

CSED101. Programming & Problem solving

Spring 2021

Programming Assignment #1 (75 points)

한정수 (saw1515@postech.ac.kr)

■ 제출 마감일: 2021.10.21 23:59

■ 개발 환경: Windows Visual Studio 2019

■ 제출물

- C 소스 코드 (assn1.c)
 - 프로그램의 소스 코드에 채점자의 이해를 돕기 위한 주석을 반드시 붙여주세요.
- 보고서 파일 (.docx, .hwp 또는 .pdf; assn1.docx, assn1.hwp 또는 assn1.pdf)
 - 보고서는 AssnReadMe.pdf를 참조하여 작성하시면 됩니다.
 - 명예 서약 (Honor code): 표지에 다음의 서약을 기입하여 제출해 주세요: “나는 이 프로그래밍 과제를 다른 사람의 부적절한 도움 없이 완수하였습니다.” 보고서 표지에 명예 서약이 기입되어 있지 않은 과제는 제출되지 않은 것으로 처리됩니다.
 - 작성한 소스 코드와 보고서 파일은 PLMS를 통해 제출해 주세요.

■ 주의 사항

- 컴파일이나 실행이 되지 않는 과제는 0점으로 채점됩니다.
- 제출 기한보다 하루 늦게 제출된 과제는 최종 20%, 이를 늦게 제출된 과제는 최종 40% 감점됩니다. 제출 기한보다 사흘 이상 늦으면 제출 받지 않습니다 (0점 처리).
- 각 문제의 제한 조건과 요구 사항을 반드시 지켜 주시기 바랍니다.
- 모든 문제의 출력 형식은 채점을 위해 아래에 제시된 예시들과 최대한 비슷하게 작성해 주세요.
- 부정행위에 관한 규정은 POSTECH 전자COMPUTER공학부 학부위원회의 “POSTECH 전자COMPUTER공학부 부정행위 정의”를 따릅니다 (PLMS의 본 과목 공지사항에 등록된 글 중, 제목이 [document about cheating]인 글에 첨부되어 있는 disciplinary.pdf를 참조하세요).
- 이번 과제는 추가 기능 구현과 관련된 추가 점수가 따로 없습니다.

■ Problem: 주루마불

(목적)

이번 과제를 통하여 조건문, 반복문, 사용자 정의 함수 및 라이브러리 함수 사용법을 익힙니다.

(주의사항)

1. 이번 과제는 함수를 정의하고 사용하는 방법을 익히는 문제이므로 main() 함수에 모든 기능을 구현한 경우 감점 처리 합니다.
2. 문서에 반드시 정의해서 사용해야 할 사용자 정의 함수가 설명되어 있으니 확인 후 구현 하도록 합니다. 이 때, 설명에서 지정한 사용자 정의 함수의 매개변수의 개수와 자료형, 함수 이름, 반환 자료형 등은 자유롭게 변경이 가능합니다. 그러나 동일한 기능을 하는 함수는 반드시 있어야 하며, 변경 시 무엇을 어떻게 변경해서 구현했는지 보고서에 기록 하도록 합니다. 이외에 필요한 함수는 정의해서 사용할 수 있습니다.
3. 프로그램 구현 시, main() 함수를 호출을 직접 하지 않습니다. 즉, 소스 코드 내에 main(); 이라고 호출하지 않습니다.
4. 전역 변수, 배열 및 goto 문은 사용할 수 없으며, 포인터의 경우 수업시간에 다룬 내용에 한해서 사용이 가능합니다.
5. 사용자 입력에서 숫자를 입력 받는 부분에는 숫자만 입력하는 것으로 가정합니다. 즉, 숫자 입력 받는 부분에는 문자 등의 입력에 대해서는 고려할 필요가 없습니다.
6. 명시된 예러 처리 외에는 고려하지 않아도 됩니다.
7. 문제의 출력 형식은 채점을 위해 아래의 실행 예시와 최대한 비슷하게 작성해 주세요.

(설명)

주루마불은 보드게임 부루마블(Blue marble)을 술자리에서 즐기기 위해 변형 시킨 게임이다. 주루마불은 일반 부루마블 게임과 같이 주사위를 던져서 해당 값만큼 자신의 게임말을 옮겨서 해당 칸에서 요구하는 미션을 클리어하면 된다. 본 과제에서는 USER가 COMPUTER를 상대로 게임을 진행하게 되며, 둘 중 정해진 주량 이상의 술을 마시게 되면 패배하게 된다.

다음으로 ‘ 주루마블 게임 ’의 규칙은 다음과 같다.

- USER와 COMPUTER가 번갈아가면서 게임을 진행을 하며, 게임은 항상 USER가 먼저 시작한다.
- USER의 순서에서, USER는 주사위를 던져서 1 ~ 6 중 하나의 숫자가 무작위로 선택되도록 하거나 특정 숫자를 입력하도록 한다. USER는 선택된 주사위 숫자만큼 보드 칸을 이동하게 되고, 이동한 칸에서 요구하는 미션이나 규칙을 따른다.
- 게임 구현의 편의를 위해서 COMPUTER는 자신의 순서에서 주사위를 던지지 않는다. 자신의 차례가 주어질때마다 술 1잔을 마시는 일만 하고, USER에게 순서를 넘긴다. (USER가 미션 칸에서 미션을 수행할 때는 예외이다.)
- USER와 COMPUTER의 주량은 소주 2병(14잔)이며, 14잔 이상을 먼저 마시게 되는 플레이어

가 패배하게 된다. USER 차례에 주량이 초과되면 COMPUTER의 차례로 넘어가지 않고 USER의 패배로 게임은 끝나게 된다.

보드 칸은 다음과 같은 5가지 종류의 칸으로 이루어져있다.

- **시작 칸:** 게임 시작 시 게임말이 위치한 칸이다.
- **미션 칸:** 미션(369 게임¹, 베스킨라빈스 게임²) 중 하나를 랜덤하게 진행한다.
- **찬스 칸:** 찬스 칸 도착 시, 복불복 카드 중 랜덤하게 하나를 시행한다. 찬스 카드 시행 후에는 상대 턴으로 넘어간다.
- **휴식 칸:** 휴식칸에 도착한 USER는 1턴을 쉬게 되며, 자동으로 COMPUTER의 차례로 넘어간다.
- **일반 칸:** 위의 4개의 칸 이외의 모든 칸이 일반 칸으로, 도착한 USER는 술 1잔을 마시게 된다.

보드의 구성은 아래와 같으며, 시작(0)의 위치에서 시작하여 숫자가 증가하는 방향으로 말이 이동하게 된다.

미션(9)	찬스(8)	일반(7)	휴식(6)
일반(10)			일반(5)
일반(11)			일반(4)
시작(0)	일반(1)	찬스(2)	미션(3)

I. 초기 선택 메뉴

프로그램 시작 시, 다음 그림 1과 같이 3가지 선택사항이 있는 메뉴가 출력되며, 사용자로부터 3가지 선택 사항 중 한가지를 입력으로 받도록 한다.

- 초기 선택 메뉴는 main() 함수 내에서 아래의 사용자 정의 함수를 호출하도록 하며, switch-case문을 사용하여 메뉴 선택을 하도록 한다.

- **int print_menu():** 초기 메뉴 화면을 출력하고, 사용자로부터 메뉴 번호를 받아 그 번호를 반환 한다.

```
[주루마블]
=====
1.  게임 설명
2.  게임 시작
3.  게임 종료
=====
입력:
```

```
메뉴를 다시 입력하세요.
[주루마블]
=====
1.  게임 설명
2.  게임 시작
3.  게임 종료
=====
입력:
```

그림 1. 메뉴 선택 화면(좌) 및 메뉴 선택 오류 시 재선택 화면(우)

입력 값은 숫자만 입력된다고 가정하며, 1,2,3 이외의 숫자가 입력될 경우 입력을 무시하고 화면

¹ 369게임 : https://namu.wiki/w/369_게임

² 베스킨라빈스게임 : [https://namu.wiki/w/베스킨라빈스\(게임\)](https://namu.wiki/w/베스킨라빈스(게임))

을 지운 후 다시 메뉴를 선택하도록 한다. 화면을 지우는 방법은 **Tips-화면 지우기**(마지막 쪽)을 참고하여 구현하도록 한다.

II. 게임 설명

그림 1에서 ‘1. 게임 설명’ 메뉴를 선택하면 화면이 지워진 후, 그림 2와 과 같이 게임 설명이 출력된다. 그 아래에 ‘메인 메뉴로 돌아가기(-1)’ 라는 메시지가 출력되며 사용자 입력을 기다린다. 사용자가 -1을 입력하면 화면 지우기 후 그림 1의 메인 화면으로 돌아간다.

```
===== 주루마블 게임 설명 =====  
  
1. USER는 매턴 주사위를 던진다. 주사위를 던질 때 주사위의 눈은 플레이어가 결정한다.  
2. COMPUTER는 주사위를 던지지 않으며 매턴 1잔의 술을 마신다.  
3. USER와 COMPUTER의 주량은 소주 2병(14잔)이며, 14잔을 먼저 마신 쪽이 패배한다.  
4. 보드 칸은 5가지 칸으로 이루어져있으며 USER는 각 칸의 요구사항을 만족시켜야한다.  
  
=====  
메인 메뉴로 돌아가기(-1):
```

그림 2. 게임 설명 화면

➤ 사용자 정의 함수

- `game_info()`: 게임에 대한 설명을 출력하는 함수

III. 게임 종료

그림 1에서 ‘3. 게임 종료’ 메뉴를 선택하면, 그림 3과 같이 프로그램 종료메시지를 출력한 후, 프로그램을 종료한다. (실행 예시의 밑줄은 사용자 입력을 말함. 밑줄은 구현할 필요 없음)

```
[주루마블]  
=====  
1. 게임 설명  
2. 게임 시작  
3. 게임 종료  
=====  
입력: 3  
프로그램을 종료합니다...  
  
계속하려면 아무 키나 누르십시오...
```

그림 3. 게임 종료 화면 예시

IV. 게임 시작

그림 1의 메뉴에서 메뉴 2를 선택하면, `game_start()` 함수가 호출되며 게임이 시작된다. 모든 게임은 `game_start()` 함수 내에서 진행되어야 한다. 설명에서 반드시 구현해야 할 사용자 정의 함수가 설명되어 있으니 확인 후, 반드시 구현하도록 한다. 이외에 필요한 함수를 정의하고 사용할 수 있다.

1. 게임 최초 화면

게임이 시작되면 제일 먼저 화면을 지운 후, `game_start()` 함수를 호출하여 그림 4와 같이 주루마블 게임 화면을 출력한다.

화면의 상단에는 주루마블 보드를 출력한다. USER 말의 위치를 아래 예시처럼 대문자 ‘O’로 표시하며, 현재 시작(0)의 위치에 USER 말이 있음을 볼 수 있다.



그림 4. 주루마블 게임 초기 화면(좌)과 주루마블 보드판을 숫자로 표시(우)

다음으로 그 아래에는 플레이어들의 위치와 현재 주량이 출력된다.

[플레이어 이름 - 현재 말의 위치] (현재 주량/14)

(단, 컴퓨터의 경우에는 보드에서의 말의 위치가 없으므로 항상 0으로 출력한다.)

그리고 그 아래 주량을 아래와 같이 14개의 네모로 나타낸다. 게임을 진행하면서 주량이 채워지게 되면 아래와 같이 채워진 부분에 대해서 채워진 네모 ‘■’로, 나머지 부분에 대해서는 빈 네모 ‘□’로 나타낸다. 아래는 2칸이 채워진 경우이다. (아래 출력은 반드시 for문으로 구현할 것)

■ ■ □ □ □ □ □ □ □ □ □ □ □ □ □ □ → (2/14)

화면의 마지막 줄에 게임 진행을 위해 “주사위를 굴리시겠습니까? (0-굴림, -1-게임포기):” 메시지가 출력되며, -1을 입력하면 현재 진행하는 게임이 중단되고 그림 1의 시작 메뉴로 돌아간다. 0을 입력하면 1~6사이의 주사위 눈이 랜덤하게 선택되고, 1~6사이의 정수를 입력하여 직접 주사위 눈을 선택하면, 이는 채점(또는 디버깅)을 위한 부분으로 입력한 주사위 눈으로 게임이

```

[주루마블 게임]

=====
+-----+-----+-----+
| 미션  | 찬스  | 일반  | 휴식  |
+-----+-----+-----+
| 일반  |                   | 일반  |
+-----+-----+-----+
| 일반  |                   | 일반  |
+-----+-----+-----+
| 시작 0 | 일반  | 찬스  | 미션  |
+-----+-----+-----+

=====

[USER - 0] ( 0/14)
□□□□□□□□□□□□□□

[COMP - 0] ( 0/14)
□□□□□□□□□□□□□□

=====

```

```

[주루마블 게임]

=====
+-----+-----+-----+-----+
| 미션   | 찬스   | 일반   | 휴식   |
+-----+-----+-----+-----+
| 일반   |         |         | 일반   |
+-----+-----+-----+-----+
| 일반   |         |         | 일반   |
+-----+-----+-----+-----+
| 시작 0 | 일반   | 찬스   | 미션   |
+-----+-----+-----+-----+

=====

[USER - 0] ( 0/14)
□□□□□□□□□□□□□□

[COMP - 0] ( 0/14)
□□□□□□□□□□□□□□

=====

주사위를 굴리시겠습니까? (0-굴림, -1-게임포기): 5

```



```

=====
▶▶▶ ROUND 1 ▶▶▶▶▶▶▶▶▶▶▶▶▶▶▶▶▶▶
=====
[USER 's Turn]
-----
주사위 눈 : 1

+-----+-----+-----+-----+
|  미션   |  찬스   |  일반   |  휴식   |
+-----+-----+-----+-----+
|  일반   |          |          |  일반   |
+-----+-----+-----+-----+
|  일반   |          |          |  일반   |
+-----+-----+-----+-----+
| 시작   | 일반 0  | 찬스   | 미션   |
+-----+-----+-----+-----+

일반 칸에 도착하였습니다.
USER가 1잔 마시게 되었습니다.

[COMPUTER 's Turn]
-----
COMPUTER가 1잔 마시게 되었습니다.

=====
[USER - 1] ( 1/14)
■ □ □ □ □ □ □ □ □ □ □ □ □ □ □

[COMP - 0] ( 1/14)
■ □ □ □ □ □ □ □ □ □ □ □ □ □ □

=====

```

```

=====
▶▶▶ ROUND 1 ▶▶▶▶▶▶▶▶▶▶▶▶▶▶▶▶
=====
[USER 's Turn]
-----
주사위 눈 : 1

+-----+-----+-----+-----+
|  미션   |  찬스   |  일반   |  휴식   |
+-----+-----+-----+-----+
|  일반   |          |  일반   |          |
+-----+-----+-----+-----+
|  일반   |          |  일반   |          |
+-----+-----+-----+-----+
| 시작   | 일반 0 | 찬스   | 미션   |
+-----+-----+-----+-----+

일반 칸에 도착하였습니다.
USER가 1잔 마시게 되었습니다.

=====
[COMPUTER 's Turn]
-----
COMPUTER가 1잔 마시게 되었습니다.

=====
[USER - 1] ( 1/14)
■ □ □ □ □ □ □ □ □ □ □ □ □ □ □

[COMP - 0] ( 1/14)
■ □ □ □ □ □ □ □ □ □ □ □ □ □ □

=====
주사위를 굴리시겠습니까? (0-굴림, -1-게임포기):

```

```

=====
▶▶▶ ROUND 1 ▶▶▶▶▶▶▶▶▶▶▶▶▶▶▶▶
=====
[USER 's Turn]
-----
주사위 눈 : 5

+-----+-----+-----+-----+
|  미션   |  찬스   |  일반   |  휴식   |
+-----+-----+-----+-----+
|  일반   |          |  일반 0 |          |
+-----+-----+-----+-----+
|  일반   |          |  일반   |          |
+-----+-----+-----+-----+
| 시작   | 일반   | 찬스   | 미션   |
+-----+-----+-----+-----+

일반 칸에 도착하였습니다.
USER가 1잔 마시게 되었습니다.

=====
[COMPUTER 's Turn]
-----
COMPUTER가 1잔 마시게 되었습니다.

=====
[USER - 5] ( 1/14)
■ □ □ □ □ □ □ □ □ □ □ □ □ □ □

[COMP - 0] ( 1/14)
■ □ □ □ □ □ □ □ □ □ □ □ □ □ □

=====
주사위를 굴리시겠습니까? (0-굴림, -1-게임포기):

```

```

=====
▶▶▶ ROUND 1 ▶▶▶▶▶▶▶▶▶▶▶▶▶▶▶▶
=====
[USER 's Turn]
-----
주사위 눈 : 1

+-----+-----+-----+-----+
|  미션  |  찬스  |  일반  |  휴식  |
+-----+-----+-----+-----+
|  일반  |         |  일반  |         |
+-----+-----+-----+-----+
|  일반  |         |  일반  |         |
+-----+-----+-----+-----+
| 시작  | 일반 0 | 찬스  | 미션  |
+-----+-----+-----+-----+

일반 칸에 도착하였습니다.
USER가 1잔 마시게 되었습니다.

=====
[COMPUTER 's Turn]
-----
COMPUTER가 1잔 마시게 되었습니다.

=====
[USER - 1] ( 1/14)
■ □ □ □ □ □ □ □ □ □ □ □ □ □ □

[COMP - 0] ( 1/14)
■ □ □ □ □ □ □ □ □ □ □ □ □ □ □

=====
주사위를 굴리시겠습니까? (0-굴림, -1-게임포기):

```

```

=====
▶▶▶ ROUND 1 ▶▶▶▶▶▶▶▶▶▶▶▶▶▶▶▶
=====
[USER 's Turn]
-----
주사위 눈 : 5

+-----+-----+-----+-----+
|  미션  |  찬스  |  일반  |  휴식  |
+-----+-----+-----+-----+
|  일반  |         |  일반  |         |
+-----+-----+-----+-----+
|  일반  |         |  일반  |         |
+-----+-----+-----+-----+
| 시작  | 일반  | 찬스  | 미션  |
+-----+-----+-----+-----+

일반 칸에 도착하였습니다.
USER가 1잔 마시게 되었습니다.

=====
[COMPUTER 's Turn]
-----
COMPUTER가 1잔 마시게 되었습니다.

=====
[USER - 5] ( 1/14)
■ □ □ □ □ □ □ □ □ □ □ □ □ □ □

[COMP - 0] ( 1/14)
■ □ □ □ □ □ □ □ □ □ □ □ □ □ □

=====
주사위를 굴리시겠습니까? (0-굴림, -1-게임포기):

```

```

=====
▶▶▶ ROUND 1 ▶▶▶▶▶▶▶▶▶▶▶▶▶▶▶▶
=====
[USER 's Turn]
-----
주사위 눈 : 1

+-----+-----+-----+-----+
|  미션  |  찬스  |  일반  |  휴식  |
+-----+-----+-----+-----+
|  일반  |         |  일반  |         |
+-----+-----+-----+-----+
|  일반  |         |  일반  |         |
+-----+-----+-----+-----+
| 시작  | 일반 0 | 찬스  | 미션  |
+-----+-----+-----+-----+

일반 칸에 도착하였습니다.
USER가 1잔 마시게 되었습니다.

=====
[COMPUTER 's Turn]
-----
COMPUTER가 1잔 마시게 되었습니다.

=====
[USER - 1] ( 1/14)
■ □ □ □ □ □ □ □ □ □ □ □ □ □ □

[COMP - 0] ( 1/14)
■ □ □ □ □ □ □ □ □ □ □ □ □ □ □

=====
주사위를 굴리시겠습니까? (0-굴림, -1-게임포기):

```

```

=====
▶▶▶ ROUND 1 ▶▶▶▶▶▶▶▶▶▶▶▶▶▶▶▶
=====
[USER 's Turn]
-----
주사위 눈 : 5

+-----+-----+-----+-----+
|  미션  |  찬스  |  일반  |  휴식  |
+-----+-----+-----+-----+
|  일반  |         |  일반  |         |
+-----+-----+-----+-----+
|  일반  |         |  일반  |         |
+-----+-----+-----+-----+
| 시작  | 일반  | 찬스  | 미션  |
+-----+-----+-----+-----+

일반 칸에 도착하였습니다.
USER가 1잔 마시게 되었습니다.

=====
[COMPUTER 's Turn]
-----
COMPUTER가 1잔 마시게 되었습니다.

=====
[USER - 5] ( 1/14)
■ □ □ □ □ □ □ □ □ □ □ □ □ □ □

[COMP - 0] ( 1/14)
■ □ □ □ □ □ □ □ □ □ □ □ □ □ □

=====
주사위를 굴리시겠습니까? (0-굴림, -1-게임포기):

```

USER가 주사위를 굴리거나 값을 입력한 후에는 화면을 지운 후 라운드, USER턴, COMP턴, 주량을 그림 5와 같이 출력한다. 라운드의 경우 매 번 주사위를 던질 때마다 한 라운드씩 증가하게 된다. 이어서 USER턴이 실행되고 던진 주사위 눈의 값과 보드를 출력하고, 도착한 보드칸의 요구사항을 수행한 후 결과를 출력하게 된다. 그리고 COMPUTER턴의 경우 미션칸을 제외하고는 모든 칸에서 1장 마시는 것을 출력하게 된다. 이어서 현재까지의 주량을 출력하고 주사위를 계속 굴릴

지, 게임을 포기할지를 선택하게 된다.

■ 사용자 정의 함수

- **void game_start():** 게임을 시작하고 필요한 변수(주량, 위치)들을 초기화한다. 매 라운드마다 USER는 주사위를 굴릴지 게임 포기를 할 것인지를 선택하도록 하며, COMPUTER는 미션 칸을 제외하고 매 턴마다 1잔을 마시도록 한다. 이는 게임 종료까지 반복된다.
- **void print_drink():** USER와 COMPUTER의 턴이 종료된 후 각자의 주량을 출력한다.
- **void print_map():** USER 말의 위치가 포함된 주루마블 보드를 출력한다.
- **char set_position(int user_pos, int pos):** 보드에서의 USER 말의 위치와 현재 보드의 위치를 전달 받아, USER 말의 위치와 현재 보드의 위치가 같으면 대문자 'O'를 반환하고, 같지 않으면 스페이스 한 칸(' ')을 반환한다. 예를 들면, set_position(0, 0)에 대해서 대문자 'O'를 반환. print_map() 함수에서 호출되도록 구현한다.

2. 게임 플레이 설명

주루마블의 경우 USER가 주사위를 굴리고 랜덤하게 나온 주사위 눈의 숫자만큼 보드에서 게임말을 이동 시킨다. 참고로, 연속적인 실행 예시는 그림 4의 게임 초기화면 → 그림 7의 ROUND 1 → 그림 8의 ROUND 2 → 그림 10의 ROUND 3 → 그림 11의 ROUND 4 → 그림 6의 ROUND 5 → 그림 9의 ROUND 6 → 그림 11의 ROUND 7 → 그림 8의 ROUND 8 → 그림 9의 ROUND 9 → 그림 12 ROUND 10의 순서로 확인할 수 있다.

(1) 게임 시작

모든 게임은 USER만 주사위를 굴리게 된다. 주사위의 눈은 1~6이며 0을 입력하여 랜덤하게 하나가 선택되도록 하거나 주사위 눈에 해당하는 정수를 직접 입력 받도록 한다. 주사위 눈의 결과가 프로그램이 실행될 때마다 다른 값을 반환하도록 **srand()** 함수를 사용할 것.

■ 사용자 정의 함수

- **void play_juru(int user_pos):** USER가 도착한 보드칸의 규칙에 따라 적절한 행동을 수행한다. switch-case문을 이용하여 각 보드 칸에서 요구하는 기능을 구현한다. 해당 함수 내에서, 각 칸에서 수행되어야 기능을 정의한 함수들(아래 설명 참조)을 호출한다.

(2) 보드칸 설명

주루마블 보드는 4x4 크기로 보드의 구성은 아래와 같으며, 시작(0)의 위치에서 시작하여 숫자가 증가하는 방향으로 말이 이동하게 된다. 모든 칸은 고정되어 있으며, 게임이 새롭게 시작되어도 칸들의 위치는 바뀌지 않는다.

미션(9)	찬스(8)	일반(7)	휴식(6)
일반(10)			일반(5)
일반(11)			일반(4)
시작(0)	일반(1)	찬스(2)	미션(3)

각 보드칸의 규칙과 실행 예시는 다음과 같다.

➤ 시작 칸

- 주루마블 게임 시작 시 USER의 게임말은 그림 3과 같이 시작 칸에 배치된다.
- 게임은 USER가 먼저 주사위를 던져서 나온 숫자만큼, 시작 칸에서 말을 움직여서 시작한다. 그림 7은 USER가 주사위를 던져 1이 나온 경우로, 1칸 이동하여 일반(1)칸으로 이동했음을 볼 수 있다.
- 만약 한바퀴를 돈 후, 시작 칸에 도착하면 휴식 칸과 같이 아무런 일도 하지 않는다(그림 6).

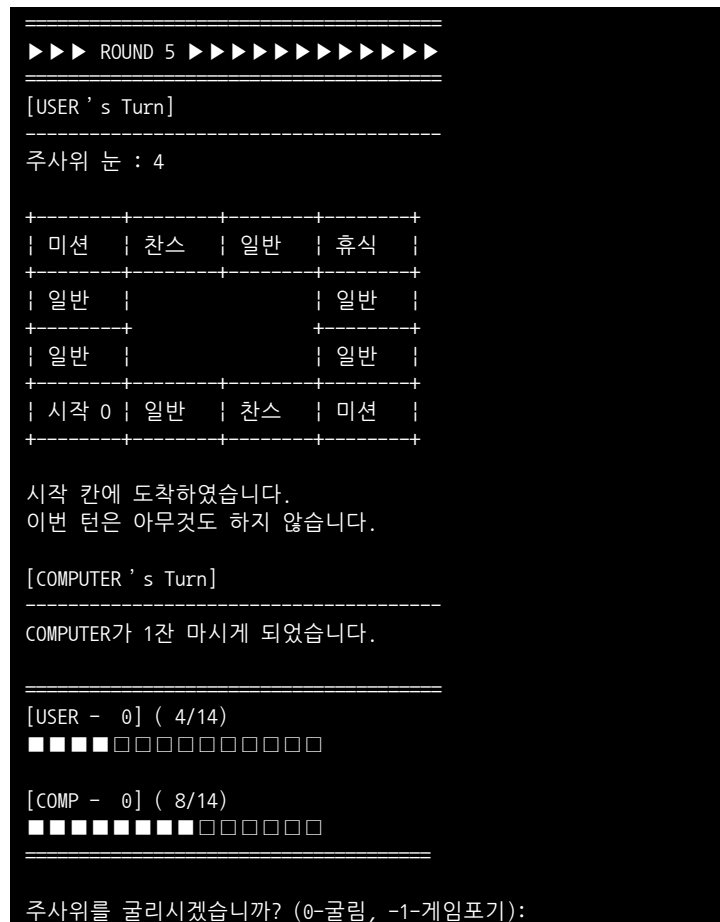


그림 6. 시작 칸 동작 화면

■ 사용자 정의 함수

- 일반 칸 도착 시, 수행해야 할 기능을 함수로 정의한 후 사용할 것

➤ 일반 칸

- 일반 칸에 도착한 USER는 술 1잔을 마시게 된다. 만약 1잔을 마신 후 주량이 14잔인 상태가 되면 패배하게 된다. 즉, 현재 주량이 13잔인 상태에서 일반 칸에 도착 시 패배하게 된다.
- 아래 그림 7은 USER의 순서에서 던진 주사위의 값은 1이며, 그 아래에 보드판에서 USER의 말이 1칸 이동하여 일반(1)의 위치에 있음을 볼 수 있다.

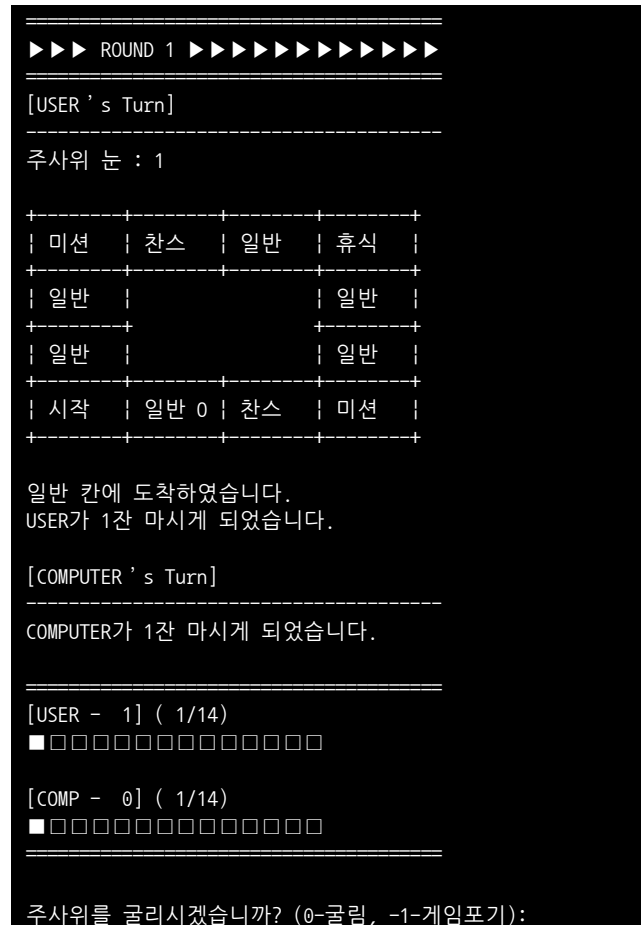


그림 7. 일반 칸 동작 화면

■ 사용자 정의 함수

- 일반 칸 도착 시, 수행해야 할 기능을 함수로 정의한 후 사용할 것

➤ 미션 칸

- 미션 칸 도착 시, ‘ 369 게임 ’ 과 ‘ 베스킨라빈스 게임 ’ 중 하나를 랜덤하게 실행한다.
- 미션 칸에 도착하여 게임 시작 시, USER는 COMPUTER를 상대로 게임을 진행하게 된다.
- 미션의 경우, USER가 이기면 COMPUTER가 4잔을 마시며, COMPUTER가 이기면 USER가 4잔을 마시게 된다.
- USER가 미션 칸에 도착해서 미션 수행 후, COMPUTER의 순서가 되면 1잔 마시는 일 없이 USER 차례로 넘어간다.

• 369 게임

- 369 게임은 1부터 숫자를 하나씩 대면서, 3, 6, 9가 들어가는 숫자는 숫자를 말하는 대신 박수를 치는 게임이다.
- 본 과제에서는 COMPUTER와 USER가 번갈아가면서 1부터 숫자를 하나씩 대면서, 369가 들어가는 숫자와 10의 배수에서 박수를 친다. 박수를 쳐야 하는 숫자에서는 0의 숫자를 대도록 한다.
- 게임은 USER와 COMPUTER 중 하나가 랜덤하게 선택되어 먼저 시작하도록 한다. 아래 예시는 COMPUTER가 먼저 시작하는 경우로 1의 숫자를 먼저 출력하며, 다음 USER의

입력을 기다리며 USER가 순서에 맞는 숫자를 입력하면 COMPUTER의 순서로 넘어간다.

- COMPUTER는 자기 순서에서 항상 올바른 숫자를 말한다(출력한다). 21의 숫자까지 USER가 올바른 숫자를 말한(입력한) 경우, USER가 승리하게 된다.

```

=====
▶▶▶ ROUND 2 ▶▶▶▶▶▶▶▶▶▶▶▶▶▶▶▶
=====
[USER 's Turn]
=====
주사위 눈 : 2

+-----+
| 미션 | 찬스 | 일반 | 휴식 |
+-----+
| 일반 |      | 일반 |      |
+-----+
| 일반 |      | 일반 |      |
+-----+
| 시작 | 일반 | 찬스 | 미션 0 |
+-----+

미션 칸에 도착하였습니다.
369 게임이 선택되었습니다.

★★★★★   369 게임   ★★★★★

COMP >> 1
USER >> 2
COMP >> 0
USER >> 4
COMP >> 5
USER >> 0
COMP >> 7
USER >> 8
COMP >> 0
USER >> 0
COMP >> 11
USER >> 12
COMP >> 0
USER >> 14
COMP >> 15
USER >> 0
COMP >> 17
USER >> 18
COMP >> 0
USER >> 0
COMP >> 21

=====
USER가 승리하였습니다.
COMPUTER가 4잔 마시게 되었습니다.

[COMPUTER 's Turn]
=====
COMPUTER가 0잔 마시게 되었습니다.

=====
[USER - 3] ( 1/14)
■□□□□□□□□□□□□□□

[COMP - 0] ( 5/14)
■ ■ ■ ■ □□□□□□□□□□

=====
주사위를 굴리시겠습니까? (0-굴림, -1-게임포기):

```

```

=====
▶▶▶ ROUND 8 ▶▶▶▶▶▶▶▶▶▶▶▶▶▶▶▶
=====
[USER 's Turn]
=====
주사위 눈 : 1

+-----+
| 미션 0 | 찬스 | 일반 | 휴식 |
+-----+
| 일반 |      |      | 일반 |
+-----+
| 일반 |      |      | 일반 |
+-----+
| 시작 | 일반 | 찬스 | 미션 |
+-----+

미션 칸에 도착하였습니다.
369 게임이 선택되었습니다.

★★★★★   369 게임   ★★★★★

COMP >> 1
USER >> 2
COMP >> 0
USER >> 3

=====
COMPUTER가 승리하였습니다.
USER가 4잔 마시게 되었습니다.

[COMPUTER 's Turn]
=====
COMPUTER가 0잔 마시게 되었습니다.

=====
[USER - 9] ( 6/14)
■ ■ ■ ■ ■ □□□□□□□□□□

[COMP - 0] (13/14)
■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ □□□□□□

=====
주사위를 굴리시겠습니까? (0-굴림, -1-게임포기):

```

그림 8. 미션 칸(369 게임) 동작 화면

• 베스킨라빈스 게임

- 베스킨라빈스 게임은 1부터 31까지의 수를 순서대로 불러서 마지막 31을 말하는 사람이 지는 게임으로, 게임 참여자들은 각자의 순서에 1~3개까지의 숫자를 연속하여

말할 수 있다.

- 본 과제에서는 COMPUTER와 USER가 번갈아가면서 게임을 진행하는데, USER는 자신의 순서에 연속하여 부를 숫자의 개수를 입력하며, COMPUTER는 자신의 순서에 연속하여 부를 숫자의 개수를 랜덤하게 선택한다. 각자의 순서에 1~3 범위 내의 숫자를 선택 하도록 한다. (부를 숫자가 3개 미만으로 남은 경우에도 1~3 범위 내의 숫자를 선택 하도록 구현한다.)
- 만약 USER가 1~3 이외에 값을 입력하게 되면 해당 값을 무시하고 다시 입력 받는다.
- 게임 시작은 랜덤하게 플레이어(COMPUTER 또는 USER)를 선택하여 시작하도록 한다.

```

=====
▶▶▶ ROUND 6 ▶▶▶▶▶▶▶▶▶▶▶▶▶▶
=====
[USER 's Turn]
=====
주사위 눈 : 3

+-----+
| 미션 | 찬스 | 일반 | 휴식 |
+-----+
| 일반 |      | 일반 |      |
+-----+
| 일반 |      | 일반 |      |
+-----+
| 시작 | 일반 | 찬스 | 미션 0 |
+-----+

미션 칸에 도착하였습니다..
베스킨라빈스 게임이 선택되었습니다.

★★★★★ 베스킨라빈스 게임 ★★★★★
=====
USER >> 3
1 2 3
COMP >> 1
4
USER >> 3
5 6 7
COMP >> 2
8 9
USER >> 1
10
COMP >> 3
11 12 13
USER >> 3
14 15 16
COMP >> 2
17 18
USER >> 3
19 20 21
COMP >> 3
22 23 24
USER >> 3
25 26 27
COMP >> 2
28 29
USER >> 1
30
COMP >> 2
31
=====
USER가 승리하였습니다.
COMPUTER가 4잔 마시게 되었습니다.

```

```

=====
▶▶▶ ROUND 9 ▶▶▶▶▶▶▶▶▶▶▶▶▶▶
=====
[USER 's Turn]
=====
주사위 눈 : 6

+-----+
| 미션 | 찬스 | 일반 | 휴식 |
+-----+
| 일반 |      | 일반 |      |
+-----+
| 일반 |      | 일반 |      |
+-----+
| 시작 | 일반 | 찬스 | 미션 0 |
+-----+

미션 칸에 도착하였습니다..
베스킨라빈스 게임이 선택되었습니다.

★★★★★ 베스킨라빈스 게임 ★★★★★
=====
USER >> 4
WRONG INPUT!
USER >> 0
WRONG INPUT!
USER >> 3
1 2 3
COMP >> 3
4 5 6
USER >> 3
7 8 9
COMP >> 2
10 11
USER >> 3
12 13 14
COMP >> 3
15 16 17
USER >> 3
18 19 20
COMP >> 2
21 22
USER >> 3
23 24 25
COMP >> 1
26
USER >> 3
27 28 29
COMP >> 1
30
USER >> 1
31
=====
COMPUTER가 승리하였습니다.

```

```

[COMPUTER 's Turn]
-----
COMPUTER가 0잔 마시게 되었습니다.

=====
[USER - 3] ( 4/14)
■■■■□□□□□□□□

[COMP - 0] (12/14)
■■■■■■■■■■■■■■□□

=====
주사위를 굴리시겠습니까? (0-굴림, -1-게임포기):

```

```

USER가 4잔 마시게 되었습니다.

[COMPUTER 's Turn]
-----
COMPUTER가 0잔 마시게 되었습니다.

=====
[USER - 3] (10/14)
■■■■■■■■■■■■■■□□□□

[COMP - 0] (13/14)
■■■■■■■■■■■■■■■□

=====
주사위를 굴리시겠습니까? (0-굴림, -1-게임포기):

```

그림 9. 미션 칸(베스킨라빈스 게임) 동작 화면

■ 사용자 정의 함수

- 미션 칸 도착 시, 수행해야 할 기능을 함수로 정의한 후 사용할 것.
이 함수에서 아래 정의한 두 함수를 호출하여 사용할 것.
- 369 게임을 실행하는 함수를 정의 후 사용할 것.
- 베스킨라빈스 게임을 실행하는 함수를 정의 후 사용할 것.

➤ 휴식 칸

- 휴식 칸 도착 시, 플레이어는 아무런 행위를 하지않고 턴을 다음 플레이어에게 넘긴다.

```

=====
▶▶▶ ROUND 3 ▶▶▶▶▶▶▶▶▶▶▶▶▶▶▶▶
=====
[USER 's Turn]
-----
주사위 눈 : 3

+-----+-----+-----+-----+
| 미션  | | 찬스  | | 일반  | | 휴식 0 |
+-----+-----+-----+-----+
| 일반  | |         | | 일반  | |
+-----+-----+-----+-----+
| 일반  | |         | | 일반  | |
+-----+-----+-----+-----+
| 시작  | | 일반  | | 찬스  | | 미션  |
+-----+-----+-----+-----+

휴식 칸에 도착하였습니다.
이번 턴은 아무것도 하지 않습니다.

[COMPUTER 's Turn]
-----
COMPUTER가 1잔 마시게 되었습니다.

=====
[USER - 6] ( 1/14)
■□□□□□□□□□□□□□□

[COMP - 0] ( 6/14)
■■■■■■■■□□□□□□□□□□

=====
주사위를 굴리시겠습니까? (0-굴림, -1-게임포기):

```

그림 10. 휴식 칸 동작 화면

➤ **찬스 칸**

- 찬스 칸 도착 시, USER는 다음 두 찬스 카드 중 하나를 뽑아서 랜덤하게 시행한다.
 - ① **세잔 더!** : ‘ 세잔 더! ’ 카드가 선택된 경우, 찬스 칸에 도착한 USER만 소주 3잔을 마시게 된다.
 - ② **여명808** : ‘ 여명808 ’ 카드가 선택된 경우, 술기운이 -2잔 만큼 해소되게 된다. 예를 들면, USER의 현재 주량 상태가 7잔일 경우에는 5잔으로 바뀌며, 현재 주량이 0잔 또는 1잔일 경우 0잔으로 바뀐다(음수 주량은 없음).

```

=====
▶▶▶ ROUND 4 ▶▶▶▶▶▶▶▶▶▶▶▶▶▶▶▶
=====
[USER 's Turn]
-----
주사위 눈 : 2

+-----+-----+-----+-----+
| 미션 | 찬스 0 | 일반 | 휴식 |
+-----+-----+-----+-----+
| 일반 |         | 일반 |     |
+-----+-----+-----+-----+
| 일반 |         | 일반 |     |
+-----+-----+-----+-----+
| 시작 | 일반 | 찬스 | 미션 |
+-----+-----+-----+-----+

찬스 칸에 도착하였습니다.
'세잔 더!' 카드를 뽑았습니다.
USER가 3잔 마시게 되었습니다.

[COMPUTER 's Turn]
-----
COMPUTER가 1잔 마시게 되었습니다.

=====
[USER - 8] ( 4/14)
■■■■□□□□□□□□□□

[COMP - 0] ( 7/14)
■■■■■■■■□□□□□□□□

=====
주사위를 굴리시겠습니까? (0-굴림, -1-게임포기):
  
```

```

=====
▶▶▶ ROUND 7 ▶▶▶▶▶▶▶▶▶▶▶▶▶▶▶▶
=====
[USER 's Turn]
-----
주사위 눈 : 5

+-----+-----+-----+-----+
| 미션 | 찬스 0 | 일반 | 휴식 |
+-----+-----+-----+-----+
| 일반 |         | 일반 |     |
+-----+-----+-----+-----+
| 일반 |         | 일반 |     |
+-----+-----+-----+-----+
| 시작 | 일반 | 찬스 | 미션 |
+-----+-----+-----+-----+

찬스 칸에 도착하였습니다.
'여명 808' 카드를 뽑았습니다.
USER가 -2잔 마시게 되었습니다.

[COMPUTER 's Turn]
-----
COMPUTER가 1잔 마시게 되었습니다.

=====
[USER - 8] ( 2/14)
■■□□□□□□□□□□□□

[COMP - 0] (13/14)
■■■■■■■■■■■■■■■■□

=====
주사위를 굴리시겠습니까? (0-굴림, -1-게임포기):
  
```

그림 11. 찬스 칸 동작 화면

3. 게임 종료 및 승패 판정

주루마불 게임은 COMPUTER나 USER 중 먼저 14잔의 술을 마신 쪽이 패배하게 된다. 게임이 종료 될 경우 "게임이 종료되었습니다."라는 메시지와 함께 USER와 COMPUTER의 최종 주량을 아래 그림 12와 같이 표시한다. 그리고 그 아래 누가 승리했는지 출력한다.

- USER가 14잔의 술을 먼저 마시게 되면, COMPUTER 순서로 넘어가지 않고 게임을 종료한다.
- 현재 주량이 13잔인 상태에서 2잔 이상을 마시게 되어 주량이 초과 되면 14잔까지만 표시를 하고 게임을 종료하도록 한다.

또한, -1을 입력하면 화면 지우기 후 그림 1과 같이 초기 선택 메뉴로 돌아간다. 단 메뉴로 돌아가기 위한 입력으로 -1 이외에는 고려할 필요가 없다.

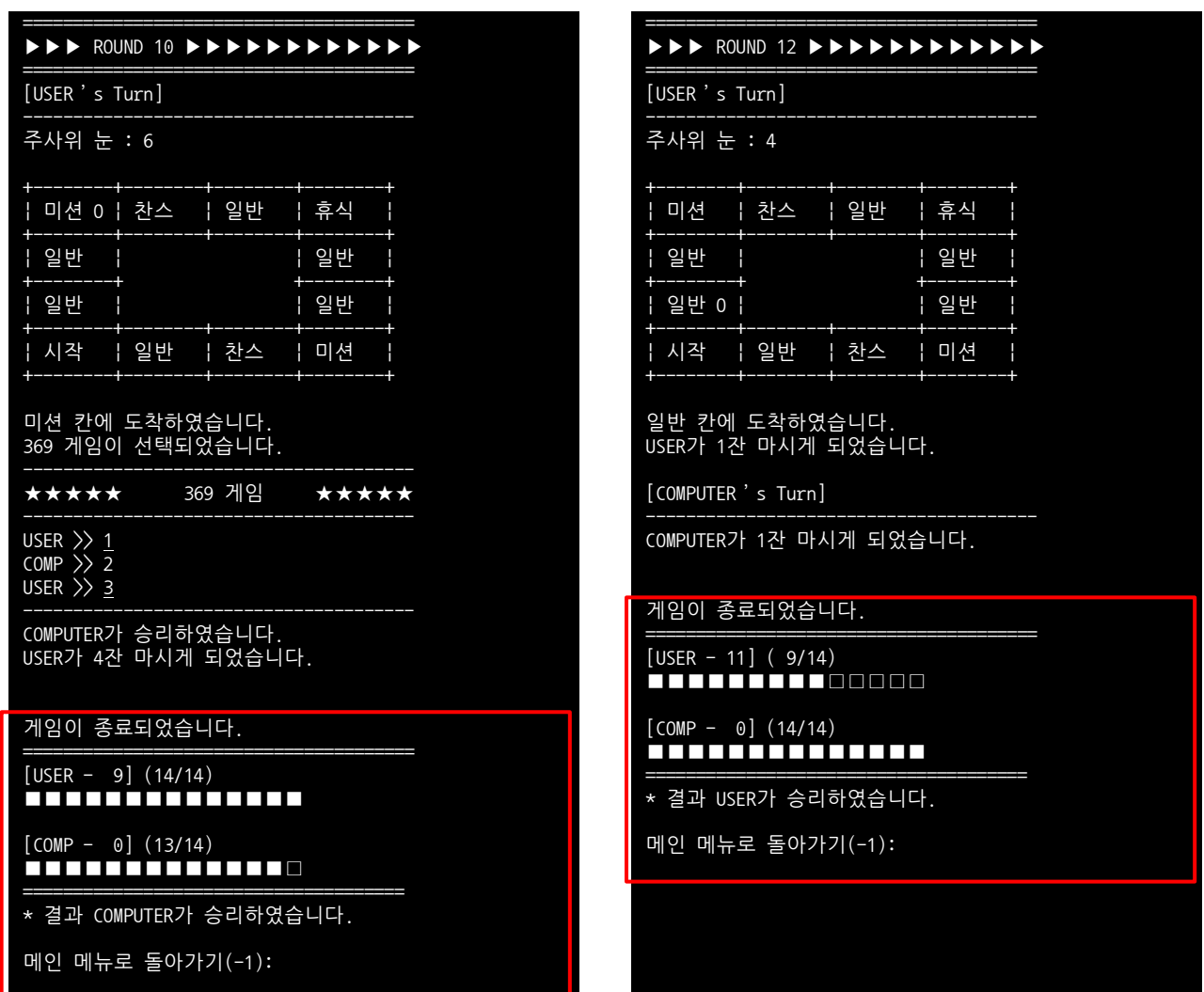


그림 12. 게임 종료 시 결과 화면

Tips.

● 화면 지우기

화면을 지우고 싶을 경우, `stdlib` 헤더 파일을 포함 시킨 후 `system()` 함수를 사용하도록 한다. `system()` 함수의 매개변수로 `"cls"`를 넘겨주면 된다. 아래의 소스 코드를 컴파일하여 실행하면 콘솔 화면에 "Erase me!" 문자열 이 출력된다. Enter키를 입력하면 화면이 지워진 후에 "Erased" 문자열이 출력되는 것을 볼 수 있다.

```
#include <stdio.h>
#include <stdlib.h>
int main()
{
    char c;
    printf("Erase me!\n");
    scanf("%c", &c);
    system("cls");
    printf("Erased\n");
    return 0;
}
```