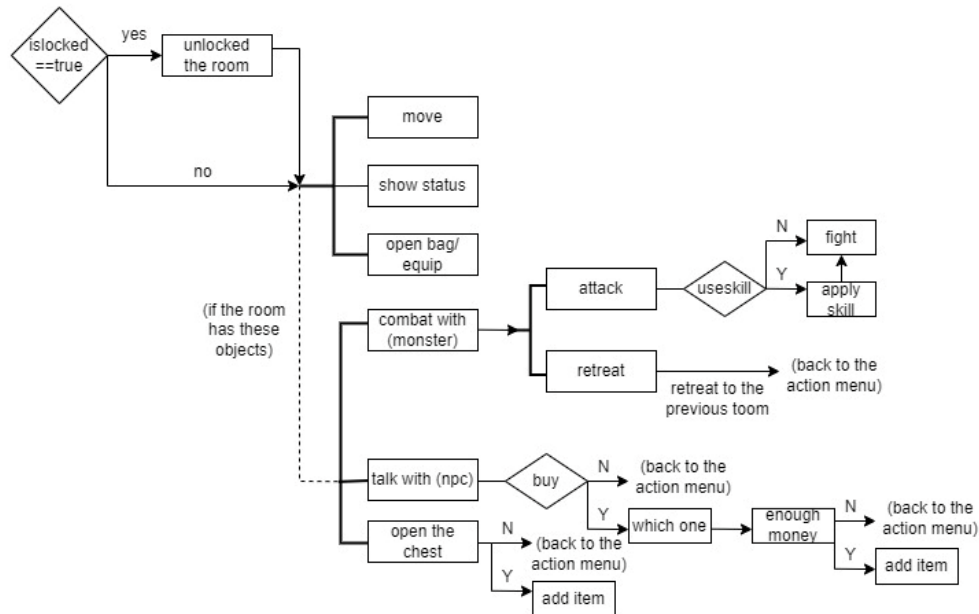


Dungeon game

一、Implementation detailed

1. Game introduction: [Dungeon class]

- (1) 玩家進入遊戲後[startGame()]，會先輸入 user name，並選擇職業 [createPlayer()]。
- (2) 接著出現 action menu(move、show status、open bag/equip)讓玩家選擇行動。[chooseAction()]



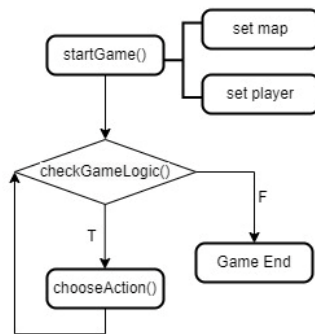
- 移動:根據所在位置，只會顯示可以移動的方向[handleMovement()]
 - Show status: 顯示玩家狀態:包含 health、attack、defense、money (以及名字和職業)
 - open bag/equip:玩家在遊戲過程中，會藉由與 npc 交易和打開寶箱獲得物品並收在<vector>bag 中，玩家可以選擇要裝備的物品，則該物品在 bag 中會消失。裝備系統會在 player 中再詳細說明。
- 根據玩家所在的房間內含有的 object(可能是 monster、npc、chest)，會另外出現對應的選項(combat、talk、open)，其中 monster 被打敗後會消失、chest 若已打開過會顯示 opened。
- money system:
 - i. 賺錢方式:打敗怪物、打開寶箱
 - ii. 花錢方式:解鎖房間、和 npc 交易

(3) 邏輯: [checkGameLogic()]

當玩家的 health≤0 即失敗並遊戲結束、當玩家打敗 boss 即獲勝並遊戲結束。/*判斷 boss 的 code 和 checkisdead(): 藉由 this->player.getCurrentRoom() == &this->rooms.back(); 判斷是否為最後的房

間，接著用 bool hasBoss 判斷 boss 是否在內，若否則 checkGameLogic() 會 return false，並結束程式。*/

(4) [runDungeon()]:



2. map design: [Dungeon ::createMap()] [Room class]

(1) set object:

宣告要放進 room 的物品。/*code 實作方式: 並非採取 random 的模式，而是固定地圖，先 declare 15 個 vector<Object*>obj，分別 declare 好 chest、npc、monster，其中 chest 和 npcc，又需再另外 declare: vector<Item>，即一個 chest 或 npc 會有多種物品，並 push_back 進 item，最後再將 obj push_back 不同物品。*/

(2) create & link room:

藉由以下方式，重複該動作，創建 16 個房間。

首先 declare room1 到 room15，接著藉由 linked list 將上下左右的 room 分別連接，最終即可連起整個地圖。

```
Room *room1 = new Room(false,false, 1, obj1);
```

```
room1->setUpRoom(NULL); room1->setDownRoom(room5); room1->setLeftRoom(room0); room1->setRightRoom(room2);
```

最後將 room1~room15 push_back 到 dungeon::<vector>rooms 裡:

```
this->rooms.push_back(*room1);
```

(3) 地圖設計共有 16 格(4*4)，如下:

Room0	Room1	Room2	Room3
<Start>	NPC [Merchant]	Chest [coin30]	<Locked> Chest[coin50、 shield]
Room4	Room5	Room6	Room7
NPC [Merchant]	Monster [devil]	<Locked> NPC [Granny]	Monster [zombie]
Room8	Room9	Room10	Room11
Monster [zombie]	Chest[coin20、 potions]	Monster [ogres]	<empty>

Room12	Room13	Room14	Room15
<Locked> Chest[medicine 、 holy sword]	NPC [Colonel]	Chest[coin20、 laser gun]	<End> Monster [BOSS]

(4)[additional design]比較特殊的是，有的房間是鎖住的，需要花\$20 去解鎖，若不解鎖或錢不夠，則無法打開和知道內含的 object 是甚麼。在 Room class 中有 bool isExit 和 bool isLocked 分別判斷，是不是最後一個(含有 boss)房間和是否是鎖住的。

3. Player:

(1) 玩家可以選擇不同職業，而各職業會有對應的不同初始值、以及遇到 monster 時可使用的技能、和對應的武器(當得到該武器時，可以使原本的加成再獲得更高的數值)。

Occupation	(HP/AT/DF/\$)	Skill (HP/AT/DF)	Weapon(HP/AT/DF)
Magician	(100,20,15,15)	Freeze Spell(-5,+10,+5)	Potions(+5/+5/+5)
Warrior	(100,30,5,10)	Roar(-5,+15,+3)	Shield(+5/+5/+5)
Knight	(100,25,10,20)	Lightning(-5,+15,+5)	Holy sword(+5/+5/+5)

(2) [useskill(Player*)、unuseskill(Player*)] 技能: 若此輪與 monster 的對戰結束，attack 和 defense 的數值會恢復原樣，另要注意，使用技能時同時會減少 health。

(3) [addItem(Item)]當打開 chest 或買物品後，會將該物品加在 vector<Item> inventory 中。

(4) [increaseStates()、changeMoney()、changeRoom(Room*)]藉由此三個 function 改變 player 的數值以及所在房間。

(5) [additional design] [equipthe(Object*)]玩家獲取物品後，除非是錢幣外，不能直接使用並增加 attack、defense 等數值，需要經過”equip”的步驟，並從 inventory 中選擇要 equip 的物品。若裝備對應職業的武器(如 Magician equip the potions)，數值會再提升。

(6) [popbag(Item*)、popequip (Item*)]玩家裝備物品後，該物品會新增至<vector>equip，並利用 popbag()函式將物品從<vector>inventory 移除。同理，玩家與 monster 的對戰結束後，除非是武器類別的物品，否則也會利用 popequip()函式將其從<vector>equip 中移除。

4. NPC

Name	Item	Script
Granny Susan	Medicine、Potions、Magic book	Magic is everything hahaha!
Merchant	Holy sword、Poison gas	Money is the power!
Colonel James	Bomb、Laser gun、Firecracker	Let failure be your energy!

5. Monster

Name	(HP/AT/DF)	Reward
------	------------	--------

Little devil	(20, 18, 10)	None
Zombie	(50, 35, 25)	\$10
Ogres	(60, 45, 30)	\$20
BOSSYBOSS	(160, 60, 40)	WIN

6. Chest:寶箱內可能有:錢幣、Item。
7. Virtual function: `[bool triggerEvent(Object*)]`
 - (1) object class 內的是 pure virtual function，根據不同 object 會做不同的事情。
 - 以下 class 內的 triggerEvent()都是 triggered by Object*Player:
 - (2) item:物品加入 player 的 bag，並扣除 player 的錢。
 - (3) player:用作 show player status，顯示玩家目前的 health、attack、defense 及 money。
 - (4) monster:有關玩家和 monster 的 combat system。

玩家可選 Attack 或 Retreat(退回前一個房間)，若 Attack，會先詢問是否要使用技能，接著利用 while(1)迴圈:

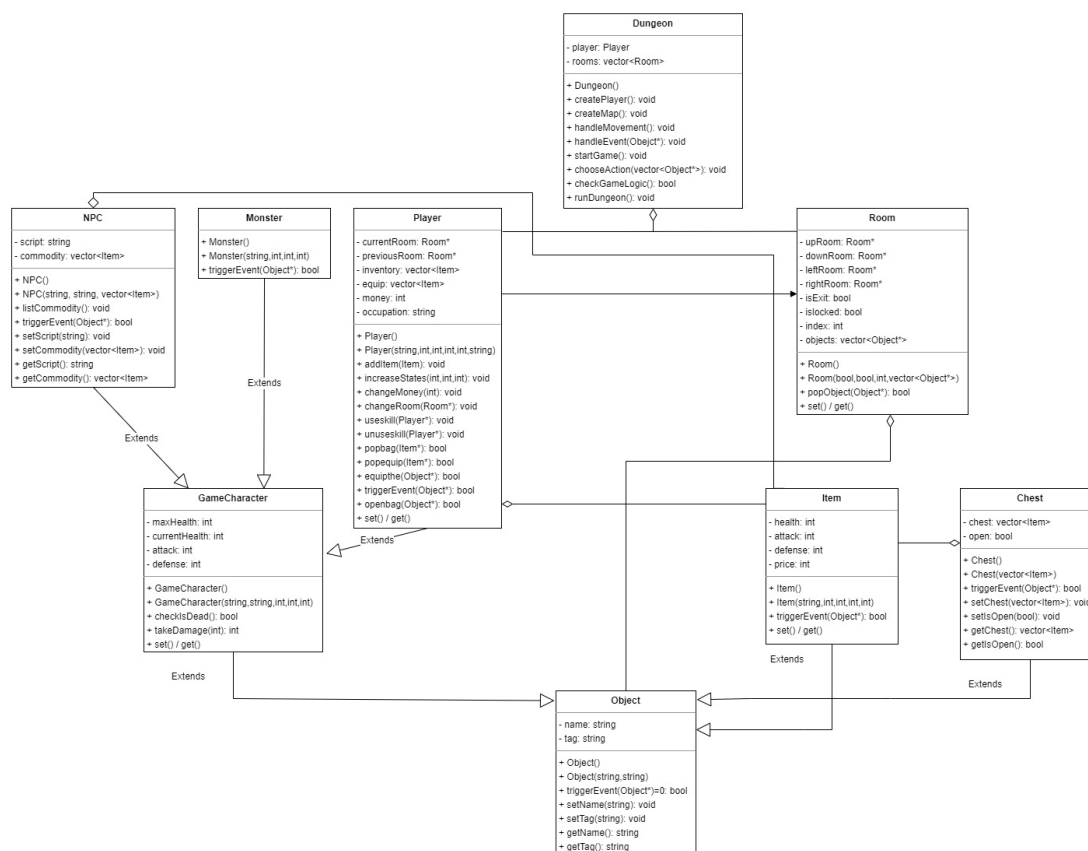
 - I. 玩家先攻擊 monster
 - II. Check monster 是否 dead，若是則 break; 若玩家打敗 Zombie 可獲得\$10，打敗 Ogres 可獲得\$20
 - III. Monster 攻擊玩家
 - IV. Check 玩家是否 dead，若是則 break; 否則繼續迴圈
 - (5) NPC:有關玩家和 NPC 的 trading system。

NPC 會先輸出其擁有的 Item，並詢問玩家是否購買，若玩家回答是，則會詢問玩家要購買哪一項 Item，若玩家的錢幣足夠便會呼叫 Item 的 triggerEvent()。
 - (6) Chest:將寶箱內的物品加入玩家的 bag，若物品為錢幣則增加玩家的 Money，最後將該 chest 的 open bool 值設為 true。

二、UML Design(圖、整體架構)

1. 以 Object class 為父類別，遊戲中主要的 Object 可分為三大類:Game Character、Item、Chest，這三個 class 各繼承了 Object class。
2. 其中，Game Character 又分為三大類: NPC、Monster、Player，因此這三項 class 又各繼承 Game Character class。
3. Room class 為另外獨立出來的 class。Room 擁有 object，因此兩個 class 間是”aggregation”的關聯。
4. Dungeon class 作為遊戲運作的最主要 class，負責遊戲整體的運行邏輯及流程判斷。而 dungeon class 擁有 player 和 room。
5. NPC、Player、Chest 都含有 Item，因此此三項與 Item 之間也有”aggregation”的關聯。

6. Player 需要知道 current room 和 previous room，因此 player 和 room 之間也有關聯，在以下圖中以”association”關聯表示。



(註、礙於篇幅，圖中某些 class 的 set 和 get 函數以 set()/get()省略)

三、Results

1.遊戲開始畫面: 為角色命名，並選擇職業，input 值錯誤會請重新輸入。

```

Welcome to the Dungeon!.....
Please name your character: test
Please choose which occupation do you want to be:
A.Magician
B.Warrior
C.Knight
A
Wrong input, enter again>>
A

```

2.進入 Action Menu:選擇要做的事情，若輸入錯誤會再進入一次。Room0 是起點，因此沒有放 object。

(1)選擇 Move: 顯示所在的 room 編號，並列出可走的方向，接著顯示走向的 room 編號。

```

.....
What do you want to do? (You are in the Room 0)
A.Move
B.Show your status
C.Open bag/Equip
A
You are now in room#0. You can move:
(1)go down
(2)go right
2
YA! You now move to room#1.
.....
What do you want to do? (You are in the Room 1)
A.Talk to Merchant
B.Move
C.Show your status
D.Open bag/Equip

```

(2)選擇 Show status:

```
.....
What do you want to do? (You are in the Room 0)
A.Move
B.Show your status
C.Open bag/Equip
B
.....
[test]
Occupation: Magician
Health: 100/100
Attack: 20
Defense: 15
Money: $15
```

(3)選擇 Open bag/equip:

```
.....
What do you want to do? (You are in the Room 0)
A.Move
B.Show your status
C.Open bag/Equip
C
.....
Your bag has: nothing:)
```

-經過蒐集，玩家的背包會有多樣物品，而玩家可以選擇要裝備的物品:

```
.....
Your bag has:
(1) Poison gas (2) Potions
Which one to equip?(enter number only/0 to exit)
1
```

-左圖為原本狀態，右圖為裝備後的狀態，而裝備後背包內的物品也會消失

<pre>..... [test] Occupation: Magician Health: 97/100 Attack: 20 Defense: 15 Money: \$50</pre>	<pre>..... [test] Occupation: Magician Health: 112/115 Attack: 30 Defense: 30 Money: \$50</pre>	<pre>..... Your bag has: (1) Potions Which one to equip?(enter number only/0 to exit)</pre>
--	---	---

-當玩家裝備其職業對應武器(如 Magician 裝備 Potions)會有加乘效果:

Item the_item8("Potions",10,5,10,15); player->increaseStates(5,5,5);

```
.....
[test]
Occupation: Magician
Health: 127/130
Attack: 40
Defense: 45
Money: $50
```

3.遇到 NPC

NPC 會詢問是否購買物品，若選擇 N(或輸入不是 'Y'):

```
.....
What do you want to do? (You are in the Room 1)
A.Talk to Merchant
B.Move
C.Show your status
D.Open bag/Equip
A
.....
Money is the power!
Merchant: Hi! I am Merchant. I have something for you to choose:
1. Holy sword cost$20.
2. Poison gas cost$15.
Merchant: Do you want to buy something? (Y/N)
N
byebye
```

選擇 Y，則會詢問購買哪一項物品，並會加入在玩家的背包中:

```

What do you want to do? (You are in the Room 1)
A.Talk to Merchant
B.Move
C.Show your status
D.Open bag/Equip
A
.....
Money is the power!
Merchant: Hi! I am Merchant. I have something for you to choose:
1. Holy sword cost$20.
2. Poison gas cost$15.
Merchant: Do you want to buy something? (Y/N)
Y
Merchant: Which one do you want?(please enter number only)
2
.....
What do you want to do? (You are in the Room 1)
A.Talk to Merchant
B.Move
C.Show your status
D.Open bag/Equip
D
.....
Your bag has:
(1) Poison gas
Which one to equip?(enter number only/0 to exit)

```

4.遇到 Chest

選擇打開寶箱則會獲得物品，若選擇打開已經開過的寶箱則不會有改變。

```

What do you want to do? (You are in the Room 2)
A.Open the chest
B.Move
C.Show your status
D.Open bag/Equip
A
You've got $30 coins!
.....
What do you want to do? (You are in the Room 2)
A.Open the chest
(It has already been opened.)
B.Move
C.Show your status
D.Open bag/Equip

```

```

.....
What do you want to do? (You are in the Room 2)
A.Open the chest
(It has already been opened.)
B.Move
C.Show your status
D.Open bag/Equip
A
.....

```

而寶箱內可能會有多個物品(如下圖):

```

A.Open the chest
B.Move
C.Show your status
D.Open bag/Equip
A
You've got a Shield
You've got $50 coins!
.....

```

5.遇到 Monster

(1)遇到怪物時，可以選擇攻擊或撤退，若選擇撤退:

```

.....
Warning!!Little devil is coming!
Which action do you want to choose?
A. Attack
B. Retreat
B
You retreat to the previous room.
.....
What do you want to do? (You are in the Room 4)
A.Talk to Merchant
B.Move
C.Show your status
D.Open bag/Equip

```

(2)若選擇攻擊，會詢問是否使用技能，以下為不使用技能，並打敗 monster 的示例，該房間的怪物被打敗後再次經過該房間便不會顯示怪物:

```

.....
What do you want to do? (You are in the Room 5)
A.Combat with Little devil
B.Show your status
C.Open bag/Equip
A
.....
Warning!!Little devil is coming!
Which action do you want to choose?
A. Attack
B. Retreat
A
Using skill? (Y/N)
N
Fighting start!.....
Little devil is attacked 10 points by you. Little devil lefts 10 health.
You are attacked 3 points by Little devil.
You left 97 health.
Little devil is attacked 10 points by you. Little devil lefts 0 health.
You beat Little devil.

```

```

.....
What do you want to do? (You are in the Room 5)
A.Move
B.Show your status
C.Open bag/Equip

```

計算被攻擊多少的算法為: (對方的 attack-自己的 defense)，因此:

-若自己的 defense>對方的 attack，會顯示”Miss”

```
Fighting start!.....
You miss!
```

-若兩方平手，則會請玩家退回前一房間

(3)若攻擊時選擇使用技能，在對戰過程中，attack、defense 值會增加，但在對戰完後便會恢復原狀。另外，打敗某些特定怪物後可獲得錢幣:

```
Warning!!Zombie is coming!
Which action do you want to choose?
A. Attack
B. Retreat
A
Using skill? (Y/N)
Y
Your skill is Freeze Spell

Fighting start!.....
Zombie is attacked 20 points by you. Zombie lefts 30 health.
You are attacked 10 points by Zombie.
You left 85 health.
Zombie is attacked 20 points by you. Zombie lefts 10 health.
You are attacked 10 points by Zombie.
You left 75 health.
Zombie is attacked 20 points by you. Zombie lefts 0 health.
You beat Zombie.
Here's some reward for you.($10)
```

```
[test]
Occupation: Magician
Health: 75/100
Attack: 35
Defense: 20
Money: $10
```

("Zombie", 50, 35, 25)，理論上對怪物造成的傷害應該是(35-25)但因為使用了技能 increaseStates(0,10,5)，所以傷害會增加。

(4)遊戲如何結束:

-當玩家死亡時(玩家輸)

```
Fighting start!.....
You miss!
You are attacked 20 points by Zombie.
You left 80 health.
You miss!
You are attacked 20 points by Zombie.
You left 60 health.
You miss!
You are attacked 20 points by Zombie.
You left 40 health.
You miss!
You are attacked 20 points by Zombie.
You left 20 health.
You miss!
You are attacked 20 points by Zombie.
You left 0 health.
YOU ARE DEAD!
GAME OVER.....
```

-當 BOSS 死亡時(玩家贏)

```
Warning!!BOSSYBOSS is coming!
Which action do you want to choose?
A. Attack
B. Retreat
A
Using skill? (Y/N)
Y
Your skill is Freeze Spell

Fighting start!.....
BOSSYBOSS is attacked 45 points by you. BOSSYBOSS lefts 115 health.
You are attacked 18 points by BOSSYBOSS.
You left 127 health.
BOSSYBOSS is attacked 45 points by you. BOSSYBOSS lefts 70 health.
You are attacked 18 points by BOSSYBOSS.
You left 109 health.
BOSSYBOSS is attacked 45 points by you. BOSSYBOSS lefts 25 health.
You are attacked 18 points by BOSSYBOSS.
You left 91 health.
BOSSYBOSS is attacked 45 points by you. BOSSYBOSS lefts 0 health.
You beat BOSSYBOSS.
CONGRATS-YOU WIN! THE GAME IS OVER.....
```

6.遇到 locked room

有的房間需要花費錢幣解鎖，若選擇’N’(如左上圖)，或選擇’Y’但錢幣不足(如左下圖)都無法解鎖，選擇’Y’且錢幣足夠才可解鎖(如右圖)。


```

.....
What do you want to do? (You are in the Room 3)
This room is locked. You have to pay $20 to unlock it.
Do you want to pay to unlocked it? (Y/N)
N
A.Move
B.Show your status
C.Open bag/Equip

```

```

.....
What do you want to do? (You are in the Room 3)
This room is locked. You have to pay $20 to unlock it.
Do you want to pay to unlocked it? (Y/N)
Y
Sorry, you don't have enough money.
A.Move
B.Show your status
C.Open bag/Equip

```

```

.....
What do you want to do? (You are in the Room 3)
This room is locked. You have to pay $20 to unlock it.
Do you want to pay to unlocked it? (Y/N)
Y
You now unlocked the room.
A.Open the chest
B.Move
C.Show your status
D.Open bag/Equip
A
You've got a Shield
You've got $50 coins!
.....
What do you want to do? (You are in the Room 3)
A.Open the chest
(I has already been opened.)
B.Move
C.Show your status
D.Open bag/Equip

```

7.成功實現的功能和特點:

- (1)玩家可以裝備物品，且裝備後物品會從背包消失。
- (2)玩家可以選擇不同職業，並選擇是否要使用技能，且使用技能時也會有相對應的代價，如:生命值減少。
- (3)除了和 NPC 交易外的另一種花錢方式:解鎖房間需要錢幣。
- (4)不同職業會有對應武器，使用時可以使武器原先增加的數值再增值。

四、Discussion

1. 問題、解決方式及可能的改進方法:

(1) 使用者輸入的包容性:

- I. 在程式運行過程中，當遇到請使用者輸入選擇的狀況，若使用者輸入的為 **wrong input**，程式可能會卡住，因為在程式碼中缺乏:若輸入值並非預期數值的 **if** 條件式。
- II. 例如:在一開始請玩家選擇職業時只能輸入 'A'、'B'、'C'，若輸入非這三個符號便會卡住，因此增加了 `else{cout<<"wrong input, enter again">>"<<endl;}`這行程式碼以避免此情形。
- III. 但在選擇要移動的方向的狀況，尚未寫若輸入是 **wrong input** 的條件式，因此使用者的輸入有受限，這部分可以再優化，且對於輸入值(如英文字母的大小寫)包容度可以再提高。

(2) 玩家在和 monster 對戰過程:

- I. 若遇到己方的 **attack** 值<對方的 **defense** 值的狀況，需另外獨立出來一個條件式，顯示"miss"，否則進入原本的計算條件式，會造成有負值出現，而 **health** 值反而會增加。
- II. 當遇到玩家和 **monster** 兩者都是互相 **miss** 的狀況下，因為分不出勝負會進入無限循環，因此也要另外獨立出一個條件式，讓玩家撤退回前一房間，去增加自己的攻擊力。

(3) triggerEvent(Object* thething):

- I. 在寫此函式時，因為傳入的參數 **type** 是 **object**，必須先使用 **dynamic_cast** 將其強制轉型成 **player**:
`Player*trade_player=dynamic_cast<Player*> (thething);`
 才方便寫後續的內容。

(4) 玩家 max health 的輸出顯示:

當顯示玩家狀態時，cur/max health 會同時出現，當遇到玩家使用物品增加 health，不管 cur health 狀態如何，max health 的數值也會一起增加，因此在設計方面有些不完善，應該改進為分成兩種狀況:

- I. 若 cur health > 100 則 max health 也會跟著改變，且值要和 cur health 相等，(如原本是 97/100，增加 15 health 後，應為 112/112)
- II. 若 cur health 又再次 < 100，max health 應該要跟著改變，(如原本是 112/112，減少 40 health 後，應為 72/100)

(5) Dungeon::handlemovement()的實作方法:

- I. handlemovement()要顯示可以去的方向，但每一個房間的選項個數都不同，若單純使用 vector，選項的 1.-4.和方向會對不起來，因此利用 direction[num[i]]的形式。
- II. code 實現方法:宣告一個<vector> num 和 array direction，即可自由變動 vector 的長度，當該 room 的上下左右 room 不是空的，就 push_back(0~3)，運用 for 迴圈 cout<< direction[num[i]];即可對應

(6) 將物品從 vector 中移除:

- I. 在 popObject()、popbag()等函式中，要將 vector 中的其中一項從 vector 中移除，因為之前沒有學習過如何將 vector 的特定一項移除，因此上網找尋相關方法。
- II. 利用 for 迴圈及 erase()、begin()函式達成:
this>objects.erase(objects.begin() + i);

(7) 玩家設計的優化:

- I. 裝備和背包是分開的概念:
一開始的設計是只要撿或買到 Item，Item 造成的效果會馬上反應在玩家狀態上，但後來發現這不太符合遊戲運作概念。因此後來改以先將蒐集到的 Item 放入背包，玩家隨時可以打開背包選擇要裝備的物品，而物品經裝備後才會對玩家造成效果，且會從背包消失。
- II. 技能只限在該次對戰後有作用:
玩家在與怪物對戰時可選擇是否使用技能，原先設計缺乏考慮技能的效果僅限在該次對戰有用，因此後來新增了 unuseskill()函式，使對戰結束後，玩家的數值會恢復原樣。

2. 程式概念的討論:

(1) 物件導向與函式導向的差異:

- I. 函式導向著重於程式中的函式及運算；而物件導向(OOP)著重於物件的封裝、繼承和多型等特性，且對於全域和私有的界定概念較為清楚，因此相較函式導向，OOP 使程式更為靈活並較易擴張。
- II. 對於遊戲設計方面，物件導向具有較大便利性和優勢。遊戲中包

含了各種不同的角色、道具、房間等元素，將其抽象成物件能夠更好地組織和管理，在設計時的思考也較直觀，另外，利用 OOP 的特性，在設計不同元素間的互動也較為輕鬆。

- III. 使用物件導向，讓每一個類別都各自獨立，撰寫程式碼時較容易發現錯誤所在。

(2) Virtual function:

- I. **Virtual function** 使同一個函式名稱可以對應不同的實作方法，使用 **Virtual function** 可以優化程式碼，減少程式碼的重複性，並提高程式碼的可讀性，在面對不同狀況時也可提高程式的靈活性。
- II. 在遊戲中，每個房間可能包含不同的物品或怪物，且會與玩家間有不同的行為互動，利用 **Virtual function** 可以為不同角色定義一個共同行為(如:triggerEvent())，但行為的具體內容會因角色類別有所不同。
- III. 在操作遊戲運行的 **Dungeon.cpp** 檔中，即可只利用一個函式，便可根據不同對象而有對應的效果，提升程式撰寫的效率。

(3) room 使用 linked list 連接:

- I. 使用 **linked list** 來連接房間，在地圖設計上能夠更加靈活調整，隨時增加、刪除或改變房間的佈置，且玩家在各房間移動的過程，也能透過 **linked list** 的指標指向更加方便找到對應的房間。

3. 心得:

這次的遊戲開發作業讓我更加熟悉 C++物件導向設計的實作方式，並且學習到如何運用繼承、**virtual function**、指標、**vector** 等工具來設計一個地牢遊戲。在這個過程中，我對物件導向整體架構有更深入的了解，並且能夠將自己的思考轉化成程式碼，例如遊戲運作、戰鬥系統、角色設計等等。這樣的設計過程讓我感受到了寫程式的樂趣，雖然在過程中也遇到了許多困難，但最後看到自己設計的遊戲能成功運行，獲得了很大的成就感。

五、Conclusion

- 1. 這次的地牢遊戲設計，讓玩家能夠選擇不同職業，並藉由與 NPC、怪物及寶箱的互動，提升攻擊及防禦力，並獲取錢幣以購買物品，最終要能打敗 BOSS 怪物才算勝利。
- 2. 在設計過程有發現一些問題，例如: 和怪物對戰時的條件式、從 **vector** 中移除物品、玩家移動選項顯示方法等，經過一番研究有解決，而最終實作結果，程式也能夠順利運行，但有些細節設計還可以更加完善，例如: 使用者的輸入、玩家狀態的顯示等。
- 3. 未來還有些可以改進的方向:
 - (1) 對戰前，可以先顯示怪物的狀態。
 - (2) 當玩家的背包有多個物品時，可以選擇一次配備兩個以上，而不用一

次一次分開選擇配備。

- (3) 創建地圖的方式，可以不要一個一個刻房間的物品，或許可考慮採用隨機的機制。
- (4) 程式碼中所有的 `get function` 可以變為 `const` 形式會更好。
- (5) 若能新增 `Record system`，會有更好的遊戲體驗。