

勾股数元组

题目描述：

如果 3 个正整数 (a,b,c) 满足 $a^2 + b^2 = c^2$ 的关系，则称 (a,b,c) 为勾股数（著名的勾三股四弦五），为了探索勾股数的规律，我们定义如果勾股数 (a,b,c) 之间两两互质（即 a 与 b ， a 与 c ， b 与 c 之间均互质，没有公约数），则其为勾股数元祖（例如 $(3,4,5)$ 是勾股数元祖， $(6,8,10)$ 则不是勾股数元祖）。请求出给定范围 $[N,M]$ 内，所有的勾股数元祖。

输入描述：

起始范围 N ， $1 \leq N \leq 10000$

结束范围 M ， $N < M \leq 10000$

输出描述：

1. a,b,c 请保证 $a < b < c$,输出格式: $a\ b\ c$;
2. 多组勾股数元祖请按照 a 升序， b 升序，最后 c 升序的方式排序输出；
3. 给定范围中如果找不到勾股数元祖时，输出” NA”。

补充说明：

示例

示例 1

输入：

1

20

输出：

3 4 5

5 12 13

8 15 17

说明：

$[1, 20]$ 范围内勾股数有： $(3\ 4\ 5)$ ， $(5\ 12\ 13)$ ， $(6\ 8\ 10)$ ， $(8\ 15\ 17)$ ， $(9\ 12\ 15)$ ， $(12\ 16\ 20)$ ；其中，满足 (a,b,c) 之间两两互质的勾股数元祖有： $(3\ 4\ 5)$ ， $(5\ 12\ 13)$ ， $(8\ 15\ 17)$ ；按输出描述中顺序要求输出结果。

示例 2

输入：

5

10

输出：

NA

说明：

[5, 10]范围内勾股数有：(6 8 10)；其中，没有满足 (a,b,c) 之间两两互质的勾股数元祖；给定范围中找不到勾股数元祖，输出”NA”。

```
import java.util.*;

/**
 * @author tanlicong
 * @since 2023/6/18
 */
public class Main {
    public static void main(String[] args) {
        Scanner in = new Scanner(System.in);
        int n = in.nextInt();
        int m = in.nextInt();
        List<int[]> tupleList = new ArrayList<>();
        for (int a = n; a < m; a++) {
            for (int b = a + 1; b < m; b++) {
                // 判断 a,b 是否互质
                if (getMaxGys(a, b) != 1) {
                    continue;
                }
                int cc = a * a + b * b;
                int c = (int) Math.sqrt(cc);
                if (c <= m && c * c == cc) {
                    // 判断 a,c;b,c 是否互质
                    if (getMaxGys(a, c) == 1 && getMaxGys(b, c) == 1) {
                        tupleList.add(new int[]{a, b, c});
                    }
                }
            }
        }
        if (tupleList.isEmpty()) {
            System.out.println("NA");
        } else {
            // 从小到大枚举 a,b 因此无需排序
            tupleList.forEach(tuple -> System.out.println(tuple[0] + " " + tuple[1]
+ " " + tuple[2]));
        }
    }
}
```

```
}

/**
 * 辗转相除法求两数的最大公约数
 * @param number1
 * @param number2
 * @return
 */
public static int getMaxGys(int number1, int number2) {
    if (number1 % number2 == 0) {
        return number2;
    } else if (number2 % number1 == 0) {
        return number1;
    }
    return getMaxGys(number2, number1 % number2);
}
}
```