

# Simple Nethack 项目报告

学院：数据科学与计算机学院

专业：软件工程

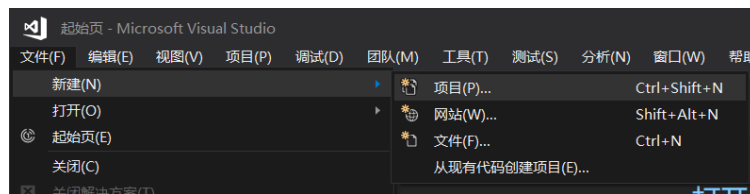
姓名：张伟焜

学号：17343155

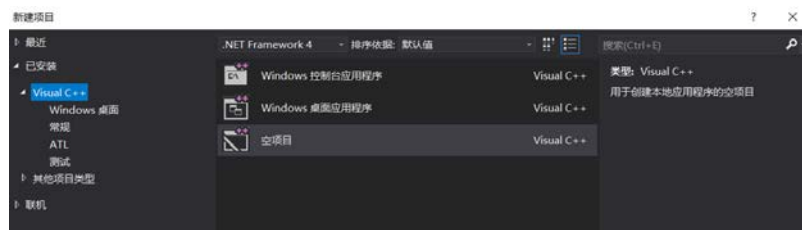
## 代码运行方法

代码编写环境：Win10 Visual Studio2017

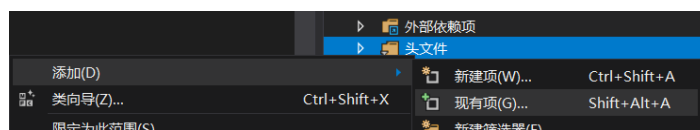
打开 visual studio 点击 文件->新建->项目



选择 visual C++ 空项目



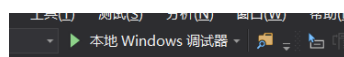
在右侧资源管理器中 右击头文件 添加现有项，将 code->include 中四个头文件添加进项目



在右侧资源管理器中 右击源文件 添加现有项，将 code->src 中五个源文件添加进项目



最后点击上方按钮



运行

## 操作方法

按下 【y】 , 【u】 , 【h】 , 【j】 , 【k】 , 【l】 , 【b】 , 【n】 ,进行八向移动；

按下 【q】 打开帮助页；

按下 【p】 打开背包页；

【z】 技能一次消耗 30MP；

【x】 技能一次消耗 20MP；

药瓶 【b】 捡去后可进入背包页进行【手动使用】；

宝剑 【s】 & 盾牌 【p】 拾取后自动增加【人物属性】；

你的目标是拾取项链 【{】，并将它放到祭台 【T】 上！

(字母操作都是用【小写字母】)

## 设计思路

构造了四个类：player, monster, package, item

player: 具有属性 HP, MP, attack(攻击值), defense(伤害值), kind, pos\_x, pos\_y 属性。具有返回位置，返回 HP，MP，八方向移动，返回并加减属性，输出显示等函数。

monster：具有 HP, attack, defense, kind, pos\_x, pos\_y battle 等属性。具有返回位置，返回血量，减少血量,设置和判断战斗状态，复制，八方向移动等函数。

item：具有 kind, pos\_x, pos\_y 等属性。具有返回位置，返回符号和种类等函数。

package：medicine, sword, shield，necklace等属性。具有打印，增加物品，减少物品，判断是否得到项链等函数。

运算符重载：<< （用于输出对象信息，monster 和 player）  
= （用于传递复制 item 和 monster 对象）

地图：

char backups[MAP\_Y][MAP\_X]; //开发者可以直接在该数组中设计地图,添加 item 或 monster, 且可以随意梗概 monster item player 初始位置。

char map\_char[MAP\_Y][MAP\_X]; //在 initialize 函数中将 backup 传递给 map\_char 用于打印地图。

int map\_int[MAP\_Y][MAP\_X] = { 0 };//用于记录不同图标的编号（墙是 1 门是 2 路是 3 人是 4 怪是 5 物品是 6）不同的编号对应不同的打印颜色。

int vision\_record, [MAP\_Y][MAP\_X] = { 0 };//vision\_record 用于记录视野。当地图对应位置视野不为 0 时显示。

怪物移动：计算曼哈顿距离，自动进行寻路。

运行流程：

1. 帮助界面
2. 装载地图 initialization () 中生成类的对象，并将地图传入 map\_char 中记录。
3. 当 run 为真时玩家可以进行操作（每次操作后都刷新一次地图）
  - 读取指令：若为 p 或 q，则打开背包或帮助界面
  - 若为 yuhjklbn 则进行移动
    - 人物移动
    - 怪物移动
  - 若为 z x 则
    - 使用技能
- 当 run 为假时游戏结束并调用 end() 函数进行胜负的判断
4. 结束

## 拓展点

### 1. 视野系统

利用 vision\_record 数组进行记录，初始为 0，player 当前及走过的位置的八个方向的 vision\_record++，打印地图时，打印不为零的位置的地图。

### 2. 视野技能

按下 z 判断当前 MP 是否大于 30，条件为真就将所有 vision\_record 值++，Sleep(3000) 即 3 秒后 vision\_record 值--，回复原来视野。

### 3. 伤害波动

在 battle () 函数中，随机产生 1~10 的随机数，再乘 0.1，（攻击方 attack-防御方 defense）\*伤害波动率。 增加游戏的随机性和可玩性。

### 4. 地图的可开发及可拓展性

开发者在程序运行前，可在 backups 数组中随意放置 monster 和 item ([b],[p],[s],[{}])，并能任意设置 player 的初始位置。（其中 monster 有 S，M，L 三种 可供开发人员选择，但为了让界面中符号不过于繁多，最终运行时同一显示为 M）由于 monster 和 item 类的对象都是数组，所以要及时更改#define 中的 MONSTER\_TOTAL 和 ITEM\_TOTAL

还有要注意#define 中的 MAP\_X 和 MAP\_Y 在更改地图时要做出相应修改。

## 不足及改进空间

1. 只设计了一层,地图空间太小，今后可以考虑扩展。
2. 由于怪物太少没能实行等级和升级系统。可以考虑加层数和怪物数，实现等级系统。
3. 设计了门 却没有设计钥匙以及开关门操作。有待今后完善。
4. 由于时间关系,monster 死后的掉落物品没有实现。【思路】:在 monster 类中添加 drop\_rate 和 drop\_kind 两个参数，并随机生成。前者表示掉落率，后者表示凋落物种类。
5. 本来想做备份，利用 char backups[ ][ ]使游戏可以读档 但由于时间关系没能实现。  
【思路】:将存档时的地图使用文件输出，产生一个数据文件。在读档时读取该文件，再触发 initialization () 函数，自动生成 player monster item 等对象。