

2022-1학기 게임프로그래밍을위한캡스톤디자인 중간고사 (2 / 2교시)

학번: _____ 이름: _____

- 2교시는 여러분이 특정 목표를 달성하기 위한 Java 코드 요소를 작성할 수 있는지를 주로 묻습니다. 여러분이 작성한 답안은 우리 수업 표준 환경을 기준으로 채점됩니다.
- 답안 칸 밖에 적은 글씨들은 절대 채점에 반영되지 않습니다(답안은 마련된 칸 안에 적어 주세요). 답안 칸 안에 적은 모든 글씨들은 채점에 반영됩니다(답안 칸에 이상한거 적지 말아 주세요). 답안 칸에 있는 글씨를 지울 수 없다면 그 글씨에 ~~가로줄을 두 개~~ 그어 주세요.
- 여러분이 적은 글자는 사람인 강사가 읽습니다. 모호한 글자를 적는 경우 다른 곳에 적은 동일한 글자들의 필체를 감안하여 인식을 수행할 예정이나, 여러분의 의도가 명백히 강사에게 전달되지 않을 가능성이 있습니다. 이 점을 감안하여 몇몇 혼동하기 쉬운 글자들은 꼭 또박또박 적어 주세요.
- 일부 문제는 답안 적을 곳을 두 개 마련해 두었습니다. 위 칸을 쓰다가 다시 적으려는 경우 줄 곳고 아래 칸을 사용해 주세요. 만약 두 칸 모두 글씨가 적혀 있다면 무조건 아래 칸의 내용을 답안으로 간주합니다.
- 2교시 시험은 50분 동안 진행되며 만점은 50점입니다.

(1 ~ 3)다음 선언들을 먼저 읽고, 문항별 목표들을 달성할 수 있는 수식을 각각 적어 주세요.

```
int number;  
  
int arr[] = new int[2];
```

1. number 자리에 담긴 값을 number 자리에 담는 수식 (2점, 1분)

2. arr에 담긴 모든 값을 다 더한 값을 number 자리에 담는 수식 (3점, 3분)

3. arr의 0번째 자리에 담긴 값이 0이면 1을, 1이면 0을 number 자리에 담는 수식 (3점, 4분)

(다음 페이지에서 계속됩니다)

(4 ~ 5) 미리 코드를 적어 둔 아래 답안 칸의 적절한 자리에, 문항별 목표를 달성할 수 있는 선언을 적어 주세요.

4. Sample.Yeshi() 정의 중괄호 안에서만 볼 수 있는 새 int 형식 Data 이름 a에 대한 선언 (3점, 3분)
5. 누구나 볼 수 있지만, 추가적인 이름 사용(코드 밖에 있거나 내가 직접 선언하거나 등등) 없이는 Sample.Yeshi() 정의 중괄호 안에서는 사용 가능해도 Sample.Example() 정의 중괄호 안에서는 사용할 수 없는 새 int 형식 Data 이름 b에 대한 선언 (4점, 3분)

```
public class Sample {  
  
    public void Yeshi() {  
  
    }  
  
    public static void Example() {  
  
    }  
  
}
```

6. '입력받은 두 수의 합이 미리 정해 둔 수와 같으면 이기는 게임'을 만들기 위해 미리 아래와 같이 구성해 두었습니다. 전체 게임 실행 흐름을 구성하기 위한 main() 메서드 정의 내용물을 적어 주세요. (8점, 8분)

```
public class Program
{
    public static int GetRandomNumber() { /* 미리 잘 작성해 둬 */ }
    public static int GetNumberFromPlayer() { /* 미리 잘 작성해 둬 */ }

    public static void main(String[] args) { /* 6번 답안이 들어가는 곳 */ }
}
```

- 미리 작성해 둔 두 메서드는 호출하면 메서드 이름에 걸맞은 적절한 값을 return해 줄 거예요.
- 답안 적을 때 GetRandomNumber(), GetNumberFromPlayer()는 각각 GRN(), GNFP()로 줄여 적어도 돼요.
- 게임이 성립하기 위해 아래 조건을 만족해야 해요:
 - 사용자 입력을 받기 전에 미리 임의의 수를 골라 두어야 해요.
 - 입력은 두 번 받아야 해요.
 - 게임 결과(이김, 짐)를 사용자에게 적절히 보여주어야 해요(sysout())과 같이 줄여 써도 돼요)

(7 ~ 9) 어떤 게임의 '아이템 섭취' 부분을 구성하기 위해 두 개의 Data 클래스와 테스트용 Code 클래스에 대한 클래스 정의를 적어 보려 합니다. 잘 읽고 순서대로 차근차근 도전해 주세요.

- 문항에서 정해 두지 않은 이름은 자유롭게 정해도 좋아요.
- 문항에서 요구하지 않은 요소들(각종 부수적인 메서드들 등)은 고려하지 않아도 좋아요.
- 편의를 위해 여기까지의 설명 내용을 총 세 장에 복붙해 놓았어요. 내용은 동일하니 안심해도 좋아요.

7. 아이템 한 뭉치에 대한 정보들을 담은 Data 클래스인 `class I`에 대한 클래스 정의를 적어 주세요. (6점, 7분)

- 필요한 필드(무얼 담을 예정인지, 필드의 형식):
 - 아이템 이름, `String` 형식
 - 섭취시 체력 회복량, `int` 형식
 - 아이템의 수량, `int` 형식
- 필요한 메서드: 없음

(다음 페이지에서 계속됩니다)

(7 ~ 9) 어떤 게임의 '아이템 섭취' 부분을 구성하기 위해 두 개의 Data 클래스와 테스트용 Code 클래스에 대한 클래스 정의를 적어 보려 합니다. 잘 읽고 순서대로 차근차근 도전해 주세요.

- 문항에서 정해 두지 않은 이름은 자유롭게 정해도 좋아요.
- 문항에서 요구하지 않은 요소들(각종 부수적인 메서드들 등)은 고려하지 않아도 좋아요.
- 편의를 위해 여기까지의 설명 내용을 총 세 장에 복붙해 놓았어요. 내용은 동일하니 안심해도 좋아요.

8. 내 캐릭터에 대한 정보들을 담기 위한 Data 클래스인 `class c`에 대한 클래스 정의를 적어 주세요. (8점, 6분)

- 필요한 필드(무얼 담을 예정인지, 필드의 형식):
 - 캐릭터 이름, `String` 형식
 - 캐릭터의 현재 체력, `int` 형식
- 필요한 메서드(수행해야 할 작업):
 - '섭취': 앞 문항에서 정의해 둔 `I` 형식 인스턴스 하나를 인수로 받아, 해당 아이템 물치의 수량을 1만큼 감소시키면서 이 캐릭터의 체력을 회복시킵니다.

(다음 페이지에서 계속됩니다)

(7 ~ 9) 어떤 게임의 '아이템 섭취' 부분을 구성하기 위해 두 개의 Data 클래스와 테스트용 Code 클래스에 대한 클래스 정의를 적어 보려 합니다. 잘 읽고 순서대로 차근차근 도전해 주세요.

- 문항에서 정해 두지 않은 이름은 자유롭게 정해도 좋아요.
- 문항에서 요구하지 않은 요소들(각종 부수적인 메서드들 등)은 고려하지 않아도 좋아요.
- 편의를 위해 여기까지의 설명 내용을 총 네 장에 복붙해 놓았어요. 내용은 동일하니 안심해도 좋아요.

9. 마지막으로, 두 Data 클래스를 테스트하기 위한 Code 클래스인 `class Tester`를 정의해 주세요. (13점, 15분)

- 필요한 필드: 없음(자유롭게 선언해 가면서 써도 돼요)
- 필요한 메서드(수행해야 할 작업):
 - `Test()`:
 - ◆ c 형식 인스턴스 하나, i 형식 인스턴스 하나를 각각 만듭니다. 필드 값은 자유롭게 지정하되, 아이템 무게의 수량은 3 이상으로 지정합니다.
 - ◆ 아이템 무게의 수량만큼 반복하는 반복 흐름을 구성하여, 매 반복마다 내 캐릭터가 아이템 무게의 아이템 하나를 섭취하도록 구성합니다. 이 때 섭취 전/후의 내 캐릭터 체력을 화면에 표시합니다(`sysout()`써도 돼요).
 - ◆ 아이템 섭취가 끝나면 "테스트 완료"를 화면에 표시합니다.

(적다가 망하면 다음 페이지에 있는 여분의 답안지를 활용해 주세요)

(7 ~ 9) 어떤 게임의 '아이템 섭취' 부분을 구성하기 위해 두 개의 Data 클래스와 테스트용 Code 클래스에 대한 클래스 정의를 적어 보려 합니다. 잘 읽고 순서대로 차근차근 도전해 주세요.

- 문항에서 정해 두지 않은 이름은 자유롭게 정해도 좋아요.
- 문항에서 요구하지 않은 요소들(각종 부수적인 메서드들 등)은 고려하지 않아도 좋아요.
- 편의를 위해 여기까지의 설명 내용을 총 네 장에 복붙해 놓았어요. 내용은 동일하니 안심해도 좋아요.

10. 마지막으로, 두 Data 클래스를 테스트하기 위한 Code 클래스인 `class Tester`를 정의해 주세요. (13점, 15분)

- 필요한 필드: 없음(자유롭게 선언해 가면서 써도 돼요)
- 필요한 메서드(수행해야 할 작업):
 - `Test()`:
 - ◆ c 형식 인스턴스 하나, i 형식 인스턴스 하나를 각각 만듭니다. 필드 값은 자유롭게 지정하되, 아이템 뭉치의 수량은 3 이상으로 지정합니다.
 - ◆ 아이템 뭉치의 수량만큼 반복하는 반복 흐름을 구성하여, 매 반복마다 내 캐릭터가 아이템 뭉치의 아이템 하나를 섭취하도록 구성합니다. 이 때 섭취 전/후의 내 캐릭터 체력을 화면에 표시합니다(`sysout()`써도 돼요).
 - ◆ 아이템 섭취가 끝나면 "테스트 완료"를 화면에 표시합니다.

(적다가 망하면 강사에게 여분의 답안지를 요청해 주세요)