264. Ugly Number II

Medium 763 53 Favorite Share

Write a program to find the `n`-th ugly number.

Ugly numbers are **positive numbers** whose prime factors only include `2, 3, 5`.

Example:

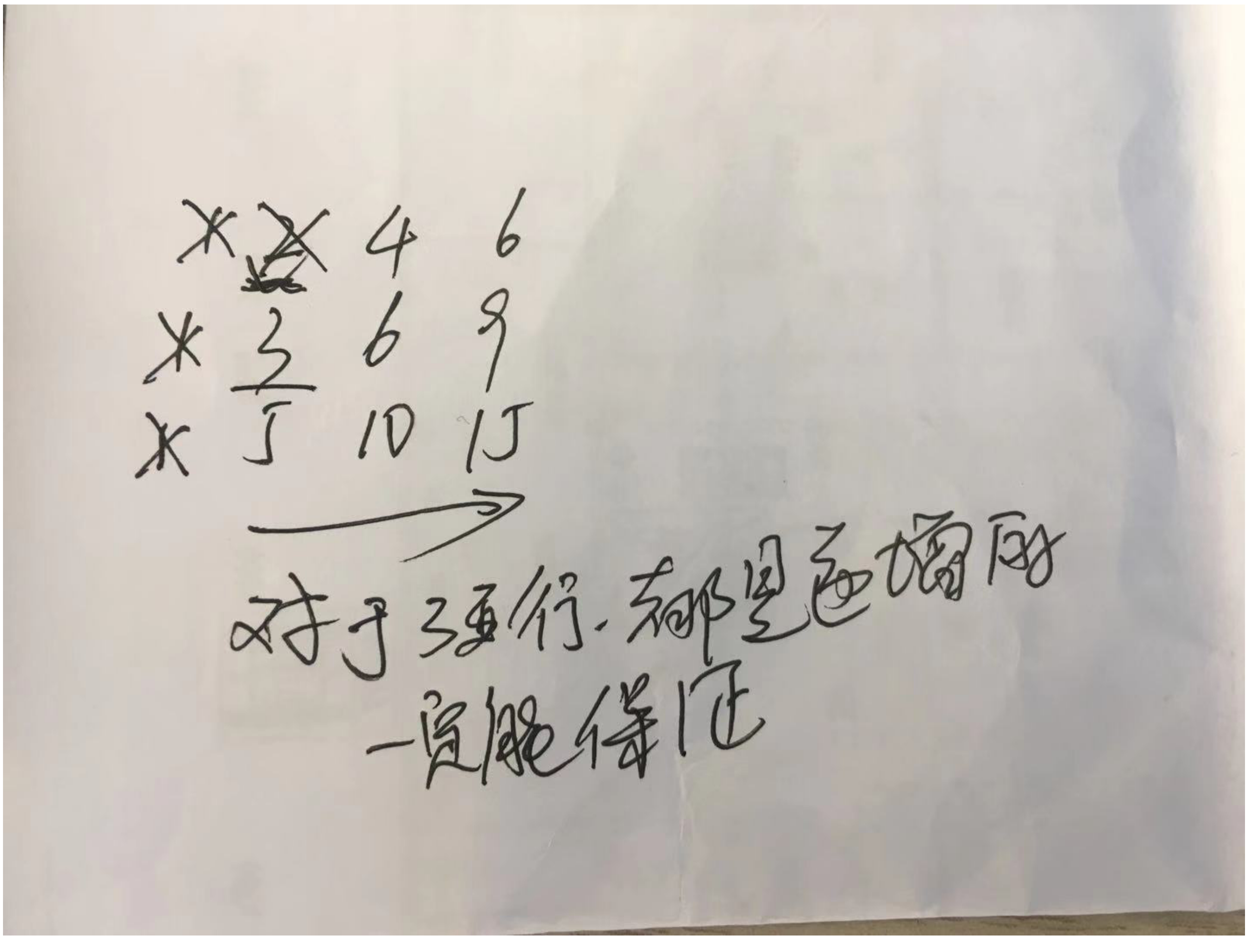
```
Input: n = 10
Output: 12
Explanation: 1, 2, 3, 4, 5, 6, 8, 9, 10, 12 is the sequence of the first 10
ugly numbers.
```

Note:

1. `1` is typically treated as an ugly number.
2. `n` **does not exceed 1690**.

Accepted 96,582 | Submissions 271,891

Seen this question in a real interview before? Yes No

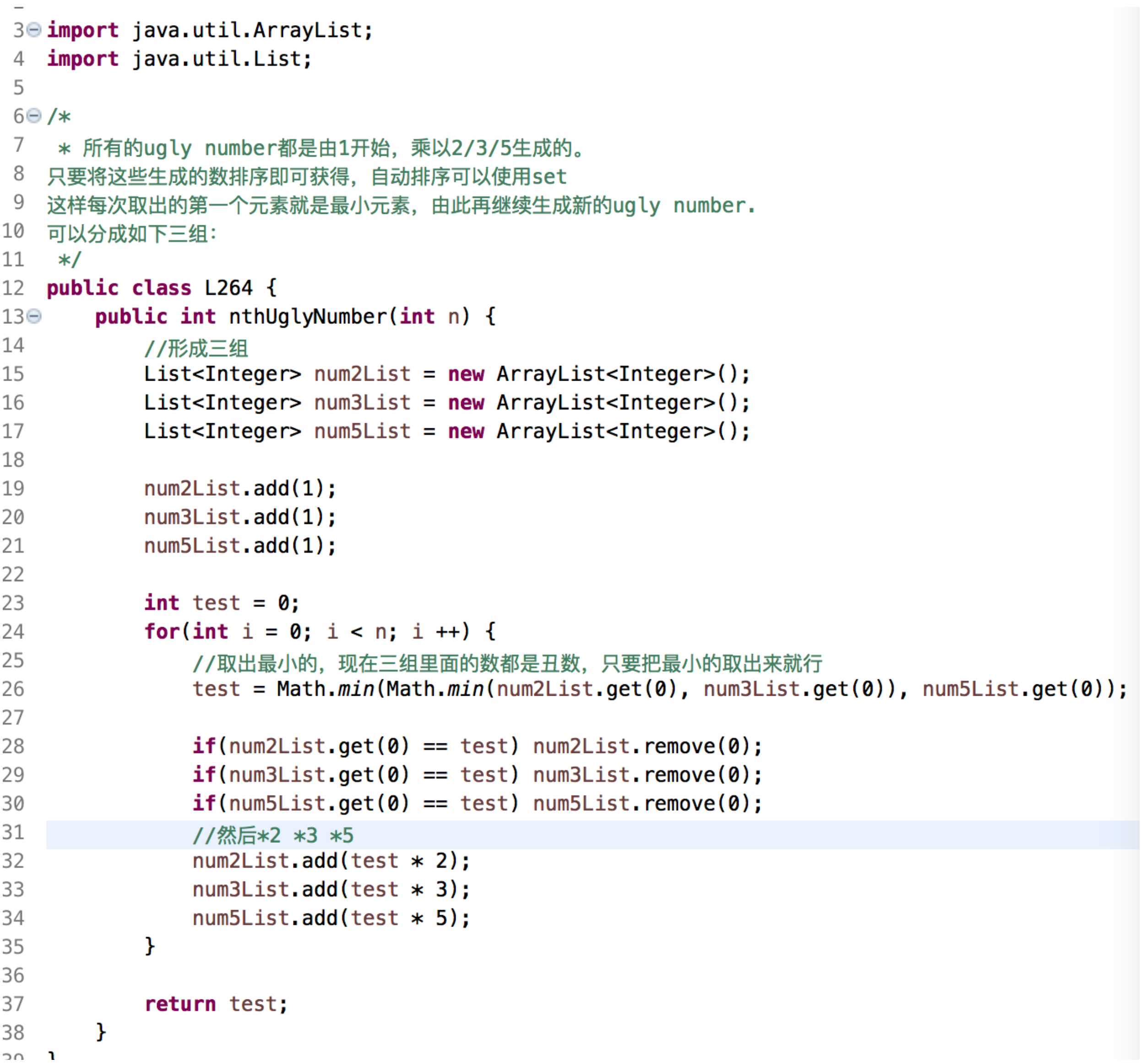


~~1~~ ~~2~~ 4 6

~~1~~ 3 6 9

~~1~~ 5 10 15

对于3个行，都是在增加
一直能保证



```java
import java.util.ArrayList;
import java.util.List;

/*
 * 所有的ugly number都是由1开始，乘以2/3/5生成的。
只要将这些生成的数排序即可获得，自动排序可以使用set
这样每次取出的第一个元素就是最小元素，由此再继续生成新的ugly number.
可以分成如下三组：
 */
public class L264 {
    public int nthUglyNumber(int n) {
        //形成三组
        List<Integer> num2List = new ArrayList<Integer>();
        List<Integer> num3List = new ArrayList<Integer>();
        List<Integer> num5List = new ArrayList<Integer>();

        num2List.add(1);
        num3List.add(1);
        num5List.add(1);

        int test = 0;
        for(int i = 0; i < n; i ++) {
            //取出最小的，现在三组里面的数都是丑数，只要把最小的取出来就行
            test = Math.min(Math.min(num2List.get(0), num3List.get(0)), num5List.get(0));

            if(num2List.get(0) == test) num2List.remove(0);
            if(num3List.get(0) == test) num3List.remove(0);
            if(num5List.get(0) == test) num5List.remove(0);
            //然后*2 *3 *5
            num2List.add(test * 2);
            num3List.add(test * 3);
            num5List.add(test * 5);
        }

        return test;
    }
}
```