

## Java方向编程题答案

day24

[编程题]36487-小易的升级之路

链接: <https://www.nowcoder.com/questionTerminal/fe6c73cb899c4fe1bdd773f8d3b42c3d>

### 【题目解析】

无

### 【解题思路】

该题目主要从三部分开始:

- 第一: 循环输入, 怪兽数量, 小易的初始化技能值
- 第二: 每一组开始, 输入怪兽防御值, 进行PK
- 第三: PK结果, 如果怪兽防御值 $\leq$ 小易的技能值, 小易技能值加怪兽防御值; 如果怪兽防御值 $>$ 小易技能值, 小易技能值加怪兽防御值和小易技能值的最大公约数

### 【代码示例】

```
import java.util.Scanner;

public class Main {
    public static void main(String[] args) {
        Scanner scan = new Scanner(System.in);
        int num = 0;
        int ability = 0;
        while (scan.hasNext()) {
            num = scan.nextInt(); //怪物的数量
            ability = scan.nextInt(); //小易初始化技能值
            for (int i = 0; i < num; ++i) {
                int monster = scan.nextInt();
                if (monster <= ability) { //怪兽防御小于等于技能, 技能加怪兽的防御值
                    ability += monster;
                } else { //怪兽防御大于技能, 技能加两者的最大公约数
                    ability += GCD(ability, monster);
                }
            }
            System.out.println(ability);
        }
        scan.close();
    }
}
```

/\*

辗转相除法, 又名欧几里德算法, 是一种非常高效的求最大公约数 (GCD) 的算法, 它无需分解两个整数的质因子, 辗转相除法的递推公式是 $GCD(a, b) == GCD(b, a \% b)$ , 当 $a \% b == 0$ 时,  $b$ 即为所求的最大公约数

\*/

```

public static int GCD(int a, int b) {
    int temp;
    while (b > 0) {
        temp = b;
        b = a % b;
        a = temp;
    }
    return a;
}
}

```

[编程题]25367-最高分是多少

链接: <https://www.nowcoder.com/questionTerminal/3897c2bcc87943ed98d8e0b9e18c4666>

### 【题目解析】

该题目描述内容比较多，但是逻辑并不复杂。简化一下内容即为：

- 第一行：n 个学生 m 老师可以询问或者更新m次
- 第二行：1~ n 个学生的成绩，学生ID也从 1 ~ n
- m 行：
  - 老师操作 Q 询问指定ID范围的学生成绩，输出最高成绩
  - 老师操作 U 更新指定ID的学生成绩

### 【解题思路】

参考题目解析，解题思路变得简单很多。

- 交互式输入：学生数量，老师可操作次数，学生成绩
- 循环执行m次，如果输入Q进行最大分数计算，如果输入U进行分数更新

### 【示例代码】

```

import java.util.Scanner;

public class Main {

    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        int n = 0;
        int m = 0;
        while (scanner.hasNext()) {
            n = scanner.nextInt(); //学生数量
            m = scanner.nextInt(); //老师可以操作的次数
            int[] stu = new int[n]; //学生成绩
            for (int i = 0; i < n; i++) {
                stu[i] = scanner.nextInt();
            }
            //老师询问
            for (int i = 0; i < m; i++) {
                String c = scanner.next();

                int a = scanner.nextInt();

```

```
int b = scanner.nextInt();
if (c.equals("Q")) {
    int s = Math.min(a, b); //开始下标
    int e = Math.max(a, b); //结束下标
    int max = stu[s - 1];
    //计算出 [s-1 , e)范围的最大值
    for (int index = s; index < e; index++) {
        max = Math.max(max, stu[index]);
    }
    System.out.println(max);
}
if (c.equals("U")) {
    stu[a - 1] = b;
}
}
}
}
```