

# Euler 0002

## Problem

Each new term in the Fibonacci sequence is generated by adding the previous two terms. By starting with 1 and 2, the first 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.

What we know:

- We need to generate the Fibonacci sequence up to 4,000,000.
- We only need to keep every other number in the sequence.

## Considerations

Time-space considerations and decisions to be made:

- We could choose to store 0 numbers and recursively generate all numbers up to 4,000,000.
  - This spends both a lot of time although it doesn't use a large permanent space.
- We could generate all of the Fib. numbers up to 4,000,000 and store them in a list, good for referencing later, and will reduce redundancy.
- We could keep only the last 2 Fib. numbers and keep dropping the last one when we get a new one.

```
fib_1 = 1
fib_2 = 2
limit = 4000000

#running total set to 2 because it is odd
running_total = 2

#keep going until fib_1 and fib_2 totaled > limit
while fib_1 + fib_2 < limit:
    fib = fib_1 + fib_2
    fib_1 = fib_2
    fib_2 = fib
```

```
#add to the running total if it is even.
```

```
if fib % 2 == 0:
```

```
    running_total += fib
```

---

Revision #3

Created 15 June 2025 16:56:53 by Maxwell

Updated 24 June 2025 12:42:14 by Maxwell