



学院: 数据科学与计算机学院      专业: 软件工程

姓名: 温卓沛      学号: 17343115      日期: 2018 年 05 月 17 日

实验题目: 期中 Project—Simple Nethack

## 一、代码运行方式

### 1. 编译方式:

1) g++ EQUIPMENT.cpp global.cpp main.cpp MAP.cpp monster.cpp PACKAGE.cpp player.cpp WEAPON.cpp -o nethacks

2) 直接将全部代码加入到 Microsoft Visual Studio 2017 一个新工程里, 然后点击编译

### 2. 运行方式:

1) start nethacks.exe

2) 在 Microsoft Visual Studio 2017 编译后使用其运行按钮

## 二、操作方法

'w' / 'W' ——up

's' / 'S' ——down

'a' / 'A' ——left

'd' / 'D' ——right

'q' / 'Q' ——left-up

'e' / 'E' ——right-up

'z' / 'Z' ——left-down

'c' / 'C' ——right-down

o ——check your package

u ——use your equipment or your weapon in your package

h ——to know what the direction button is

m ——details about this game

@是玩家 M 是怪物 . 和#是道路 其它字母则各有其自身的意义。

## 三、设计思路

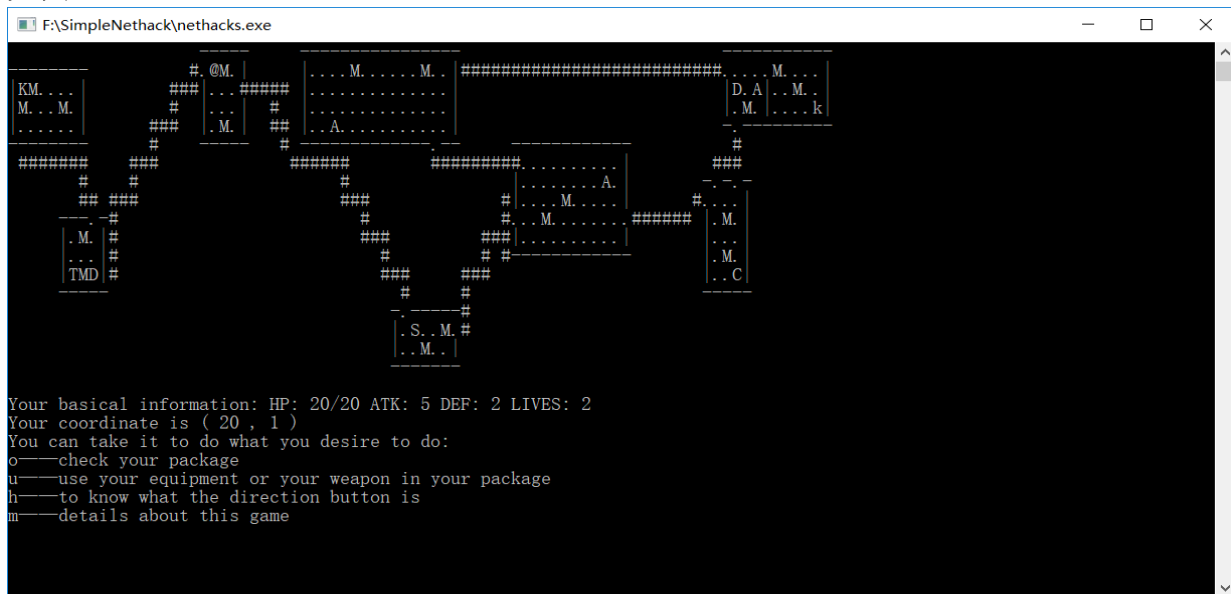
1. 对于 monster 的类而言, 我使用 get 函数与 change 函数来实现私有数据成员的接口公有化, 而其他成员函数也只有寻径与攻击与被攻击等。

2. 因此, player 的类的实现便尤为重要: 同样的, 用于 monster 类相似的方法封装私有数据, 而包含 package 类的 player 类则以一个最为重要的 action 成员函数来对输入与其行为之间的关系进行判断。

3. 而运算符重载, 我则应用在了一些<<的运算符重载, 使得一些输出主函数更为简便。



如图:

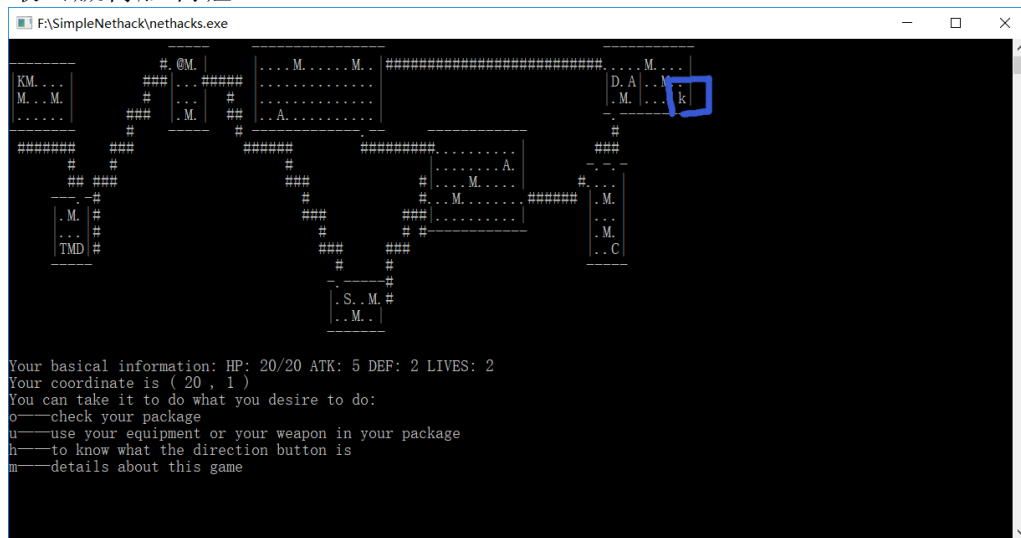


## 四、改进空间

1. 虽然尝试过用 rand 来生成诸多个房间的地图,但是效果不好.因此地图是写死的,这局限了游戏的可玩性,同时因为地图是写死的,MAP 类并没有开发得很好,仅仅只是有生成一个固定的地图的功能。
2. PLAY 类的结构相对复杂,条件判断输出语句近半在类内实现,这对于 test 函数要求较高。

## 五、其他

此并非简单的刷怪刷经验升级,而是只有一层的地图,需要达成某些条件才能赢得最终赢得胜利喔。



(小钥匙)