

2.能被 3,5,7 整除的数

问题分析：怎么实现不同情况，若一个数能被 3,5,7 同时整除，则输出 3,5,7.若一个数只能被 3,5 整除，则只输出 3,5.这道题的精髓就在于不同情况输出不同的值。

解决方案：我们可以给不同的情况设置一个独特的 case 值，当 case 值为这个值时就输出相应与之配对的编程语言和输出。

编程实现：

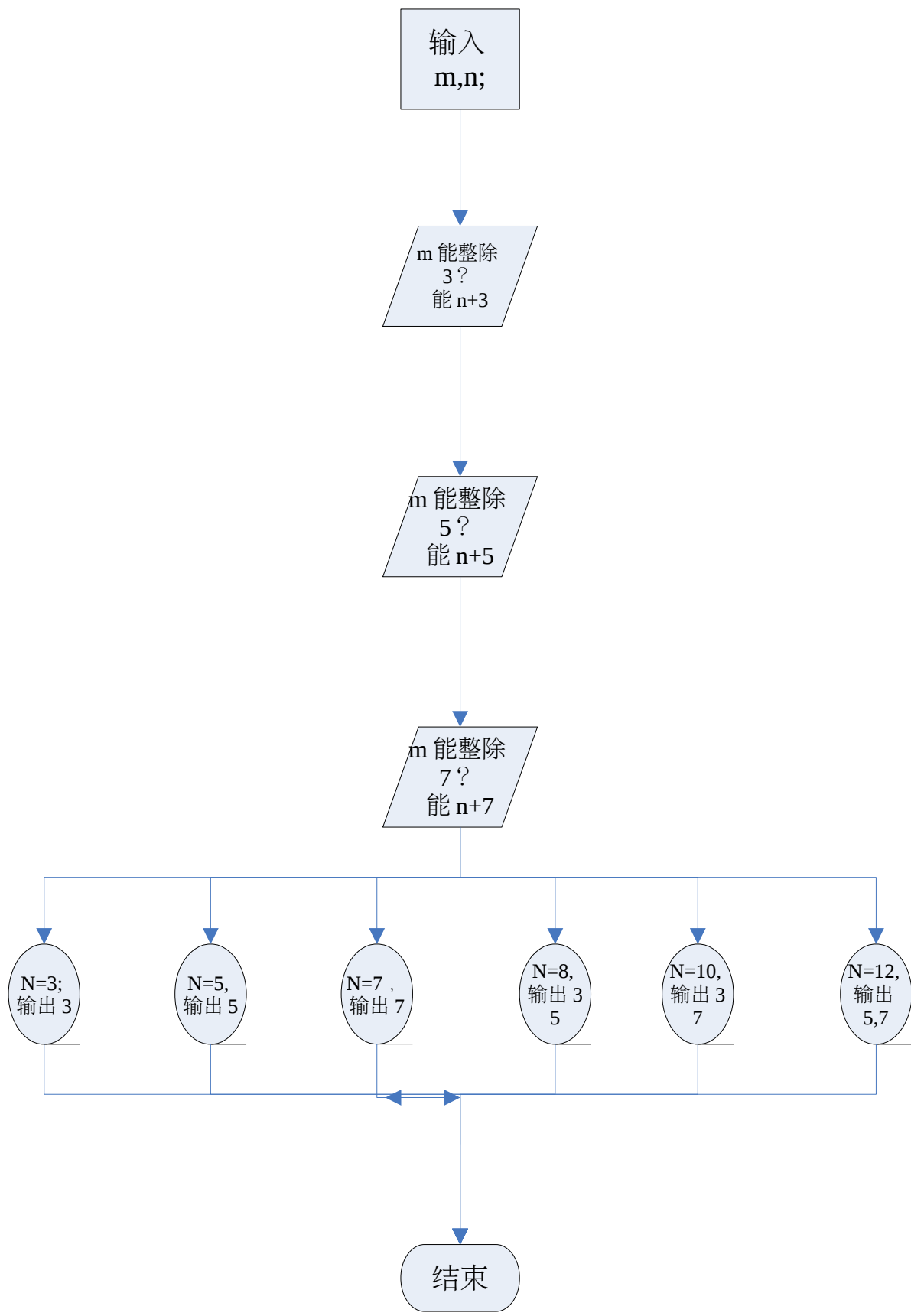
```
/*  
> File Name: 能被 3,5,7 整除的数.cpp  
> Author: zhengziqiang  
> Mail: 1174986943@qq.com  
> Created Time: 2016 年 04 月 17 日 星期日 19 时 19 分 22 秒  
*/
```

```
#include<iostream>  
using namespace std;  
int main(){  
    int m;  
    while(cin>>m){  
        int n=0;  
        if(m%3==0)n+=3;  
        if(m%5==0)n+=5;  
        if(m%7==0)n+=7;  
        switch(n){  
            case 3:cout<<"3"<<endl;break;  
            case 5:cout<<"5"<<endl;break;  
            case 7:cout<<"7"<<endl;break;  
            case 8:cout<<"3 "<<"5"<<endl;break;  
            case 10:cout<<"3 "<<"7"<<endl;break;  
            case 12:cout<<"5 "<<"7"<<endl;break;  
            case 15:cout<<"3 "<<" "<<"5 "<<"7"<<endl;break;  
        }  
    }  
    return 0;  
}
```

通过 openjudge.

总结体会：这道题主要考察的是不同的情况输出不同的值，而为了解决这道题，我给不同的情况赋给不同的 case 值.所以执行 switch 语句时，我只要考虑在不同的 case 值下的输出就可以了。

流程图：



4 简单计算器

问题分析： 给定三个输入值，前两个为整数，后一个为运算符包括+ - * /，我们要做的就是输入这三个值后能得到结果，起到一个简单计算器的作用。

解决方案： 也是使用 switch 语句，分情况进行加减乘除运算。与上一道题不同的是我们要考虑进行除法运算时除数不能为 0.当输入的运算符不为加减乘除时。输出 invalid operator.

编程实现：

```
#include<iostream>
using namespace std;
int main()
{
    int m,n;
    char a;
    cin>>m>>n;
    cin>>a;
    switch(a){

        case '*':
            cout<<m*n<<endl;break;
        case '+':
            cout<<m+n<<endl;break;
        case '-':
            cout<<m-n<<endl;break;
        case '/':
            {
                if(n==0)
                {
                    cout<<"Divided by zero!"<<endl;
                }
                else
                    cout<<m/n<<endl;
                break;
            }
        default:cout<<"Invalid operator!"<<endl;
    }
    return 0;
}
```

总结体会：

这道编程题与 2 题不同的是它要多考虑一些情况，当输入的运算符不为加减乘除时输出 invalid operator.并且除数不能为 0.

通过 openjudge

流程图：

