fibonacci

2016 年 4 月 4 日

1 斐波纳契数列

查找斐波纳契数列中第 N 个数。
所谓的斐波纳契数列是指：
前 2 个数是 0 和 1 。第 i 个数是第 i-1 个数和第 i-2 个数的和。斐波纳契数列的前 10 个数字是：
0, 1, 1, 2, 3, 5, 8, 13, 21, 34 ...

样例 给定 1，返回 0
给定 2，返回 1
给定 10，返回 34

递归实现

```
In [1]: import time

        class Solution:
            result = 0

            def __init__(self, n):
                self.result = self.fibonacci(n)

            # @param n: an integer
            # @return an integer f(n)
            def fibonacci(self, n):
                # write your code here
                if n == 1:
                    return 0
                elif n == 2:
                    return 1
                else:
                    return self.fibonacci(n-1) + self.fibonacci(n-2)
```

```
        start = time.clock()
        sol = Solution(40)
        end = time.clock()
        print 'result = ', sol.result, "; ", end-start, "s"
```

```
result =  63245986 ;  58.036035 s
```

非递归实现

```
In [2]: import time

        class Solution:
            result = 0

            def __init__(self, n):
                self.result = self.fibonacci(n)

            # @param n: an integer
            # @return an integer f(n)
            def fibonacci(self, n):
                # write your code here
                a, b = 0, 1
                for i in range(1, n):
                    a, b = b, a + b
                return a

        start = time.clock()
        sol = Solution(40)
        end = time.clock()
        print 'result = ', sol.result, "; ", end-start, "s"
```

```
result =  63245986 ;  0.000175999999996 s
```