

## 1. 서론

1. 프로젝트 목적 및 배경 : 7주차까지 배운 내용에 대한 실습을 위해 진행
2. 목표 : 간단한 Mud 게임 구현

## 2. 요구사항

1. 사용자 요구사항 : 유저가 상하좌우로만 이동하며 목적지에 도착하는 게임
2. 기능 계획
  - 유저는 체력 20을 가지고 게임 시작
  - 처음 명령문을 입력 받을 때마다 HP 함께 출력
  - 사용자가 이동할 때마다 사용자 체력 1씩 감소
  - HP가 0이 되면 "실패"를 출력하고 종료
  - 아이템, 포션, 적을 만났을 때 그에 대한 메시지를 출력(적을 만날 경우 HP가 2가 줄어 들고 포션을 만날 경우 HP가 2가 늘어남)
3. 함수 계획
  - void checkState(int map[][mapX],int user\_x,int user\_y) 함수 생성 후 호출

## 3. 설계 및 구현

```
// 유저의 체력을 저장할 변수
int hp=20;
```

```
cout << "현재 HP: "<<hp;
cout<<" 명령어를 입력하세요 (up,down,left,right,map,end): ";
```

```
else {
    cout << "위로 한 칸 올라갑니다." << endl;
    displayMap(map, user_x, user_y);
    hp-=1; // 이동해서 체력 1 감소
    if(hp==0){
        cout<<"HP가 0 이하가 되었습니다. 실패했습니다."<<endl;
        cout<<"게임을 종료합니다."<<endl;
        break;
    } // HP가 0이 되면 "실패"를 출력하고 종료
}
```

```

checkState(map,user_x,user_y);

if (hp==0){
    cout<<"HP가 0 이하가 되었습니다. 실패했습니다."<<endl;
    cout<<"게임을 종료합니다."<<endl;
    break;
} // HP가 0이 되면 "실패"를 출력하고 종료

```

위,아래,좌,우 확인하는 if문마다 hp를 1감소시키고 0이 되었다면 실패를 출력하는 코드를 넣음. 그리고 checkState함수 이후에도 0이 되었다면 실패를 출력하는 코드를 넣음.

```

// 아이템, 포션, 적을 만났을 때 그에 대한 메시지 출력
checkState(map,user_x,user_y);

```

메인 함수에서 함수 호출함.

```

// 아이템, 포션, 적을 만났을 때 그에 대한 메시지 출력하는 함수
void checkState(int map[][mapX],int user_x,int user_y){
    if(map[user_y][user_x] == 1){
        cout<<"아이템이 있습니다"<<endl;
    }
    else if(map[user_y][user_x] == 2){
        cout<<"적이 있습니다. HP가 2 줄어듭니다."<<endl;
        hp-=2;
    }
    else if(map[user_y][user_x] == 3){
        cout<<"포션이 있습니다. HP가 2 늘어납니다."<<endl;
        hp+=2;
    }
}

```

map[][]=전체 지도, user\_x=유저 x값, user\_y=유저 y값

반환값은 없음.

아이템인지 포션인지 적인지 확인 후 해당 메시지 출력. 포션은 hp를 2증가시키고 적은 hp를 2감소시킴.

#### 4. 테스트

##### 1. 기능 별 테스트 결과

- 유저는 체력 20을 가지고 게임 시작

- 처음 명령문을 입력 받을 때마다 HP 함께 출력

```
현재 HP: 20 명령어를 입력하세요 (up,down,left,right,map,end):
```

- 사용자가 이동할 때마다 사용자 체력 1씩 감소

```
현재 HP: 20 명령어를 입력하세요 (up,down,left,right,map,end): right
오른쪽으로 이동합니다.
```

```
      | USER | 적 |      |목적지|
-----
아이템|      |   |   |   |   |
-----
      |      |   |   |   |   |
-----
      |  적  | 포션 |   |   |   |
-----
포션 |      |   |   |   |   |
-----
```

아이템이 있습니다

```
현재 HP: 19 명령어를 입력하세요 (up,down,left,right,map,end):
```

- 아이템, 포션, 적을 만났을 때 그에 대한 메시지를 출력(적을 만날 경우 HP가 2가 줄어듦)

```
현재 HP: 19 명령어를 입력하세요 (up,down,left,right,map,end): right
오른쪽으로 이동합니다.
```

```
      |아이템| USER |      |목적지|
-----
아이템|      |   |   |   |   |
-----
      |      |   |   |   |   |
-----
      |  적  | 포션 |   |   |   |
-----
포션 |      |   |   |   |   |
-----
```

적이 있습니다. HP가 2 줄어듭니다.

```
현재 HP: 16 명령어를 입력하세요 (up,down,left,right,map,end):
```

현재 HP: 14 명령어를 입력하세요 (up,down,left,right,map,end): down  
아래로 한 칸 내려갑니다.

	아이템	적		목적지
아이템				적
		적	USER	
포션				적

포션이 있습니다. HP가 2 늘어납니다.

현재 HP: 15 명령어를 입력하세요 (up,down,left,right,map,end):

- HP가 0이 되면 "실패"를 출력하고 종료

현재 HP: 3 명령어를 입력하세요 (up,down,left,right,map,end): right  
오른쪽으로 이동합니다.

	아이템	적		목적지
아이템				USER
		적	포션	
포션				적

적이 있습니다. HP가 2 줄어듭니다.

HP가 0 이하가 되었습니다. 실패했습니다.

게임을 종료합니다.

## 2. 최종 테스트 스크린샷

```

현재 HP: 16 명령어를 입력하세요 (up,down,left,right,map,end): right
오른쪽으로 이동합니다.
  |아이템|  적  | USER |목적지|
-----
아이템|      |      |  적  |      |
-----
      |      |      |      |      |
-----
      |  적  | 포션  |      |      |
-----
포션  |      |      |      |  적  |
-----
현재 HP: 15 명령어를 입력하세요 (up,down,left,right,map,end): right
오른쪽으로 이동합니다.
  |아이템|  적  |      | USER |
-----
아이템|      |      |  적  |      |
-----
      |      |      |      |      |
-----
      |  적  | 포션  |      |      |
-----
포션  |      |      |      |  적  |
-----
목적지에 도착했습니다! 축하합니다!
게임을 종료합니다.

```

## 5. 결과 및 결론

1. 프로젝트 결과 : mud game을 만들었다.
2. 느낀 점 : tictactoe보다는 쉬웠던 것 같다.