

## JAVA- 求最小步数

### 题目描述：

求从坐标零点到坐标点 `n` 的最小步数，一次只能沿横坐标轴向左或向右移动 2 或 3 。

> 注意：途径的坐标点可以为负数

### 输入描述：

坐标点 n

### 输出描述：

输出从坐标零点移动到坐标点 n 的最小步数

### 补充说明：

$1 \leq n \leq 10^9$

### 示例 1

#### 输入：

4

#### 输出：

2

#### 说明：

从坐标零点移动到 4，最小需要两步，即右移 2，再右移 2

```
import java.util.Scanner;
import javax.script.ScriptEngine;
import javax.script.ScriptEngineManager;
import javax.script.ScriptException;
import java.lang.reflect.Array;
import java.math.BigDecimal;
import java.math.BigInteger;
import java.util.*;
import java.util.stream.Collectors;
// 注意类名必须为 Main，不要有任何 package xxx 信息
public class Main {
    public static void main(String[] args) {
        ques2Up(args);
    }

    public static void ques2Up(String[] args) {

        Scanner in = new Scanner(System.in);
        // 注意 hasNext 和 hasNextLine 的区别
        int end = Integer.valueOf(in.nextLine());
```

```
end = Math.abs(end);
```

```
Map<Integer, Integer> dpMap = new HashMap<>(100);
```

```
int dp[] = new int[4];
```

```
dp[0] = 0;
```

```
dp[1] = 2;
```

```
dp[2] = 1;
```

```
dp[3] = 1;
```

```
if(end <= 3){
```

```
    System.out.println(dp[end]);
```

```
    return;
```

```
}
```

```
dp = new int[3];
```

```
dp[0] = 2;
```

```
dp[1] = 1;
```

```
dp[2] = 1;
```

```
for(int i = 4; i < end + 1; i++){
```

```
    int min = Math.min(dp[0] + 1, dp[1] + 1);
```

```
    dp[0] = dp[1];
```

```
    dp[1] = dp[2];
```

```
    dp[2] = min;
```

```
}
```

```
System.out.println(dp[2]);
```

```
}
```

```
}
```