

# 问题求解与实践 ——数论相关问题

主讲教师： 陈雨亭、沈艳艳

# 数论相关问题

## 数论问题范围很广

- 整数的整除性
- 带余除法
- 奇数与偶数
- 约数与倍数
- 质数与合数
- 整数的分解与分拆
- 进制转换
- .....

最小公倍数问题

求小于 $n$ 的质数

# 求两个数的最小公倍数

假设求 8 和 20 的最小公倍数

## 方法一：列举法

从小到大先找出8的倍数，再判断这个数是否是20的倍数，从中找出最小的、公共的一个。

考察：8, 16, 24, 32, .....

## 方法二：分解质因数法

分别把两个数分解质因数，8和20公倍数里，应当既包含8的所有质因数，又包含20的所有质因数。对于两个数共有的质因数，比如2，在8中出现3次，在20中出现2次，则取3；再把独有的质因数都取出来，把它们相乘，积就是最小公倍数，具体做法如下：

$$8 = 2 \times 2 \times 2$$



$$20 = 2 \times 2 \times 5$$

$$8 \text{ 和 } 20 \text{ 的最小公倍数是：} 2 \times 2 \times 2 \times 5 = 40$$

# 求两个数的最小公倍数

## 方法一：列举法的伪代码描述

```
A=8, B=20
```

```
k = A
```

```
i=1
```

```
while ( k%B ! = 0 )
```

```
{
```

```
    k = i*A
```

```
    i=i+1
```

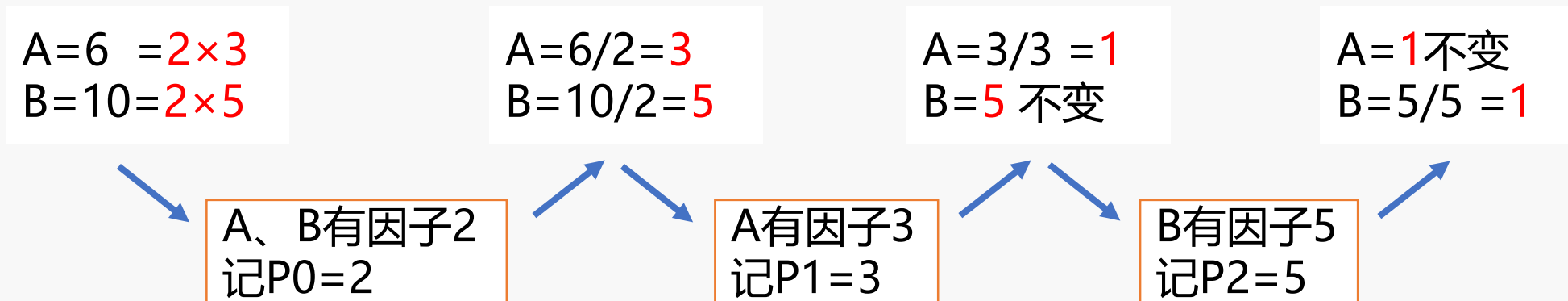
```
}
```

```
输出k
```

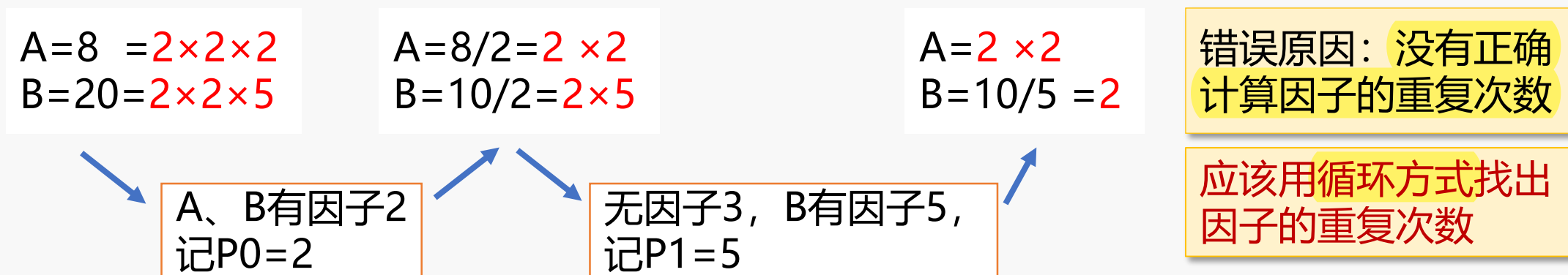
# 求两个数的最小公倍数

## 方法二：分解质因数法实现分析

按照**从小到大**的顺序查找质因数，即判断A、B是否有因子2, 3, 5, 7, .....



$$\text{最小公倍数} = P0 \times P1 \times P2 = 30$$

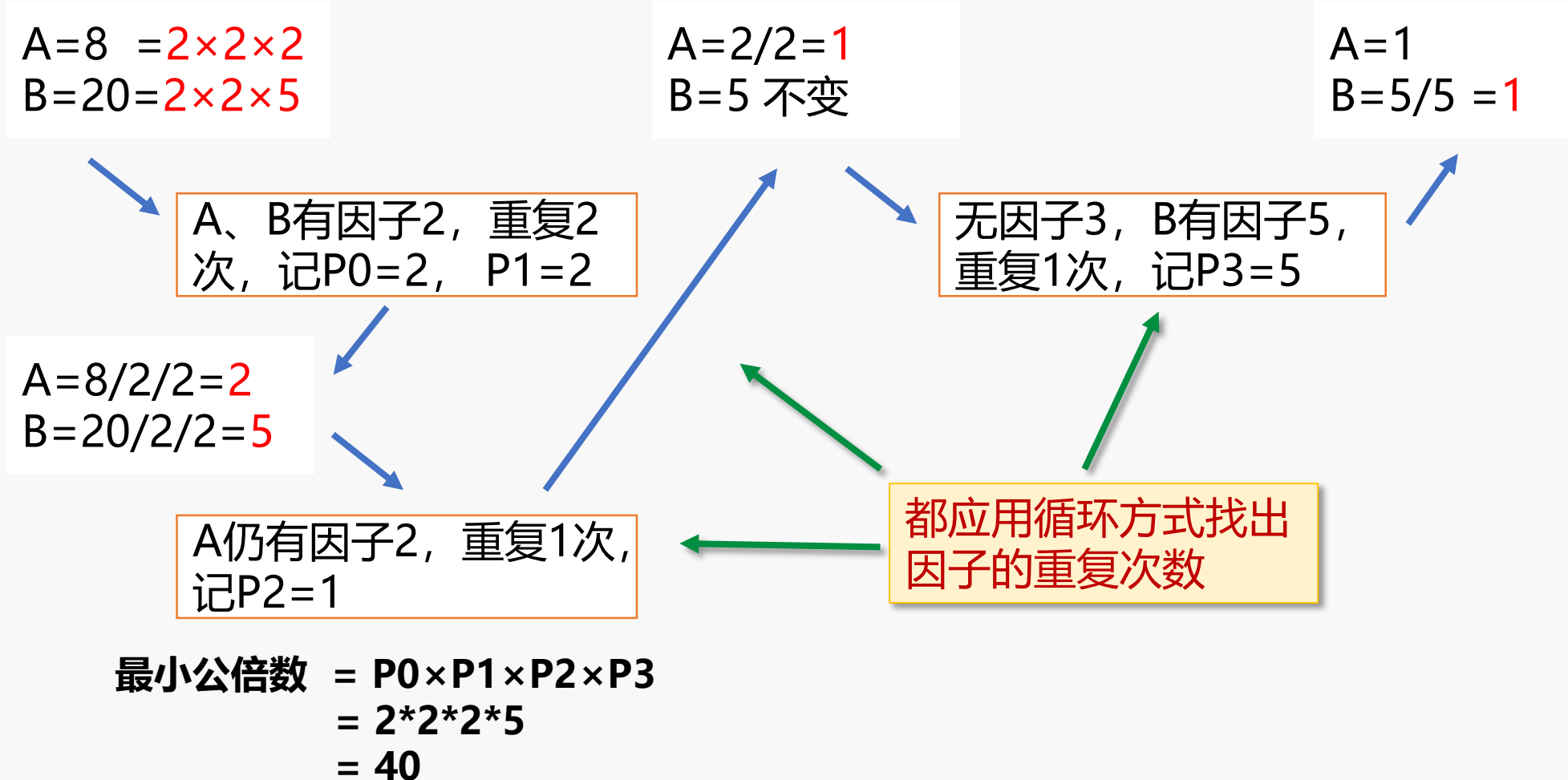


$$\text{最小公倍数} ? = P0 \times P1 = 10$$

# 求两个数的最小公倍数

## 方法二：分解质因数法算法实现分析

按照从小到大的顺序查找质因数，即判断A、B是否有因子2, 3, 5, 7, .....



# 求两个数的最小公倍数

## 方法二：分解质因数法算法描述

```
i = 0, C = max ( A,B )    // p[i]记录所有因子  
for( k=2; k<=C; k++ ) {
```

```
    while (k是A和B的因子) {  
        p[i] = k;  
        A=A/k; B=B/k; i++;  
    }
```

```
    while (k是A的因子) {  
        p[i] = k; A=A/k;  
        i++;  
    }
```

```
    while (k是B的因子) {  
        p[i] = k; B=B/k;  
        i++;  
    }
```

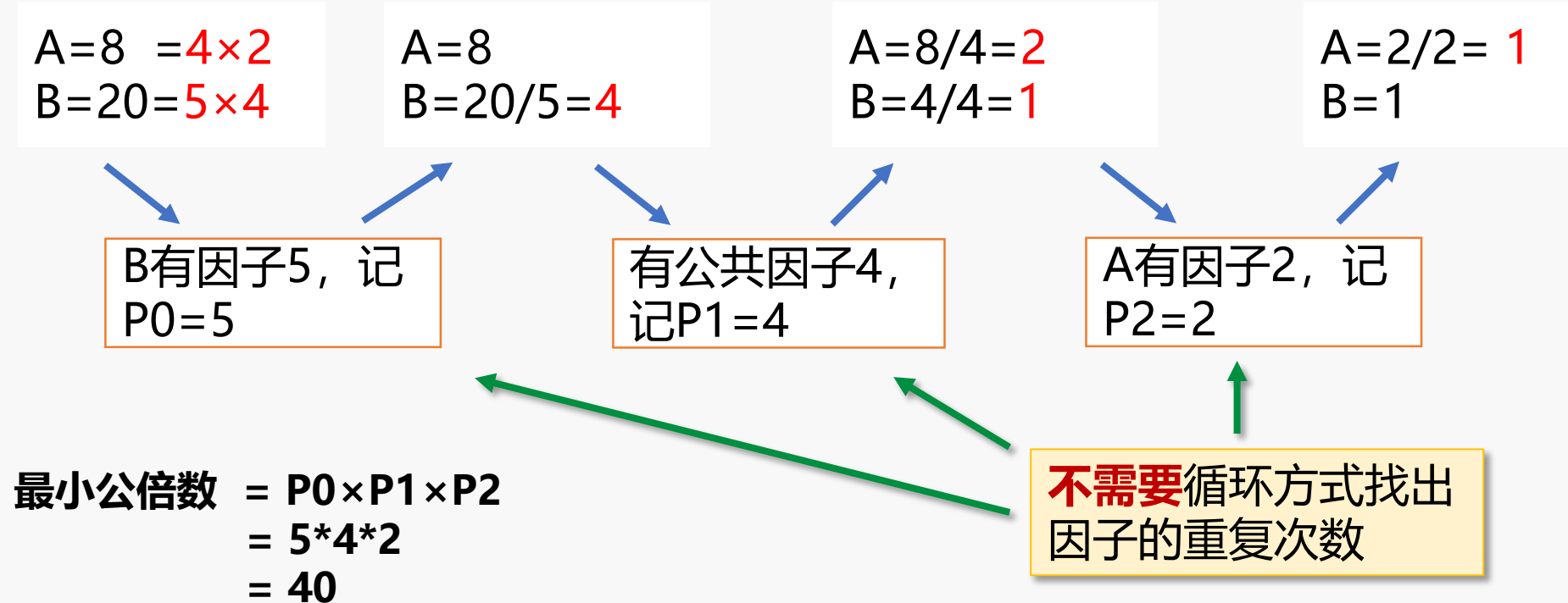
```
}
```

计算 $p[0]*p[1]*...*p[i-1]$ 得到最小公倍数

# 求两个数的最小公倍数

## 方法三：改进的分解因数法算法

- 不求**质因数**，而是求**普通的因数**
- 按照**从大到小**的顺序查找因数，**共同的因子只取一次**（乘到最小公倍数中）





# 求小于N的所有质数

问题：输入正整数N，请输出小于N的所有质数

## ■ 通用方法

```
for (k 从 2 到 N-1)
{  判断 k 是否是质数, 若是则输出  }
```



```
for (i 从 2 到 k/2)
{
    if ( k%i == 0 ) 则 k 是合数, 跳出循环
}
if  以上循环提前结束
    则k 是合数
else
    k 是质数
```

就从2一直尝试到 $k/2$ ,  
考察是否有k的因子

# 求小于N的所有质数

## ■ 筛法计算质数表

特点：能够快速筛选出区间  $[2, N]$  中的所有质数

算法：

在整数区间 $[2, N]$ 内，

- ① 保留 2 同时去掉其他 2 的倍数，
- ② 保留 3 同时去掉其他 3 的倍数，
- ③ 保留 5 同时去掉其他 5 的倍数，
- ④ 保留 7 同时去掉其他的倍数，……，以此类推，一直到所有小于  $N$  的质数的倍数都被去掉，剩余的数字就是质数表

# 求小于N的所有质数

## ■ 筛法计算质数表实例

求出2到20之间的质数

建立一个表[ 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 ]

➤ 第一步，取第一个非零数2（是质数），将其后面2的倍数全部置零(删掉)。得到

[ 2 3 0 5 0 7 0 9 0 11 0 13 0 15 0 17 0 19 0 ]

➤ 第二步，取下一个非零数3（是质数），将其后面3的倍数全部置零(删掉)。得到

[ 2 3 0 5 0 7 0 0 0 11 0 13 0 0 0 17 0 19 0 ]

➤ 第三步，取下一个非零数5（是质数），将其后面5的倍数全部置零(删掉)。得到

[ 2 3 0 5 0 7 0 0 0 11 0 13 0 0 0 17 0 19 0 ]

➤ 第四步，取下一个非零数7（是质数），将其后面7的倍数全部置零(删掉)。得到

[ 2 3 0 5 0 7 0 0 0 11 0 13 0 0 0 17 0 19 0 ]

➤ .....

# 求小于N的所有质数

## ■ 筛法程序实现分析

```
int prime[N];    // 定义一个长度为N的数组，N为前面定义的常数
```

```
void PrimeTable( int N ) {
```

```
    int i, j;
```

```
    for(i = 2; i <= N; i++)    // 建立表[ 2 3 4 5 6 7 8 9 10 .....]
```

```
        prime[i] = i;
```

```
    for(i = 2; i < N; i++)  
    {
```

```
        if( prime[i] != 0 )
```

```
            for(j = i+i; j <= N; j+=i)    // 将2*i, 3*i, 4*i ..... 置零
```

```
                prime[j] = 0;
```

```
    }
```

```
}
```