x][position.monster_y] != -1);
138     map[position.monster_x][position.monster_y] = monsterRoom;
139 }
140
141 //创建其它房间,每个房间可以走的概率为x%
142 void CreateOtherRoom(int x) {
143     for (int i = 1; i <= n; i++)
144         for (int j = 1; j <= m; j++)
145             if (map[i][j] == -1) {
146                 int now = random(1, 100);
147                 if (now < x) map[i][j] = emptyRoom;
148                 else map[i][j] = cannot_go;
149             }
150 }
151
152 //将地图初始化
153 void CreateMap() {
154     srand((int)time(0));
155     for (int i = 1; i <= n; i++)
156         for (int j = 1; j <= m; j++)
157             map[i][j] = -1, vis[i][j] = false;
158     CreateSpecialPosition();
159     CreateOtherRoom(80);
160     vis[position.init_x][position.init_y] = true;
161 }
162
163 //打印地图,0-全部,1-部分
164 void PrintMap(int type) {
165     cout << "the map is as bellow,size=" << n << "*" << m << endl;
166     cout << "H:the position of the player\n";
167     cout << "#:there is no room\n";
168     cout << " *:lobby\n";
169     cout << "P:princess room\n";
170     cout << "M:monster room\n";
171     cout << "_:empty room\n";
172     cout << "-----\n";
173     for (int i = 1; i <= n; i++) {
174         for (int j = 1; j <= m; j++) {
175             if (i == position.now_x && j == position.now_y) { //玩家位置
176                 HighLightPrint(" @ ", 7);
177                 continue;
178             }
179             if (!vis[i][j] && type == 1) {
180                 cout << " ? ";
181                 continue;
182             }

```

```

183         switch (map[i][j]) {
184             case 0:cout << " # "; break;//不能走
185             case 1:HighLightPrint(" * ", 1); break;//出生点(大厅)
186             case 2:HighLightPrint(" P ", 3); break;//公主
187             case 3:HighLightPrint(" M ", 1); break;//怪兽
188             case 4:cout << " _ "; break;//空房间
189             default:cout << " | "; break;
190         }
191     }
192     printf("\n");
193 }
194 printf("\n");
195 }
196
197 //北-0,南-1,西-2,东-3
198 void MovePlayer(int head) {
199     int nextRoom;
200     if (head == 0) {//向上(北)
201         nextRoom = GetRoom(position.now_x - 1, position.now_y);
202         if (nextRoom == 0)
203             HighLightPrint("this move is illegal, please try
another\n", 1);
204         else
205             position.now_x--;
206     }
207     else if (head == 1) {//向下(南)
208         nextRoom = GetRoom(position.now_x + 1, position.now_y);
209         if (nextRoom == 0)
210             HighLightPrint("this move is illegal, please try
another\n", 1);
211         else
212             position.now_x++;
213     }
214     else if (head == 2) {//向左(西)
215         nextRoom = GetRoom(position.now_x, position.now_y - 1);
216         if (nextRoom == 0)
217             HighLightPrint("this move is illegal, please try
another\n", 1);
218         else
219             position.now_y--;
220     }
221     else {//向右(东)
222         nextRoom = GetRoom(position.now_x, position.now_y + 1);
223         if (nextRoom == 0)
224             HighLightPrint("this move is illegal, please try
another\n", 1);
225         else
226             position.now_y++;
227     }
228 }
229
230 //获取当前房间的出口
231 int GetNextStep() {
232     int x = position.now_x, y = position.now_y;
233     int ans = 0;
234     //北-0,南-1,西-2,东-3
235     if (GetRoom(x - 1, y) != 0)NextStep[0] = true, ans++;
236     else NextStep[0] = false;

```

```

237     if (GetRoom(x + 1, y) != 0)NextStep[1] = true, ans++;
238     else NextStep[1] = false;
239     if (GetRoom(x, y - 1) != 0)NextStep[2] = true, ans++;
240     else NextStep[2] = false;
241     if (GetRoom(x, y + 1) != 0)NextStep[3] = true, ans++;
242     else NextStep[3] = false;
243     return ans;
244 }
245
246 //将公主从房间带出
247 void TakeOutPrincess() {
248     map[position.princess_x][position.princess_y] = emptyRoom;
249 }
250 public:
251     //进行游戏的初始化
252     void reset() {
253         cout << "please input the size of the map(<30*30)\n";
254         cout << "row:";
255         cin >> n;
256         //n = 10;
257         cout << "column:";
258         cin >> m;
259         //m = 10;
260         cout << "now is init a map, its size is " << n << "*" << m << "\n";
261         int nowTime = clock();
262         CreateMap();
263         cout << "create map succeed, cost " << (clock() - nowTime) <<
264         "ms\n\n\n";
265     }
266
267 //开始游戏
268 void StartAdventure() {
269     reset();
270     string command;
271     //找公主
272     while (true) {
273         if (GetNowRoom() == monsterRoom) {
274             HighLightPrint("You meet the monster, game end\n", 1);
275             HighLightPrint("Please try another time\n", 1);
276             cout << "\n\n";
277             return;
278         }
279         else if (GetNowRoom() == princessRoom) {
280             HighLightPrint("You meet the princess, please go back to
281 the lobby\n", 1);
282             TakeOutPrincess();
283             cout << "\n\n";
284             break;
285         }
286     }
287
288     int room = GetNowRoom();
289     PrintMap(1);
290     cout << "welcome to the " << rootName[room] << ". ";
291     cout << "There are " << GetNextStep() << " exits:";
292     //北-0,南-1,西-2,东-3
293     if (NextStep[0])cout << "north, ";
294     if (NextStep[1])cout << "south, ";
295     if (NextStep[2])cout << "west, ";

```

```

293         if (NextStep[3])cout << "east, ";
294         cout << endl;
295         cout << "Enter your command:" << endl;
296
297         cin >> command;
298         if (command == "exit")return;
299         if (command != "go") {
300             HighLightPrint("the command is illegal\n\n", 1);
301             continue;
302         }
303         cin >> command;
304
305         if (command == "north")MovePlayer(0);
306         else if (command == "south")MovePlayer(1);
307         else if (command == "west")MovePlayer(2);
308         else if (command == "east")MovePlayer(3);
309         else HighLightPrint("the command is illegal\n", 1);
310         cout << "\n\n";
311     }
312     //回大厅
313     while (true) {
314         if (GetNowRoom() == monsterRoom) {
315             HighLightPrint("You meet the monster, game end\n", 1);
316             HighLightPrint("Please try another time\n", 1);
317             return;
318         }
319         else if (GetNowRoom() == lobby) {
320             HighLightPrint("You save the princess, congratulations!\n",
1);
321             return;
322         }
323
324         int room = GetNowRoom();
325         PrintMap(1);
326         cout << "\n\n";
327         cout << "welcome to the " << rootName[room] << ". ";
328         cout << "There are " << GetNextStep() << " exits:";
329         //北-0,南-1,西-2,东-3
330         if (NextStep[0])cout << "north, ";
331         if (NextStep[1])cout << "south, ";
332         if (NextStep[2])cout << "west, ";
333         if (NextStep[3])cout << "east, ";
334         cout << endl;
335         cout << "Enter your command:" << endl;
336
337         cin >> command;
338         if (command == "exit")return;
339         if (command != "go") {
340             HighLightPrint("the command is illegal\n\n", 1);
341             continue;
342         }
343         cin >> command;
344
345         if (command == "north")MovePlayer(0);
346         else if (command == "south")MovePlayer(1);
347         else if (command == "west")MovePlayer(2);
348         else if (command == "east")MovePlayer(3);
349         else HighLightPrint("the command is illegal\n", 1);

```

```
350         cout << "\\n\\n";
351     }
352 }
353 };
354 Adventure now;
355 int main(){
356     now.StartAdventure();
357     return 0;
358 }
```