

时间机器

有一台时间机器可以根据时间轴正向或逆向调整小时数和分钟数，可以连续调整多次，以便实现到过去或未来的穿越。请你设计**C/C++**程序，根据初始时刻（**24** 小时制）、调整次数、每次想穿越的方向以及想穿越的小时数和分钟数，计算穿越后的时刻。

（ 禁止使用数组，禁止使用**STL**库中提供的函数）

思路： **n次调整等同于1次调整**

小时全部转换为分钟？

最后将时间转为**24**小时制

```
int h0, m0, n;
cin >> h0 >> m0 >> n;
for (int i = 0; i < n; i++) {
    int f, h, m;
    cin >> f >> h >> m;
    h0 += f * h;
    m0 += f * m;
}
int m = (h0 * 60 + m0) % (24 * 60);
if (m < 0)
    m += (24 * 60);
h0 = (m / 60) % 24;
m0 = m % 60;
cout << h0 << " " << m0 << endl;
```

考拉兹猜想

对任意正整数 n 进行以下计算：若 n 为偶数则除以 2，若 n 为奇数则乘以 3 再加 1；然后对计算结果持续进行以上计算；最终计算结果必为 1。即，

根据上述考拉兹猜想，5 经过 5 步得到 1（5 \rightarrow 16 \rightarrow 8 \rightarrow 4 \rightarrow 2 \rightarrow 1），即 5 对应的 k 为 5；12 经过 9 步得到 1（12 \rightarrow 6 \rightarrow 3 \rightarrow 10 \rightarrow 5 \rightarrow 16 \rightarrow 8 \rightarrow 4 \rightarrow 2 \rightarrow 1），即 12 对应的 k 为 9。

请你设计一个C/C++程序，找出正整数区间 $[a,b]$ 中的一个数 x ，其对应的 k 比区间中其他数对应的 k 都大。如果有多个数对应的 k 相等，则输出较小的数。要求输入 a 和 b ，输出 x 及其对应的 k 。

（ 禁止使用数组，禁止使用STL库中提供的函数）

```
int k(int a) {
    int ans = 0;
    while (a > 1) {
        if (a % 2 == 0)
            a = a / 2;
        else
            a = 3 * a + 1;
        ans++;
    }
    return ans;
}
```

```
int main() {
    //n=1时, step取值最小为0, 所以step初始
    化为比下界还小的数, 比如-1
    int a, b, n, step = -1, ans = 1;
    cin >> a >> b;
    for (int i = a; i <= b; i++) {
        n = k(i);
        if (n > step) {
            ans = i;
            step = n;
        }
    }
    cout << ans << " " << step << endl;
    return 0;
}
```