

## Số hoàn hảo - Euclid

Nhà toán học Euclid đã chứng minh rằng các số hoàn hảo chỉ phải có dạng  $2^{i-1} \times (2^i - 1)$  với  $i=2, 3, \dots$ . Thí dụ:

| i   | $2^{i-1} \times (2^i - 1)$ | Giá trị | là số hoàn hảo (x) |
|-----|----------------------------|---------|--------------------|
| 2   | $2^1 \times (2^2 - 1)$     | 6       | x                  |
| 3   | $2^2 \times (2^3 - 1)$     | 28      | x                  |
| 4   | $2^3 \times (2^4 - 1)$     | 120     |                    |
| 5   | $2^4 \times (2^5 - 1)$     | 496     | x                  |
| 6   | $2^5 \times (2^6 - 1)$     | 2016    |                    |
| ... |                            |         |                    |

Do đó chúng ta để tìm các số hoàn hảo  $\leq n$  với  $n$  lớn ( $10^{15}$ ), chúng ta chỉ cần dùng đoạn lệnh sau:

```
for (i=1; pow(2,i-1) * (pow(2,i)-1) <= n; i++) {  
    test= pow(2,i-1) * (pow(2,i)-1);  
    if soHH(test) cout << test << " ";  
}
```

và hàm chức năng SoHH() như sau:

```
bool SoHH(long long n) {  
    long long TUS=1; // Tổng ước số  
    for (int i=2; i*i<n; i++)  
        if (n % i==0) TUS += i + n/i;  
    if (TUS==n) return true;  
    else return false;  
}
```