

## CSED232 Object Oriented Programming (Spring 2024)

### Assignment #4

#### Working with Classes

Due: 2024년 4월 26일 금요일 23시 59분 59초

담당 조교: 양희민 ([heeminid@postech.ac.kr](mailto:heeminid@postech.ac.kr))

#### 주의 사항

- 클래스는 .h 파일과 .cpp 파일에 나눠서 구현해야 합니다.
- 문제에 따로 명시되어 있지 않아도 클래스의 생성자 (constructor), 소멸자 (destructor)는 필수적으로 구현되어야 합니다.
- 문법 사용에 제약은 없습니다.
- 문제의 구현 요구 사항을 꼼꼼히 읽어 보시기 바랍니다.

#### 감점 사항

- 제출 기한을 넘길 시 20% 감점
  - 24시간이 경과할 때마다 추가 20% 감점
  - 제출 기한 0시간 ~ 24시간 후: 20% 감점, 24시간 ~ 28시간 후: 40% 감점, ...
- 컴파일이 되지 않으면 0점 처리
- 제출 방법 위반 시 감점

#### 제출 방법

- Assignment #1에 첨부된 개발 환경 설정 파일을 참고하여 제출해 주십시오.
- 소스코드의 폴더명을 학번 (20240000)으로 설정해 주시고 report의 파일명을 학번\_report.pdf (20240000\_report.pdf)로 설정하여 같이 압축 후 제출해 주십시오. 압축파일명 역시 학번.zip로 설정 부탁드립니다.
- 문제에 명시되어 있지 않다면 파일명은 자유롭게 설정 가능합니다.

## 공통 채점 기준

### 1. 프로그램 기능

- 프로그램이 요구 사항을 모두 만족하면서 올바르게 실행되는가?

### 2. 프로그램 설계 및 구현

- 요구 사항을 만족하기 위한 변수 및 알고리즘 설계가 잘 되었는가?
- 문제에서 제시된 세부 조건을 모두 만족하였는가?
- 설계된 내용이 요구된 언어를 이용하여 적절히 구현되었는가?

### 3. 프로그램 가독성

- 프로그램이 읽기 쉽고 이해하기 쉽게 작성되었는가?
- 변수명이 무엇을 의미하는지 이해하기 쉬운가?
- 프로그램의 소스 코드를 이해하기 쉽도록 주석을 잘 붙였는가?

### 4. 보고서 구성 및 내용, 양식

- 보고서는 적절한 내용으로 이해하기 쉽고 보기 좋게 잘 작성되었는가?
- 보고서의 양식을 잘 따랐는가?

다른 사람의 프로그램 혹은 인터넷에 있는 프로그램을 복사(copy)하거나 간단히 수정해서 제출하면 학점은 무조건 'F'가 됩니다. 이러한 부정행위가 발견되면 학과에서 정한 기준에 따라 추가적인 불이익이 있을 수 있습니다.

# Text-based RPG

## 1. 과제 개요

이번 과제는 C++ class를 사용하여 게임에 존재하는 다양한 객체들을 관리하는 것을 목표로 한다. 이를 위해, 다양한 객체들을 관리할 수 있는 텍스트 기반의 RPG를 제작해보자.

## 2. 과제 설명

본 과제에서 구현하고자 하는 게임에서 플레이어는 캐릭터를 성장시키며 보스 몬스터를 잡는 것을 목표로 한다. 아래 설명을 읽고 각각의 요소를 구현해 보자. 명시되지 않은 멤버 변수 및 메서드, 혹은 명시되지 않은 클래스도 추가 구현 가능하며, 이 경우 보고서에 반드시 적도록 한다.

### 2.1 구현해야 하는 클래스

#### [class Character]

Character 클래스는 플레이어가 조종하는 캐릭터에 대한 정보를 관리한다. 해당 클래스가 관리해야 하는 정보는 다음과 같으며, 괄호 안의 값은 캐릭터 생성 시 기본값이다.

- int attack: 캐릭터의 공격력 (10)
- int defense: 캐릭터의 방어력 (2)
- int gold: 캐릭터가 소지하고 있는 골드의 양 (0)
- int level: 캐릭터의 현재 레벨 (1)
- int max\_hp: 캐릭터의 최대 HP (100)
- int curr\_hp: 캐릭터의 현재 HP (100)
- int max\_mp: 캐릭터의 최대 MP (20)
- int curr\_mp: 캐릭터의 현재 MP (20)
- int max\_exp: 다음 레벨까지 필요한 총 경험치 (EXP) (100)
- int curr\_exp: 캐릭터의 현재 EXP (0)

캐릭터가 할 수 있는 행동은 기본 공격, 스킬 사용, 포션 사용이다.

- attack\_basic: 캐릭터가 몬스터에게 기본 공격을 가한다. 해당 공격 이후, 몬스터의 HP는 (캐릭터의 공격력) - (몬스터의 방어력) 만큼 감소한다.

- attack\_skill: 캐릭터가 몬스터에게 단일 공격 스킬로 공격을 가한다. 해당 공격 이후, 몬스터의 HP는 (캐릭터의 공격력 \* 1.5) - (몬스터의 방어력) 만큼 감소하며, 캐릭터의 MP는 15만큼 감소한다.

- attack\_fire: 캐릭터가 몬스터에게 불 속성 공격 스킬로 공격을 가한다. 해당 공격 이후, 불 속성 몬스터의 HP는 (캐릭터의 공격력 \* 2) - (몬스터의 방어력) 만큼 감소하며, 불 속성 이외의 몬스터의 HP는 (캐릭터의 공격력) - (몬스터의 방어력) 만큼 감소한다. 캐릭터의 MP는 15만큼 감소한다.

- attack\_grass: 캐릭터가 몬스터에게 풀 속성 공격 스킬로 공격을 가한다. 해당 공격 이후, 물 속성 몬스터의 HP는 (캐릭터의 공격력 \* 2) - (몬스터의 방어력) 만큼 감소하며, 물 속성 이외의 몬스터의 HP는 (캐릭터의 공격력) - (몬스터의 방어력) 만큼 감소한다. 캐릭터의 MP는 15만큼 감소한다.

- attack\_water: 캐릭터가 몬스터에게 물 속성 공격 스킬로 공격을 가한다. 해당 공격 이후, 불 속성 몬스터의 HP는 (캐릭터의 공격력 \* 2) - (몬스터의 방어력) 만큼 감소하며, 불 속성 이외의 몬스터의 HP는 (캐릭터의 공격력) - (몬스터의 방어력) 만큼 감소한다. 캐릭터의 MP는 15만큼 감소한다.

## **[class Map]**

Map 클래스는 격자 형태 맵의 정보를 관리한다. 본 게임에서 맵은 정사각형의 격자 형태다. 맵 크기는 8 \* 8 격자로 설정하며, 격자의 내용은 모두가 동일하게 초기화하도록 한다. 게임 시작 시 맵을 초기화하는 방법은 후술한다.

## **[class Monster]**

Monster 클래스는 몬스터의 정보를 관리한다. Monster 객체는 플레이어가 MONSTER 칸에 진입했을 때 생성되어 플레이어와 전투를 진행한다. Monster 클래스는 몬스터의 HP, 공격력, 방어력, 속성 등의 정보를 관리한다. 몬스터의 속성은 풀 속성, 물 속성, 불 속성, 속성 없음의 4개 중 하나로 랜덤하게 정해진다.

## **[class Shop]**

Shop 클래스는 맵에 존재하는 상점의 정보를 관리한다. 맵에 존재하는 상점은 재고가 있는 아이템을 보유하고 있으며, 캐릭터는 상점에 진입했을 때 골드를 지불하고 해당 아이템을 구매할 수 있다. 상점에서 관리하는 아이템은 다음과 같다.

- Attack +10 (700 골드, 구매 시 attack 10 증가, 재고 1)
- Defense +5 (700 골드, 구매 시 defense 5 증가, 재고 1)
- Exp +300 (700 골드, 구매 시 exp 300 증가, 재고 1)
- HP 전부 회복 (500 골드, 구매 시 HP 전부 회복, 재고 무한)
- MP 전부 회복 (500 골드, 구매 시 MP 전부 회복, 재고 무한)

## 2.2 게임 진행

프로그램을 실행하면 아래와 같은 화면이 출력된다.

```
=====
=====
#####
# ##### # # ##### # # # # #
# # # # # # # # #
# ##### ## # ##### ##### # ####
# # ## # # # # # #
# # # # # # # # #
# ##### # # # # #####

=====
=====
----- Menu -----
1. Start game
2. Exit
> 
```

아스키 아트 출력에 사용된 코드는 아래와 같다.

```
std::cout << "===== " << std::endl;
std::cout << "===== " << std::endl;
std::cout << "===== " << std::endl;
std::cout << " #####          #####  #####  " << std::endl;
std::cout << "  # ##### #  # #####  #  # #  # #  # " << std::endl;
std::cout << "  # #      # # #      #  # # #  # #      " << std::endl;
std::cout << "  # #####  ## #      #####  #####  #  #### " << std::endl;
std::cout << "  # #      ## #      #  # # #      #  # " << std::endl;
std::cout << "  # #      # # #      #  # # #      #  # " << std::endl;
std::cout << "  # ##### #  # #      #  # #      ##### " << std::endl;
std::cout << "===== " << std::endl;
std::cout << "===== " << std::endl;
std::cout << "===== " << std::endl;
```

해당 화면이 출력되고 나서, 프로그램은 사용자가 두 가지 메뉴 중 하나를 선택할 때까지 기다린다. 1과 2 외의 옵션이 입력될 경우 다시 입력을 받는다. 정수 이외의 것은 입력되지 않는다고 가정한다.

```
=====
=====
#####
# ##### # # ##### # # # # #
# # # # # # # # # #
# ##### ## # ##### # #
# # ## # # # # #
# # # # # # # # #
# ##### # # # # #
#####

----- Menu -----
1. Start game
2. Exit
> 3
> 5
>
```

2. Exit 옵션이 선택됐을 경우 프로그램을 아래와 같이 종료한다.

```
=====
=====
#####
# ##### # # ##### # # # # #
# # # # # # # # # #
# ##### ## # ##### # #
# # ## # # # # #
# # # # # # # # #
# ##### # # # # #
#####

----- Menu -----
1. Start game
2. Exit
> 3
> 5
> 2
Exit...
```

1. Start game 옵션이 선택됐을 경우 게임이 시작된다. 게임 중에는 기본적으로 아래와 같은 화면을 출력한다.

```
Game start!



|     |   |   |   |   |   |   |   |
|-----|---|---|---|---|---|---|---|
| You | ? | ? | ? | ? | ? | ? | ? |
| ?   | ? | ? | ? | ? | ? | ? | ? |
| ?   | ? | ? | ? | ? | ? | ? | ? |
| ?   | ? | ? | ? | ? | ? | ? | ? |
| ?   | ? | ? | ? | ? | ? | ? | ? |
| ?   | ? | ? | ? | ? | ? | ? | ? |
| ?   | ? | ? | ? | ? | ? | ? | ? |
| ?   | ? | ? | ? | ? | ? | ? | ? |



1. Choose direction
2. Show character status
3. Exit game
> 
```

캐릭터가 위치한 칸에는 “You”를 표시하며, 아직 방문하지 않은 칸은 “?”로 표시한다. 이 화면에서 프로그램은 사용자가 세 가지 메뉴 중 하나를 선택할 때까지 기다린다. 3번 옵션을 선택했을 경우 상술한 것과 똑같이 프로그램을 종료한다. 2번 옵션을 선택했을 경우 아래처럼 캐릭터의 현재 상태를 출력한 후 맵과 옵션 선택 창을 다시 출력한다.

```
1. Choose direction
2. Show character status
3. Exit game
> 2

Level: 1
Attack: 10
Defense: 2
Gold: 0
HP: 100 / 100
MP: 20 / 20
EXP: 0 / 100



|     |   |   |   |   |   |   |   |
|-----|---|---|---|---|---|---|---|
| You | ? | ? | ? | ? | ? | ? | ? |
| ?   | ? | ? | ? | ? | ? | ? | ? |
| ?   | ? | ? | ? | ? | ? | ? | ? |
| ?   | ? | ? | ? | ? | ? | ? | ? |
| ?   | ? | ? | ? | ? | ? | ? | ? |
| ?   | ? | ? | ? | ? | ? | ? | ? |
| ?   | ? | ? | ? | ? | ? | ? | ? |
| ?   | ? | ? | ? | ? | ? | ? | ? |



1. Choose direction
2. Show character status
3. Exit game
> 
```



1번 옵션을 선택했을 경우 아래와 같은 화면이 출력된다.

```

You ? ? ? ? ? ? ?
? ? ? ? ? ? ? ?
? ? ? ? ? ? ? ?
? ? ? ? ? ? ? ?
? ? ? ? ? ? ? ?
? ? ? ? ? ? ? ?
? ? ? ? ? ? ? ?
? ? ? ? ? ? ? ?
? ? ? ? ? ? ? ?

1. Choose direction
2. Show character status
3. Exit game
> 1
Where do you want to go?
> 
```

이 화면에서 유저는 캐릭터가 움직일 방향을 선택한다. 캐릭터는 (w: 위쪽), (a: 왼쪽), (s: 아래쪽), (d: 오른쪽)으로 이동한다. 이 외의 문자가 입력될 경우 상술한 것과 같이 입력을 다시 받으며, 단일 문자 이외의 것은 입력되지 않는다고 가정한다.

이동할 수 없는 방향을 입력했을 경우 메시지를 출력하고 입력을 다시 받는다.

```

You ? ? ? ? ? ? ?
? ? ? ? ? ? ? ?
? ? ? ? ? ? ? ?
? ? ? ? ? ? ? ?
? ? ? ? ? ? ? ?
? ? ? ? ? ? ? ?
? ? ? ? ? ? ? ?
? ? ? ? ? ? ? ?
? ? ? ? ? ? ? ?

1. Choose direction
2. Show character status
3. Exit game
> 1
Where do you want to go?
> w
You cannot move to that direction.
> 
```

올바른 방향을 입력했을 경우 캐릭터를 이동시키며 맵을 다시 출력한다. 이 경우, 이동한 칸이 어디인지에 따라 다른 행동을 취한다.

- 이동한 칸이 EMPTY일 경우

```
Where do you want to go?
> s


|     |   |   |   |   |   |   |   |
|-----|---|---|---|---|---|---|---|
|     | ? | ? | ? | ? | ? | ? | ? |
| You | ? | ? | ? | ? | ? | ? | ? |
| ?   | ? | ? | ? | ? | ? | ? | ? |
| ?   | ? | ? | ? | ? | ? | ? | ? |
| ?   | ? | ? | ? | ? | ? | ? | ? |
| ?   | ? | ? | ? | ? | ? | ? | ? |
| ?   | ? | ? | ? | ? | ? | ? | ? |
| ?   | ? | ? | ? | ? | ? | ? | ? |


1. Choose direction
2. Show character status
3. Exit game
> 
```

캐릭터를 이동시키고 인게임 메뉴 입력 창으로 되돌아간다.

- 이동한 칸이 RANDOM일 경우

- 25% 확률로 현재 HP 30% 감소 (소수점 있을 경우 반올림)
- 25% 확률로 현재 HP 30% 회복 (소수점 있을 경우 반올림)
- 25% 확률로 100 골드 획득
- 25% 확률로 아무 일도 일어나지 않음

난수 생성에는 cstdlib 헤더에 있는 rand() 함수를 활용하도록 한다.

(<https://cplusplus.com/reference/cstdlib/rand/>)

```
Random encounter!

You lost HP...



|     |   |   |   |   |   |   |   |
|-----|---|---|---|---|---|---|---|
|     | ? | ? | ? | ? | ? | ? | ? |
|     | ? | ? | ? | ? | ? | ? | ? |
| You | ? | ? | ? | ? | ? | ? | ? |
| ?   | ? | ? | ? | ? | ? | ? | ? |
| ?   | ? | ? | ? | ? | ? | ? | ? |
| ?   | ? | ? | ? | ? | ? | ? | ? |
| ?   | ? | ? | ? | ? | ? | ? | ? |
| ?   | ? | ? | ? | ? | ? | ? | ? |



1. Choose direction
2. Show character status
3. Exit game
> 2

Level: 1
Attack: 10
Defense: 2
Gold: 0
HP: 70 / 100
MP: 20 / 20
EXP: 0 / 100



|     |   |   |   |   |   |   |   |
|-----|---|---|---|---|---|---|---|
|     | ? | ? | ? | ? | ? | ? | ? |
|     | ? | ? | ? | ? | ? | ? | ? |
| You | ? | ? | ? | ? | ? | ? | ? |
| ?   | ? | ? | ? | ? | ? | ? | ? |
| ?   | ? | ? | ? | ? | ? | ? | ? |
| ?   | ? | ? | ? | ? | ? | ? | ? |
| ?   | ? | ? | ? | ? | ? | ? | ? |
| ?   | ? | ? | ? | ? | ? | ? | ? |



1. Choose direction
2. Show character status
3. Exit game
> █
```

HP를 잃었을 경우 (Status 출력 시 HP가 감소된 것을 확인할 수 있음. 회복된 경우는 반대이며, "You restored HP" 출력)

```
Random encounter!

You found 100 gold!



|     |   |   |   |   |   |   |   |
|-----|---|---|---|---|---|---|---|
|     | ? | ? | ? | ? | ? | ? | ? |
|     | ? | ? | ? | ? | ? | ? | ? |
| You | ? | ? | ? | ? | ? | ? | ? |
| ?   | ? | ? | ? | ? | ? | ? | ? |
| ?   | ? | ? | ? | ? | ? | ? | ? |
| ?   | ? | ? | ? | ? | ? | ? | ? |
| ?   | ? | ? | ? | ? | ? | ? | ? |
| ?   | ? | ? | ? | ? | ? | ? | ? |



1. Choose direction
2. Show character status
3. Exit game
> 2

Level: 1
Attack: 10
Defense: 2
Gold: 100
HP: 100 / 100
MP: 20 / 20
EXP: 0 / 100



|     |   |   |   |   |   |   |   |
|-----|---|---|---|---|---|---|---|
|     | ? | ? | ? | ? | ? | ? | ? |
|     | ? | ? | ? | ? | ? | ? | ? |
| You | ? | ? | ? | ? | ? | ? | ? |
| ?   | ? | ? | ? | ? | ? | ? | ? |
| ?   | ? | ? | ? | ? | ? | ? | ? |
| ?   | ? | ? | ? | ? | ? | ? | ? |
| ?   | ? | ? | ? | ? | ? | ? | ? |
| ?   | ? | ? | ? | ? | ? | ? | ? |



1. Choose direction
2. Show character status
3. Exit game
> 
```

100 골드를 획득했을 경우 (Status 출력 시 골드가 증가한 것을 확인할 수 있음)

```
Random encounter!

Nothing happened...



|     |   |   |   |   |   |   |   |
|-----|---|---|---|---|---|---|---|
|     | ? | ? | ? | ? | ? | ? | ? |
|     | ? | ? | ? | ? | ? | ? | ? |
| You | ? | ? | ? | ? | ? | ? | ? |
| ?   | ? | ? | ? | ? | ? | ? | ? |
| ?   | ? | ? | ? | ? | ? | ? | ? |
| ?   | ? | ? | ? | ? | ? | ? | ? |
| ?   | ? | ? | ? | ? | ? | ? | ? |
| ?   | ? | ? | ? | ? | ? | ? | ? |



1. Choose direction
2. Show character status
3. Exit game
> 
```

아무 일도 일어나지 않았을 경우

- 이동한 칸이 MONSTER일 경우

몬스터를 마주할 경우 몬스터 객체가 생성되며, 아래와 같이 캐릭터의 상태창과 몬스터의 상태창이 출력된다. 그 후, 캐릭터가 취할 수 있는 액션 리스트를 보여주면서 액션의 입력을 기다린다. 이번에도 역시 범위 이외의 입력이 들어올 경우 무시하고 다시 입력을 기다린다. 이 때, 액션의 입력으로는 숫자만 들어온다고 가정한다.

```
Wild monster appeared!

=====
Character status
Level: 1
Attack: 10
Defense: 2
HP: 100 / 100
MP: 20 / 20
EXP: 0 / 100
=====
Monster status
Attack: 11
Defense: 3
HP: 65 / 65
=====
Choose your action
1. Meele attack
2. Magic attack
3. Fire magic attack
4. Grass magic attack
5. Water magic attack
> 
```

몬스터를 마주치면 해당 몬스터의 difficulty가 결정된다. difficulty의 계산 공식은 캐릭터의 시작 위치와의 Manhattan distance다. 즉, 캐릭터의 생성 위치가 (0, 0)이고 몬스터를 마주친 위치가 (x, y)라면 difficulty는  $x+y$ 로 계산된다. 이 difficulty에 따라 몬스터 객체의 스펙을 다음과 같이 정한다.

- $HP: 50 + (difficulty * 5)$
- $Attack: 5 + (difficulty * 2)$
- $Defense: 2 + (difficulty / 2)$
- $Reward: (100 + difficulty * 10) \text{ gold}$   
 $(100 + difficulty * 10) \text{ EXP}$

몬스터와 전투 시 행동은 플레이어가 먼저 취하며, 그 뒤로는 몬스터의 공격과 플레이어의 행동이 번갈아 가면서 이뤄진다.

```
Choose your action
1. Meele attack
2. Magic attack
3. Fire magic attack
4. Grass magic attack
5. Water magic attack
> 1

You hit the monster with 7 damage!
Monster hit you with 9 damage!

=====
Character status
Level: 1
Attack: 10
Defense: 2
HP: 91 / 100
MP: 20 / 20
EXP: 0 / 100
=====
Monster status
Attack: 11
Defense: 3
HP: 58 / 65
=====
Choose your action
1. Meele attack
2. Magic attack
3. Fire magic attack
4. Grass magic attack
5. Water magic attack
> █
```

MP를 소모하는 스킬을 사용했을 때 MP가 부족했을 경우 캐릭터는 몬스터에게 공격을 가하지 못하고 몬스터만 캐릭터에게 공격을 가한다. 아래의 사진이 그 예시다.

```
Choose your action
1. Meele attack
2. Magic attack
3. Fire magic attack
4. Grass magic attack
5. Water magic attack
> 2

You hit the monster with 12 damage!
Monster hit you with 9 damage!

=====
Character status
Level: 1
Attack: 10
Defense: 2
HP: 91 / 100
MP: 5 / 20
EXP: 0 / 100
=====
Monster status
Attack: 11
Defense: 3
HP: 53 / 65
=====
Choose your action
1. Meele attack
2. Magic attack
3. Fire magic attack
4. Grass magic attack
5. Water magic attack
> 2

Not enough MP...
Monster hit you with 9 damage!

=====
Character status
Level: 1
Attack: 10
Defense: 2
HP: 82 / 100
MP: 5 / 20
EXP: 0 / 100
=====
Monster status
Attack: 11
Defense: 3
HP: 53 / 65
=====
Choose your action
1. Meele attack
2. Magic attack
3. Fire magic attack
4. Grass magic attack
5. Water magic attack
> 
```



스킬 사용 시 데미지 계산 공식은 상술한 스킬 설명을 따른다.

전투 과정에서 캐릭터의 체력이 0보다 낮아지면 캐릭터가 사망하며 게임이 종료된다.

```
=====
Character status
Level: 1
Attack: 10
Defense: 2
HP: 1 / 100
MP: 5 / 20
EXP: 0 / 100
=====
Monster status
Attack: 11
Defense: 3
HP: 53 / 65
=====
Choose your action
1. Meele attack
2. Magic attack
3. Fire magic attack
4. Grass magic attack
5. Water magic attack
> 2

Not enough MP...
Monster hit you with 9 damage!

You died...
```

전투 과정에서 몬스터의 체력이 0보다 낮아지면 몬스터가 사망하며 캐릭터에게 보상으로 골드와 경험치가 지급된다. 이 때, 캐릭터의 현재 경험치에 보상 경험치를 더한 것이 캐릭터의 최대 경험치량을 초과한다면 캐릭터의 레벨이 상승한다. 레벨이 상승하면 최대 경험치량은 (캐릭터의 레벨 \* 100)으로 변경되며, 공격력은 3만큼 상승하고 방어력은 1만큼 상승한다. 또한, 최대 HP와 최대 MP는 각각 50과 10만큼 상승하며 최대치로 회복한다.

```
You hit the monster with 7 damage!
Monster hit you with 9 damage!

You defeated the monster!
Reward: 130 Gold, 130 EXP

Level up!
=====
Character status
Level: 1 => 2
Attack: 10 => 13
Defense: 2 => 3
HP: 19 / 100 => 150 / 150
MP: 5 / 20 => 30 / 30
EXP: 130 / 100 => 30 / 200
=====


|     |   |   |   |   |   |   |   |
|-----|---|---|---|---|---|---|---|
|     | ? | ? | ? | ? | ? | ? | ? |
|     | ? | ? | ? | ? | ? | ? | ? |
|     | ? | ? | ? | ? | ? | ? | ? |
| You | ? | ? | ? | ? | ? | ? | ? |
| ?   | ? | ? | ? | ? | ? | ? | ? |
| ?   | ? | ? | ? | ? | ? | ? | ? |
| ?   | ? | ? | ? | ? | ? | ? | ? |
| ?   | ? | ? | ? | ? | ? | ? | ? |


1. Choose direction
2. Show character status
3. Exit game
> 
```

- 이동한 칸이 SHOP일 경우

상점에 진입했을 경우 아래와 같이 캐릭터가 보유 중인 골드, 상점의 아이템 리스트와 재고를 출력하며, 유저의 선택을 기다린다. 이 때, 정수 이외의 입력은 들어오지 않는다고 가정한다.

```
Welcome to the item shop!
You have 500 gold.

=====
===== Item list =====
=====

1. Attack +10 (700 Gold) (Available)
2. Defense +5 (700 Gold) (Available)
3. Exp +300 (700 Gold) (Available)
4. Restore HP (500 Gold) (Always available)
5. Restore MP (500 Gold) (Always available)
6. Exit shop

=====
> 
```

6번을 선택할 경우 상점을 나간다. (2행 1열이 상점의 위치)

```
Welcome to the item shop!
You have 500 gold.

=====
===== Item list =====
=====

1. Attack +10 (700 Gold) (Available)
2. Defense +5 (700 Gold) (Available)
3. Exp +300 (700 Gold) (Available)
4. Restore HP (500 Gold) (Always available)
5. Restore MP (500 Gold) (Always available)
6. Exit shop

=====
> 6



|     |   |   |   |   |   |   |   |
|-----|---|---|---|---|---|---|---|
|     | ? | ? | ? | ? | ? | ? | ? |
| You | ? | ? | ? | ? | ? | ? | ? |
| ?   | ? | ? | ? | ? | ? | ? | ? |
| ?   | ? | ? | ? | ? | ? | ? | ? |
| ?   | ? | ? | ? | ? | ? | ? | ? |
| ?   | ? | ? | ? | ? | ? | ? | ? |
| ?   | ? | ? | ? | ? | ? | ? | ? |
| ?   | ? | ? | ? | ? | ? | ? | ? |



1. Choose direction
2. Show character status
3. Exit game
> 
```

재고가 없는 아이템을 선택할 경우 에러 메시지를 출력하고 입력을 다시 기다린다.

```
Welcome to the item shop!
You have 500 gold.

=====
===== Item list =====
=====

1. Attack +10 (700 Gold) (Sold out)
2. Defense +5 (700 Gold) (Available)
3. Exp +300 (700 Gold) (Available)
4. Restore HP (500 Gold) (Always available)
5. Restore MP (500 Gold) (Always available)
6. Exit shop

=====

> 1
Selected item is not available...

> 
```

재고가 있는 아이템을 선택했지만 골드가 부족할 경우 에러 메시지를 출력하고 입력을 다시 기다린다.

```
Welcome to the item shop!
You have 500 gold.

=====
===== Item list =====
=====

1. Attack +10 (700 Gold) (Sold out)
2. Defense +5 (700 Gold) (Available)
3. Exp +300 (700 Gold) (Available)
4. Restore HP (500 Gold) (Always available)
5. Restore MP (500 Gold) (Always available)
6. Exit shop

=====

> 2
You do not have enough gold...

> 
```

재고가 존재하고 보유 골드도 충분할 경우 아이템이 구매된다. 재고가 하나뿐인 아이템을 구매했을 경우 재고도 감소한다.

```
Welcome to the item shop!
You have 1000 gold.

=====
===== Item list =====
=====

1. Attack +10 (700 Gold) (Available)
2. Defense +5 (700 Gold) (Available)
3. Exp +300 (700 Gold) (Available)
4. Restore HP (500 Gold) (Always available)
5. Restore MP (500 Gold) (Always available)
6. Exit shop

=====
> 1
```

```
You bought an item!

=====
Character status
Level: 1
Attack: 10 => 20
Defense: 2
HP: 100 / 100
MP: 20 / 20
EXP: 0 / 100
=====
Shop item list
1. Attack +10 (700 Gold) (Available) => (Sold out)
2. Defense +5 (700 Gold) (Available)
3. Exp +300 (700 Gold) (Available)
4. Restore HP (500 Gold) (Always available)
5. Restore MP (500 Gold) (Always available)
=====
```

	?	?	?	?	?	?	?
You	?	?	?	?	?	?	?
?	?	?	?	?	?	?	?
?	?	?	?	?	?	?	?
?	?	?	?	?	?	?	?
?	?	?	?	?	?	?	?
?	?	?	?	?	?	?	?
?	?	?	?	?	?	?	?

```
1. Choose direction
2. Show character status
3. Exit game
> 
```

- 이동한 칸이 BOSS일 경우

보스 몬스터와 조우했을 경우 일반 몬스터를 조우한 것과 마찬가지로 전투에 진입한다. 보스 몬스터는 기존 몬스터보다 강력하며, 다음의 스펙으로 고정한다. 보스 몬스터의 타입 역시 랜덤으로 결정한다.

- HP: 200

- Attack: 30

- Defense: 10

보스 몬스터를 물리쳤을 경우 게임이 클리어 되며, 승리 메시지를 출력하고 프로그램을 종료한다. 승리 메시지는 자유롭게 설정 가능하다.

## 2.3. 안내 사항

- 게임의 난이도가 너무 쉽거나 어려울 경우 밸런스를 조정할 수 있습니다. 이 경우, 몬스터의 스펙이나 캐릭터의 스펙을 어떻게 조정했는지 보고서에 명시해 주십시오.
- Class 설명에 적어놓은 메서드 및 변수는 예시입니다. 자유롭게 설정할 수 있으며, class 디자인에 대한 내용 역시 보고서에 명시해 주시기 바랍니다.
- Class의 선언은 헤더 파일 (.h)에 해주시고 상세한 구현은 소스 파일 (.cpp)에 해주시기 바랍니다. 또한, Shop 클래스를 구현할 경우 내용을 shop.h 파일 및 shop.cpp 파일에 구현해 주십시오. 다른 클래스들 또한 마찬가지입니다. (Abcd 클래스를 구현할 경우 abcd.h 파일 및 abcd.cpp 파일에 구현)
- 맵은 같이 첨부된 map\_info.txt 를 참조하여 초기화 하시기 바랍니다.
- 클래스 디자인을 결정하신 이유도 함께 기술해 주시기 바랍니다.
  - 맵을 어떻게 디자인 했는지, 내부 변수는 왜 이것들을 넣었는지 등

## Q&A 내용 정리

2024.04.16 -----

Q. vector, algorithm, cmath 등의 라이브러리 사용 가능한가요?

A. 사용 가능합니다.

Q. 한 번 방문한 칸에 다시 방문할 경우 어떻게 되나요?

A. SHOP을 제외하고는 한 번 방문한 칸은 비어있는 칸으로 바뀝니다. SHOP의 경우 재방문할 경우 이전과 재고가 똑같이 유지되며, 구매한 아이템은 재고가 없음으로 처리됩니다.

Q. 몬스터가 공격하는 공식은 어떻게 되나요?

A. (몬스터의 공격력) - (캐릭터의 방어력) 만큼의 대미지가 캐릭터에게 가해집니다.

Q. 공격 당하는 측의 방어력이 공격하는 측의 공격력보다 높으면 대미지를 어떻게 처리하나요?

A. 대미지를 0으로 처리 부탁드립니다.

Q. map\_info.txt. 파일에 I와 O라는 알파벳이 있는데 무엇을 의미하나요?

A. 오타입니다. 잘못 업로드 드린 점 죄송하며, 새로운 파일을 업로드 했으니 확인 부탁드립니다.

Q. 아스키 아트가 깨져서 보입니다.

A. 아스키 아트는 사용하지 않고 간단한 텍스트로 대체하셔도 무방합니다.

Q. 랜덤 칸에서 HP 30% 감소는 최대 HP에 대해서인가요, 현재 HP에 대해서인가요?

A. 현재 HP에 대해서이며 HP는 반올림 부탁드립니다.

Q. rand()의 seed가 고정되어 항상 같은 값이 나옵니다.

A. srand(time(NULL)); 로 시드 설정 부탁드립니다.



Q. 상점은 여러 개 존재할 수 있나요?

A. 상점은 한 곳만 존재한다고 가정합니다.

Q. Start game 옵션 선택 후 1 2 3 이외의 정수가 입력될 경우 어떻게 하나요?

A. 정수 옵션을 선택하는 모든 경우에서 선택지 이외의 정수가 입력될 경우 입력을 다시 받도록 구현해 주십시오. 정수 이외의 옵션은 입력되지 않는다고 가정합니다.

Q. map은 어떻게 초기화하나요?

A. map\_info.txt와 동일하게 초기화 부탁드립니다.

Q. 플레이어의 공격으로 몬스터가 처치되는 상황에서도 플레이어의 공격 이후에 몬스터의 공격이 이어집니다. 이 공격으로 둘 모두의 체력이 0 이하가 되면 어떻게 되나요?

A. 캐릭터의 HP가 0 이하가 되었으므로 캐릭터는 사망하며 게임이 종료됩니다.