

문제은행 [3, 4] 복습풀이 + 질문 통합

12/20 개강 SW개발자 양성과정 [손현지]

2021. 01. 10

<문제은행[3] - 1번, Answer1 보며 다시 풀이하기>

기존의 코드

```
import java.util.Scanner;

public class Practice1 {
    public static void main(String[] args) {
        //문제1. 아래와 같은 등비 수열이 있다.
        // 1, 2, 4, 8, 16, 32, 64, 128, 256, 512, 1024, ...
        // 사용자 입력을 통해 원하는 위치의 값을 뽑아내도록 프로그래밍 해보자!
        //(1~32번째 혹은 31번째 항까지만 올바른 결과가 나올 것임)

        int[] arr;

        //1. import Scanner를 사용해 내가 구할 위치를 입력할 수 있는 문장이 출력되도록 한다.
        Scanner scan = new Scanner(System.in);
        System.out.print("몇 번째 항의 결과를 찾고 싶나요? : ");
        int end = scan.nextInt(); // 정수를 입력할 수 있음.

        //2. 매 항마다 숫자가 x2씩 커지는 배열을 만들어준다.
        //2-1. 이 때, 1항의 숫자는 1이므로 arr[0] = 1;은 직접 입력해준다.

        arr = new int[end];
        arr[0] = 1;

        //2-2. for문을 이용하여 반복문이 내가 입력하는 수 만큼만 arr를 배열하도록 만든다.

        int num = 1;

        for(int i = 1; i < end; i++){
            arr[i] = num * 2;
            System.out.printf("arr[%d] = %d\n", i, arr[i]);

            num = arr[i]; //for 문 맨 위로 돌아가 반복하기 전, num의 값을 바꿔준다.
        }

        //꼭 31번째 항(0포함 32번째)부터 -가 붙어서 나옵니다.
    }
}
```

복습 결과

```
//복습 포인트: 1. final int 사용해보기!
//                2. 1항도 검색하면 나올 수 있도록 해보기!

//31번째 항 부리는 숫자가 제대로 나오지 않으므로
final int MAX = 31;
final int START = 0;

int[] arr;

//1. import Scanner를 사용해 내가 구할 위치를 입력할 수 있는 문장이 출력되도록 한다.
Scanner scan = new Scanner(System.in);
System.out.print("몇 번째 항의 결과를 찾고 싶나요? : ");
int end = scan.nextInt(); // 정수를 입력할 수 있음.

//2. 매 항마다 숫자가 x2씩 커지는 배열을 만들어준다.
//2-1. 이 때, 1항의 숫자는 1이므로 arr[0] = 1;은 직접 입력해준다.

arr = new int[end];

//2-2. for문을 이용하여 반복문이 내가 입력하는 수 만큼만 arr를 배열하도록 만든다.

if(end > MAX){ //찾고자 하는 항이 MAX보다 큰 경우
    System.out.println("출력값이 너무 커 계산할 수 없습니다!");
} else{
    for(int i = 0; i < end; i++){
        arr[i] = (int) Math.pow(2, i); //1에 2를 제곱해준다.
        System.out.printf("arr[%d] = %d\n", i, arr[i]);

        //배열이니까 따로 값을 백업해줄 변수가 없어도 arr[i]의 값이 알아서 백업이 되는거구나?
    }

    //꼭 31번째 항(0포함 32번째)부터 -가 붙어서 나옵니다.
}
}
```

0항에서 1의 값은 0인데 제공하면 1이 나오는 이유...
실제로 java 내에서 0은 딱 떨어지는 숫자가 아니라 0.nnnnn인 건가요?

<문제은행[3] - 8번, Answer8 보며 다시 풀이하기(1)>

선생님 답 풀이 중 의문점(1)

```
// 각 플레이어들의 주사위 값을 비교해서 특수 주사위를 굴릴지 여부 판정
for (int i = 0; i < PLAYER_NUM; i++) {

    if (diceSum[i] % 2 == 0) { // diceSum[i] 값이 짝수인 경우만 if문 조건을 만족한다.
        dice = (int) (Math.random() * range + MIN);

        if (dice == SKILL_NUM1) {
            System.out.println("상대방 주사위 눈금을 2 뺄군다.");

            for (int j = 0; j < PLAYER_NUM; j++) {
                if (j == i) {
                    continue; // skip의 의미임
                }
                diceSum[j] -= 2; //
            }
        }
        else if (dice == SKILL_NUM2) {
            System.out.println("모두 함께 자폭 ^^ ->");

            for (int j = 0; j < PLAYER_NUM; j++) {
                diceSum[j] -= 0;
            }
        }
    }
}
```

이와 같은 스킵이라는 말을...
SKILL_NUM1을 발동시킨 플레이어의 주사위 합에도 -2되면 안되니까
continue로 처리하는 건가요?

diceSum[j]안의 j가 어떤 식으로 작동하는 것인지 아직 이해가 잘 가지 않습니다.
처음으로 옥면체 주사위 2개를 굴려서 주사위값 두 개를 더한 변수는
dice[i], dice[i+1] 위치에 저장된 것 아닌가요?
어떻게 주사위를 처음 굴렸을 때 쓴 적이 없는 dice[j]에 -=2가 적용될 수 있는 건지 모르겠어요.
dice[i]였던 것이 왜 조건식을 특수주사위를 굴리지 못한 나머지 dice[j]로 처리되는 걸까요?

그리고 여기에선 왜 변수 두 개가 다 적용되는 거지... dice[i] = dice[j]라고 볼 수 있는 걸까요?
dice 배열인 게 중요한 거고 인덱스 인지는 항목 번호를 찾아주는 것 뿐이라 상관이 없는건가요?
변수 이름은 고유한 거니까 이유를 잘 모르겠어요...
똑같은 내용물을 잠시 다른 박스에 담았을 뿐이라고 생각하면 되는 걸까요?

코드 전문을 보셔야 확인이 더 편하실텐데 8번 문제 코드가 길어서
한 페이지에 들어가질 않아서 잘랐습니다 πππ

continue란? 반복문 끝으로 이동하여 다음 반복으로 넘어가게 해주는 역할

선생님 답 풀이 중 의문점(2) 특수 주사위 돌아간 이후

```
// 음수 처리
for (int i = 0; i < PLAYER_NUM; i++) {
    if (diceSum[i] < 0) {
        diceSum[i] = 0; // int의 음수를 표현하지 못하는 것은 아니지만 음이 0으로 변경해주는 이유가 궁금해요!
    }
}
```

→ int도 음수값을 표현할 수 있을텐데
왜 음수인 숫자들을 0으로 변경해주는 건가요??

〈문제은행[3] - 8번, Answer8 보며 다시 풀이하기(2)〉

복습 도중 발생한 의문점

그런데 for문 안에 있는 if문 조건에는 배열 인덱스?를 넣을 수 있는데
반복문 밖에서는 안되네요.
위에서 이미 firstDiceSum[i](플레이어1)과 [i+1]플레이어2 의 주사위 값 결과가
배열에 저장되어서 사용할 수 있을 거라고 생각했는데
왜 반복문 밖에선 안되는 걸까요?

처음 구상했던 구조

처음으로 굴린 육면체주사위 2개 값을 더한 결과가
조커문1 짝수여야만 들어갈 수 있는 조건식을 만든다

육면체 주사위 1개를 다시 굴린다.

if / else if / else문으로
특수 주사위 스킬을 작동시킨다.

```

C:\Users\Administrator\.jdk\azul-11.0.13\bin\java.exe -javaagent:C:\
1회차. 플레이어[1]의 주사위 값은 = 10
1회차. 플레이어[2]의 주사위 값은 = 5
꼭사까 나온 플레이어[1]는 육면체 주사위를 한 번 더 굴릴 수 있습니다!
2회차. 플레이어[1]의 주사위 값은 = 10
꼭스수철4. 플레이어[1]의 가 모든 상태의 눈금을 3씩 빼앗습니다.
꼭사까 나온 플레이어[2]는 육면체 주사위를 한 번 더 굴릴 수 있습니다!
2회차. 플레이어[2]의 주사위 값은 = 2
꼭스수철1. 플레이어[2]의 가 상태의 주사위 눈금을 2 점어뜨립니다.
최종 합계. 플레이어[1]의 주사위 값은 = 11
최종 합계. 플레이어[2]의 주사위 값은 = 2

```

그래서 for문 안에 짝수 주사위값을 판별하는 조건식을 넣고
그 안에 특수 주사위 스킬을 작동시킬 수 있도록 했는데 (선생님의 풀이와 같은 방식)
결과를 출력해보면
추가 주사위를 굴린다 > 특수 스킬이 발동한다 > 만약 이로 인해 점수가 깎여 기존에
추가주사위를 굴릴 수 없었던 플레이어의 주사위 눈이 짝수가 될 경우 이 녀석도 갑자기
추가로 주사위를 굴린다.
라는 문제가 발생되어요... 이걸 어떻게 고치면 좋을걸까요...

문제[4] - 5,6,7,9,10. 학생 클래스 만들기

```
public void addSubject () {  
    // 실제로는 Java의 Collection에 있는 Map이나 Vector를 사용해야 하는데  
    // 여기서 직접 자료구조를 구현하는것도 예바고  
    // 그렇다고 문제를 안 풀수도 없으니 일종의 꼼수로서  
    // 백업 배열을 하나 만들어서 Java의 Collection 역할을 하게 만든다.  
    String[] backup = new String[ARR_MAX];  
  
    System.out.print("다루고 싶은 과목을 입력하세요: ");  
  
    // 보편적으로 지지분한 코드는 어느 시점에 루프를 멈출지를 결정하는 부분에서 만들어진다.  
    // 그러므로 루프를 언제 멈출지 판정하는 부분을 매서드화하면  
    // 상대적으로 깔끔한 코드를 얻을 수 있다.  
    do {  
        backup[cnt++] = scan.nextLine();  
  
        // 현재 다시 입력하세요를 구현하는 부분에서 boolean 변수가 추가됨에 따라  
        // 코드의 가독성을 저해하는 부분이 있어  
        // 기본적인 로직 작성후 매서드화 시키는 부분이 좋다 판단하여 삭제 처리  
    } while (checkInput());  
  
    scoreName = new String[cnt];  
  
    for (int i = 0; i < cnt; i++) {  
        scoreName[i] = backup[i];  
    }  
}
```

선생님 답 풀이 중 의문점
Score 클래스

scan부분의 출력을 마치고 yOrn 중 n을 누르면 **score = null** 이 나오며 종료되어서 **cnt**값은 어디로 어떻게 입력하여 정해주는 것인지 코드를 보면서 잘 이해가 가지 않습니다.

그리고 아직 **toString()**이 어떤 역할을 하는 것인지 잘 이해가 안돼요.
일종의 **return**문을 만드는 방식 중 하나라고 생각하면 되는 걸까요?