

完美数

对于一个 **正整数**，如果它和除了它自身以外的所有 **正因子** 之和相等，我们称它为「完美数」。

给定一个 **整数** n ，如果是完美数，返回 `true`，否则返回 `false`

示例 1:

输入: 28

输出: True

解释: $28 = 1 + 2 + 4 + 7 + 14$

1, 2, 4, 7, 和 14 是 28 的所有正因子。

示例 2:

输入: `num = 6`

输出: `true`

示例 3:

输入: `num = 496`

输出: `true`

示例 4:

输入: `num = 8128`

输出: `true`

示例 5:

输入: `num = 2`

输出: `false`

暴力筛选法会超过时间限制，对此采用枚举法:

N=28 从 1 到 `sqrt(28)` 遍历 找 1 2 4 对应 28 14 7

`Sum=1+28+2+14+4+7==2*28`

```
/*
```

在枚举时，我们只需要从 1 到 \sqrt{n} 进行枚举即可。这是因为如果 n 有一个大于 \sqrt{n} 的因数 x ，那么它一定有一个小于 \sqrt{n} 的因数 n/x 。因此我们可以从 1 到 \sqrt{n} 枚举 n 的因数，当出现一个 n 的因数 x 时，我们还需要算上 n/x 。此外还需要考虑特殊情况，即 $x = n/x$ ，这时我们不能重复计算

```
*/
bool checkPerfectNumber(int num){
    int sum=0;
    for(int i=1;i<sqrt(num);i++)
    {
        if(num%i==0)
        {
            sum+=i;
            if(i*i!=num)
            {
                sum+=num/i;
            }
        }
    }
    if(sum==num*2)
    {
        return true;
    }
    return false;
}
```