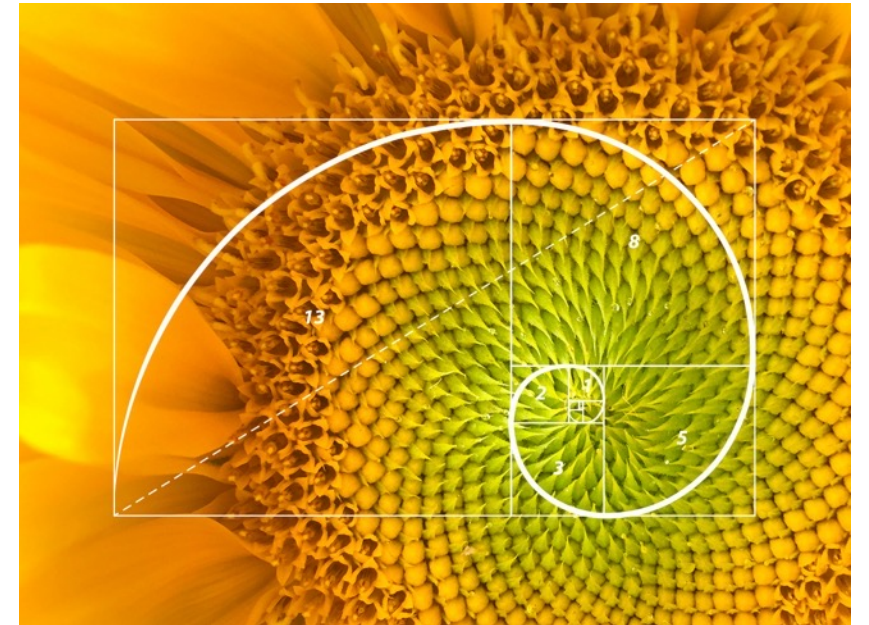


CPSC 5031 - Algorithms

Algorithm Review via the Fibonacci Sequence

Outcomes

- Understand the Fibonacci Sequence
- Understand the many ways to calculate it
 - Recursive
 - Iterative
 - Better recursive
 - Simple function

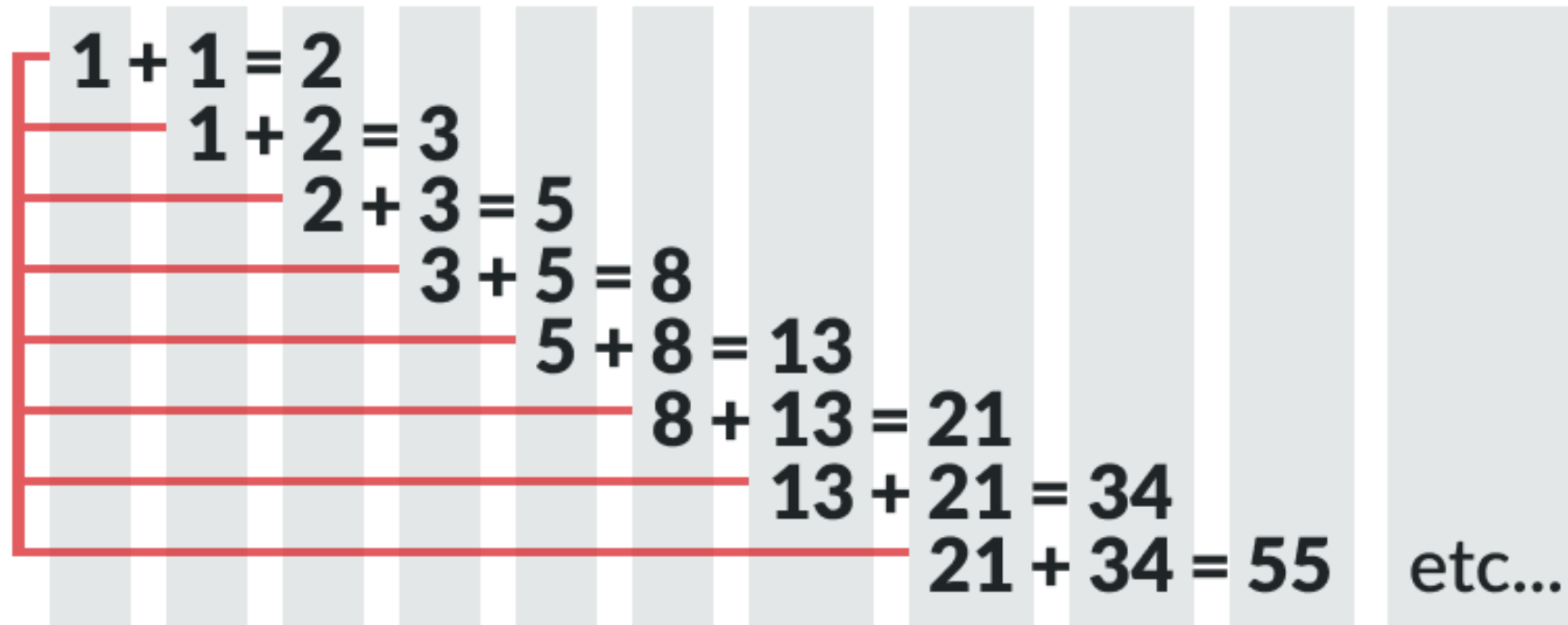


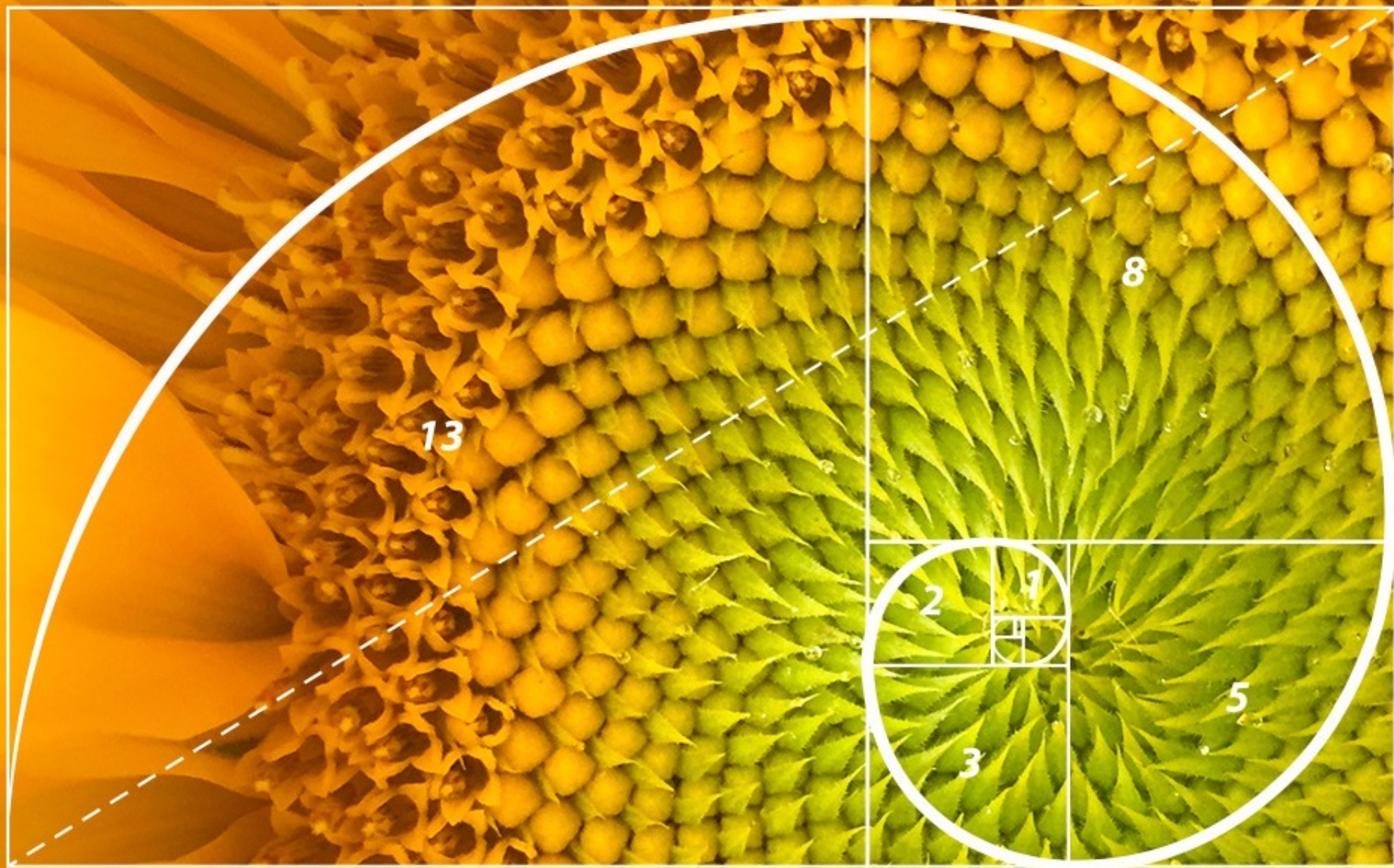
Fibonacci

$$f(n) = \begin{cases} 0 & \text{if } n = 0 \\ 1 & \text{if } n = 1 \\ F(n-1) + F(n-2) & \text{if } n > 1 \end{cases}$$

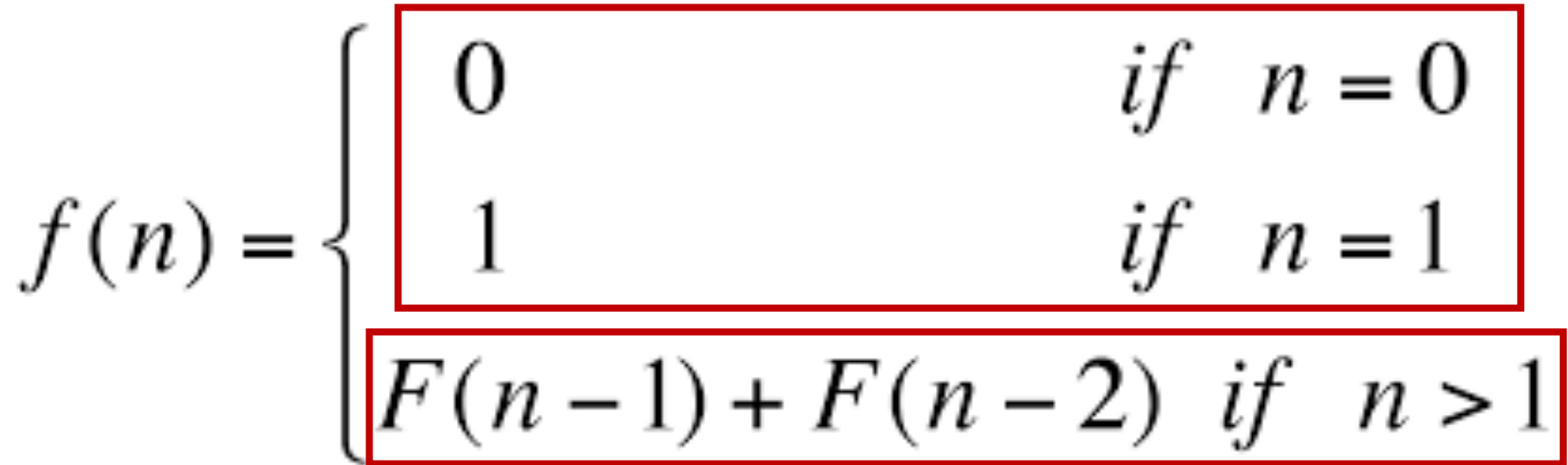
- Take a few minutes and write some code...
 - Any language, any approach

Fibonacci





Fibonacci

$$f(n) = \begin{cases} 0 & \text{if } n = 0 \\ 1 & \text{if } n = 1 \\ F(n-1) + F(n-2) & \text{if } n > 1 \end{cases}$$


- Take a few minutes and write some code...
 - Any language, any approach

Recursive step

Recursive (classic) solution

- Advantages?
- Disadvantages?
 - In Fib(8), how many times is Fib(3) calculated?

```
1  int Fib(int n)
2  {
3      switch (n)
4      {
5          case 0:
6              return 0;
7          case 1:
8              return 1;
9          default:
10             return Fib(n-1) + Fib(n-2);
11     }
12 }
```

Iterative solution

- Advantages?
- Disadvantages?

```
int FibIter(int n)
{
    if (n == 0 || n == 1)
    {
        return n;
    }

    int a = 0;
    int b = 1;
    int c = 1;

    for (int i = 2; i ≤ n; i++)
    {
        c = a + b;
        a = b;
        b = c;
    }

    return c;
}
```


Recursive (more efficient) solution

- Advantages?
- Disadvantages?

```
int FibRecurseAccum(int n, int a = 0, int b = 1)
{
    switch (n)
    {
        case 0:
            return a;
        default:
            return FibRecurseAccum(n - 1, b, a + b);
    }
}
```

Constant time solution

- Advantages?

$$Fib(n) = \frac{A^n - B^n}{\sqrt{5}} \quad \text{where} \quad A = \frac{1 + \sqrt{5}}{2} \quad B = \frac{1 - \sqrt{5}}{2}$$

- Disadvantages?

```
8  my $a = ( 1 + sqrt( 5 ) ) / 2.0;
9  my $b = ( 1 - sqrt( 5 ) ) / 2.0;
10
11  sub fib {
12      my $n = shift @_;
13
14      return floor( ( $a**$n - $b**$n ) / sqrt( 5 ) );
15  }
```

Why are we talking Fibonacci?

- A very simple function...
- ...that can be expressed algorithmically in very different ways.

“An algorithm is a sequence of unambiguous instructions for solving a problem obtaining a required output for any legitimate input in a finite amount of time.”

Homework...

Fun with Fibonacci!