

Euler problem 02

于船长

书山有路勤为径，学海无涯苦作舟

本期内容

- 一. 题目讲解
- 二. 代码演示
- 三. 程序优化

一. 题目讲解

一. 题目讲解

Even Fibonacci numbers

Each new term in the Fibonacci sequence is generated by adding the previous two terms. By starting with 1 and 2, the first 10 terms will be: 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, ...

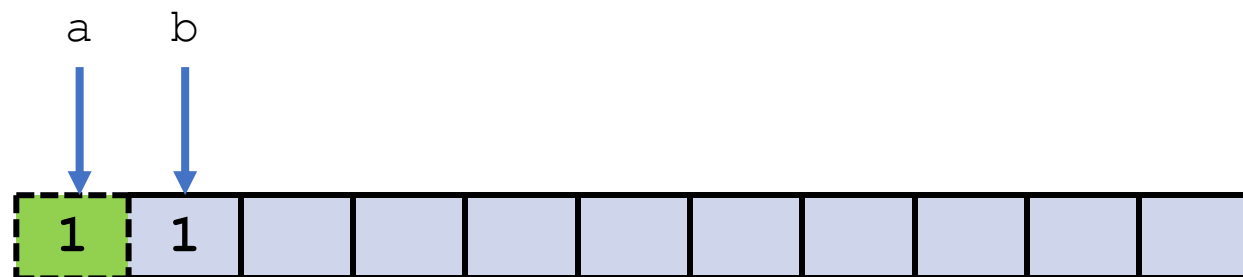
By considering the terms in the Fibonacci sequence whose values do not exceed four million, find the sum of the even-valued terms.

-----偶斐
波那契数

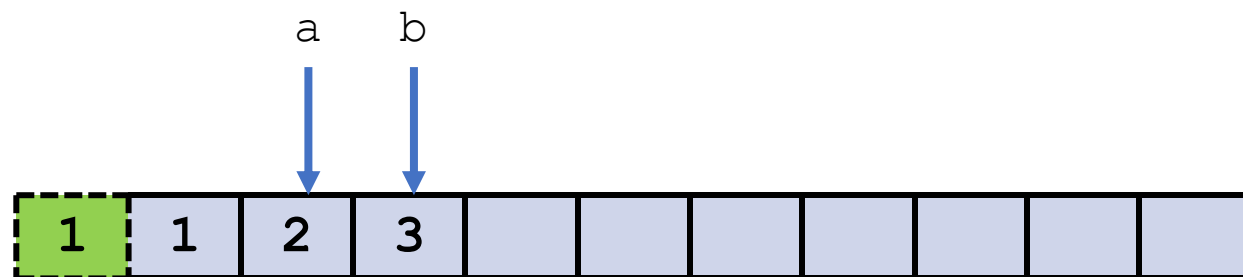
斐波那契数列中的每一项都是前两项的和。由1和2开始生成的斐波那契数列的前10项为：1, 2, 3, 5, 8, 13, 21, 34, 55, 89, ...

考虑该斐波那契数列中不超过四百万的项，求其中为偶数的项之和。

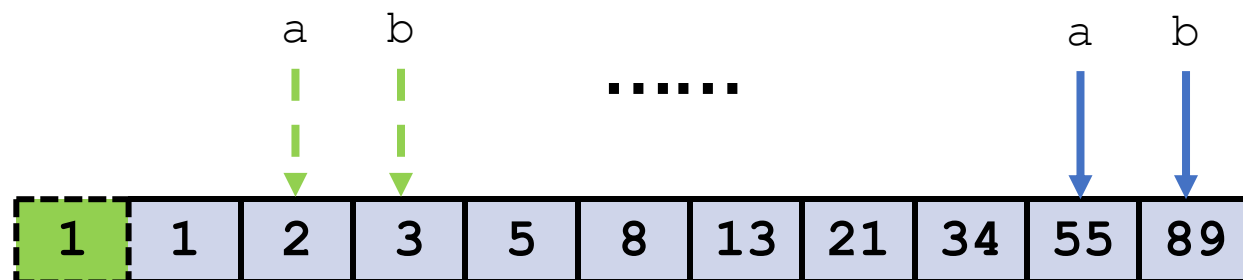
一. 题目讲解



一. 题目讲解



一. 题目讲解



一. 题目讲解

$$\text{fib}[1] = 1$$

$$\text{fib}[2] = 1$$

$$\text{fib}[n] = \text{fib}[n - 1] + \text{fib}[n - 2] \quad (n > 2)$$

一. 题目讲解

解题思路是什么？

1. 循环生成每一个斐波那契数
2. 判断当前的斐波那契数是否是偶数
3. 上述条件成立后，进行加和
4. 输出打印

二. 代码演示

二. 代码演示

```
#include <stdio.h>
#define max_n 4000000
int fib[45];

int main() {
    fib[1] = 1, fib[2] = 2;
    int n = 2;
    while (fib[n - 1] + fib[n] < max_n) {
        n++;
        fib[n] = fib[n - 1] + fib[n - 2];
    }
    long long sum = 0;
    for (int i = 1; i <= n; i++) {
        if (fib[i] % 2) continue;
        sum += fib[i];
    }
    printf("%lld\n", sum);
    return 0;
}
```

Q:程序能否再优化一下?

三. 程序优化

三. 程序优化

真的需要存储每一项斐波那契数的值么？

三. 程序优化

```
#include <stdio.h>
#define max_n 4000000

int fib[3] = {0, 1};

int main() {
    int n = 1;
    int sum = 0;
    while (fib[n % 3] + fib[(n - 1) % 3] < max_n) {
        n += 1;
        fib[n % 3] = fib[(n - 1) % 3] + fib[(n - 2) % 3];
        if (!(fib[n % 3] & 1)) sum += fib[n % 3];
    }
    printf("%d\n", sum);
    return 0;
}
```

Q: 能否用2个变量，完成斐波那契数的生成

三. 程序优化

```
#include <stdio.h>
#define max_n 4000000
int main() {
    int a = 1, b = 2;
    int sum = 2;
    while (a + b < max_n) {
        b = a + b;
        a = b - a;
        if (!(b & 1)) sum += b;
    }
    printf("%d\n", sum);
    return 0;
}
```