

250712因数（题解）

题目链接：

<https://www.matiji.net/exam/brushquestion/70/4009/C448715ED43BEA9D2D47CED523050945?from=1>

给定一个整数 n ，定义 n 的一个因子 y 为 k 次因子，当且仅当存在因子 x ，满足 $x^k=y$ 。注意，这里的 x,y 可以相等。

现在知道了数字 n,k ，请你输出 n 的所有 k 次因子。由于因子 1 比较特殊，它是任何数字的任意 k 次因子。所以对于输入的整数 n ，当只有因子 1 满足要求时请输出 `'onlyone'`。

输入格式：

一行输入两个数 $n,k(1\leq n\leq 2000,1\leq k\leq 10)$

输出格式：

在一行中从小到大输出 n 的所有 k 次因子。如果只有因子 1 满足要求，则输出 `'onlyone'`。

输入：

12 1

输出：

1 2 3 4 6 12

输入：

81 2

输出：

1 9 81

输入：

13 2

输出：

onlyone

解法一：

问题理解：

给定两个整数 n 和 k ，我们需要找出 n 所有的因子 y ，使得存在一个整数 x 满足 $x^k = y$ （即 y 是某个整数的 k 次幂），并且 y 也是 n 的因子。

条件分析：

- 枚举 n 的所有因子 y ；

- 检查是否存在某个整数 x , 使得 $x^k == y$;
 - 若有多个符合条件的因子, 按从小到大输出;
 - 若只有 1 满足, 输出 `onlyone` 。
 - 遍历 $i = 1$ 到 n , 如果 $n \% i == 0$, 说明 i 是因子;
 - 判断是否存在整数 x , 使得 $x^k == i$;
 - 由于 x 最大不会超过 i 的开 k 次根, 所以只需枚举 x 从 1 到 $\text{pow}(i, 1.0/k)$;
 - 用一个 `vector` 存储所有符合条件的因子 y ;
 - 特判: 如果结果只有 1 , 输出 `onlyone` 。
-

```
#include <bits/stdc++.h>
using namespace std;

// 判断某个数 y 是否是某个整数的 k 次幂, 即是否存在 x 满足  $x^k == y$ 
bool isKthPower(int y, int k) {
    // 从 x=1 开始枚举, 直到  $x^k$  超过 y
    for (int x = 1; pow(x, k) <= y; ++x) {
        // 判断  $\text{pow}(x, k)$  是否刚好等于 y
        if ((int)pow(x, k) == y) {
            return true; // 是 k 次幂
        }
    }
    return false; // 不存在这样的 x
}

int main() {
    int n, k;
    cin >> n >> k; // 输入整数 n 和 k

    vector<int> res; // 存储满足条件的因子

    // 遍历所有可能的因子 i (从1到n)
    for (int i = 1; i <= n; ++i) {
        // 判断 i 是否是 n 的因子, 且 i 是某个整数的 k 次幂
        if (n % i == 0 && isKthPower(i, k)) {
            res.push_back(i); // 把这个符合条件的因子加入结果中
        }
    }

    // 特殊情况: 结果中只有 1, 说明只有1满足要求
    if (res.size() == 1 && res[0] == 1) {
        cout << "onlyone\n"; // 特判输出
    } else {
        sort(res.begin(), res.end()); // 按从小到大排序
    }
}
```

```
// 输出所有满足条件的因子
for (int i = 0; i < res.size(); ++i) {
    cout << res[i];
    if (i < res.size() - 1) cout << " "; // 空格分隔
}
cout << endl;
}

return 0;
}
```