第K个丑数\_剑指Offer\_49

# 第K个丑数\_剑指Offer\_49

## 题目描述

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*剑指Offer49：丑数\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

\* 把只包含质因子2、3和5的数称作丑数（Ugly Number）。

\* 例如6、8都是丑数，但14不是，因为它包含质因子7。

\* 习惯上我们把1当做是第一个丑数。求按从小到大的顺序的第N个丑数。

## 思路分析

\* 思路分析:定义一个长度为k的数组，依次存放从小到大的丑数。

\* 定义4个索引，nextIndex用于依次添加丑数，另外3个是2/3/5对应的索引，

\* 如indeFor2用于标记乘以2后，小于等于此时最大丑数的索引，下次再次乘以2只需要

\* 从该处乘以2既可以了。indexFor3、indexFor5同理。

\* minOfThree从三个数中选择最小的，三个参数正好是刚刚大于前一个丑数的丑数，

\* 从三者中选择最小的作为下一个丑数。

\*/

## Java代码

public int **GetUglyNumber**\_**Solution**(int k) {

if(k <= 0) return 0;//默认为0

if(k == 1) return 1;

int[] uglyNums = new int[k];

uglyNums[0] = 1;//第一个丑数为1

int indexFor2 = 0,indexFor3 = 0,indexFor5 = 0;

**int nextIndex = 1;**

while(nextIndex < k){

//选择最小的作为下一个丑数

**uglyNums[nextIndex] = minOfThree(uglyNums[indexFor2]\*2,uglyNums[indexFor3]\*3,uglyNums[indexFor5]\*5);**

while(uglyNums[indexFor2]\*2 <= uglyNums[nextIndex]) indexFor2++;//indexFor2前进

while(uglyNums[indexFor3]\*3 <= uglyNums[nextIndex]) indexFor3++;

while(uglyNums[indexFor5]\*5 <= uglyNums[nextIndex]) indexFor5++;

nextIndex++;//下一个丑数索引

}

return uglyNums[k-1];

}

//获取三个数中的最小值

public int minOfThree(int a,int b,int c){

int min = a < b?a:b;

return min < c ? min :c;

}