

PAT 甲级刷题实录——1015

作者: [aopstudio](#)

原文链接: <https://ld246.com/article/1580199037576>

来源网站: [链滴](#)

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)

原题链接

<https://pintia.cn/problem-sets/994805342720868352/problems/994805495863296000>

思路

这题要求判断一个数在翻转前后是否都是质数。有一个难以理解的地方是在给定进制下的翻转，这是转成该进制表示的数之后翻转，再将翻转后的数转换回十进制判断。比如 11 在二进制下的翻转，首先转换为二进制表示形式 1011，之后翻转为 1101，再转换回十进制为 13 判断。另外，将十进制转换其他进制的方法是用原数不断去除以进制数，并将每次得到的余数倒序记录下来，而由于我们还需要翻转一次，所以在计算时直接依次将余数存在 vector 中最终得到的就是在给定进制下翻转后的数。

代码如下

代码

```
#include <iostream>
#include <vector>
using namespace std;

int main()
{
    int num, radix, reverse;
    vector<int> result; //1代表Yes, 0代表No
    cin >> num;
    while (num >= 0)
    {
        reverse = 0; //用于存储翻转后的数
        if (num < 2) //1和0
        {
            result.push_back(0);
            cin >> num;
            continue; //直接进入下一次循环
        }
        int flag = 1;
        cin >> radix;
        for (int i = 2; i < num; i++) //判断原数是否为质数
        {
            if (num%i == 0)
            {
                flag = 0;
                break;
            }
        }
        vector<int> radixNum;
        while (num / radix > 0)
        {
            radixNum.push_back(num%radix);
            num /= radix;
        }
        radixNum.push_back(num);
        int exp = 1;
```

```

for (int i = radixNum.size() - 1; i >= 0; i--)
{
    reverse += radixNum[i] * exp;
    exp *= radix;
}
if (reverse < 2)    //1和0
{
    result.push_back(0);
    cin >> num;
    continue;    //直接进入下一次循环
}
for (int i = 2; i < reverse; i++)    //判断翻转数是否为质数
{
    if (reverse % i == 0)
    {
        flag = 0;
        break;
    }
}
result.push_back(flag);
cin >> num;
}
for (int i = 0; i < result.size(); i++)
{
    if (result[i] == 1)
        cout << "Yes" << endl;
    else
        cout << "No" << endl;
}
return 0;
}

```