intro to mathematics in software engineering

Fontys University of Applied Sciences

April 23, 2025

objectives

scary looking functions

math related software engineering concepts

translate math to programming

fundamentals of mathematical and function notation

$$f(x) = x^2$$

fundamentals of mathematical and function notation

$a \cdot f(x)$

$f(x/b)$

$f(x - c)$

$f(x) + d$

fundamentals of mathematical and function notation

$a \cdot f(x) \Rightarrow$ multiplies the y-value by a

$f(x/b) \Rightarrow$ multiplies the x-value by b

$f(x - c) \Rightarrow$ shifts graph c units to the right

$f(x) + d \Rightarrow$ shifts graph d units upward

breaking down a scary looking function

$$g(x) = \frac{4}{\pi} \sum_{n=1}^{\infty} \frac{\sin(2\pi(2n-1)ft)}{2n-1}$$

(where $t = \texttt{time}$, $f = \texttt{frequency}$, $n = \texttt{iterations}$)

breaking down a scary looking function

$$g(x) = \frac{4}{\pi} \sum_{n=1}^{\infty} \frac{\sin(2\pi(2n-1)ft)}{2n-1}$$

we are dealing with a function built from multiple smaller functions added together

breaking down a scary looking function

$$g(x) = \frac{4}{\pi} \sum_{n=1}^{\infty} \frac{\sin\left(2\pi(2n-1)ft\right)}{2n-1}$$

the sin on the inside suggests we are dealing with waves or oscillations

breaking down a scary looking function

$$g(x) = \frac{4}{\pi} \sum_{n=1}^{\infty} \frac{\sin(2\pi(2n-1)ft)}{2n-1}$$

denominator $2n-1$ hints that terms get smaller as n increases - later terms have less influence

breaking down a scary looking function

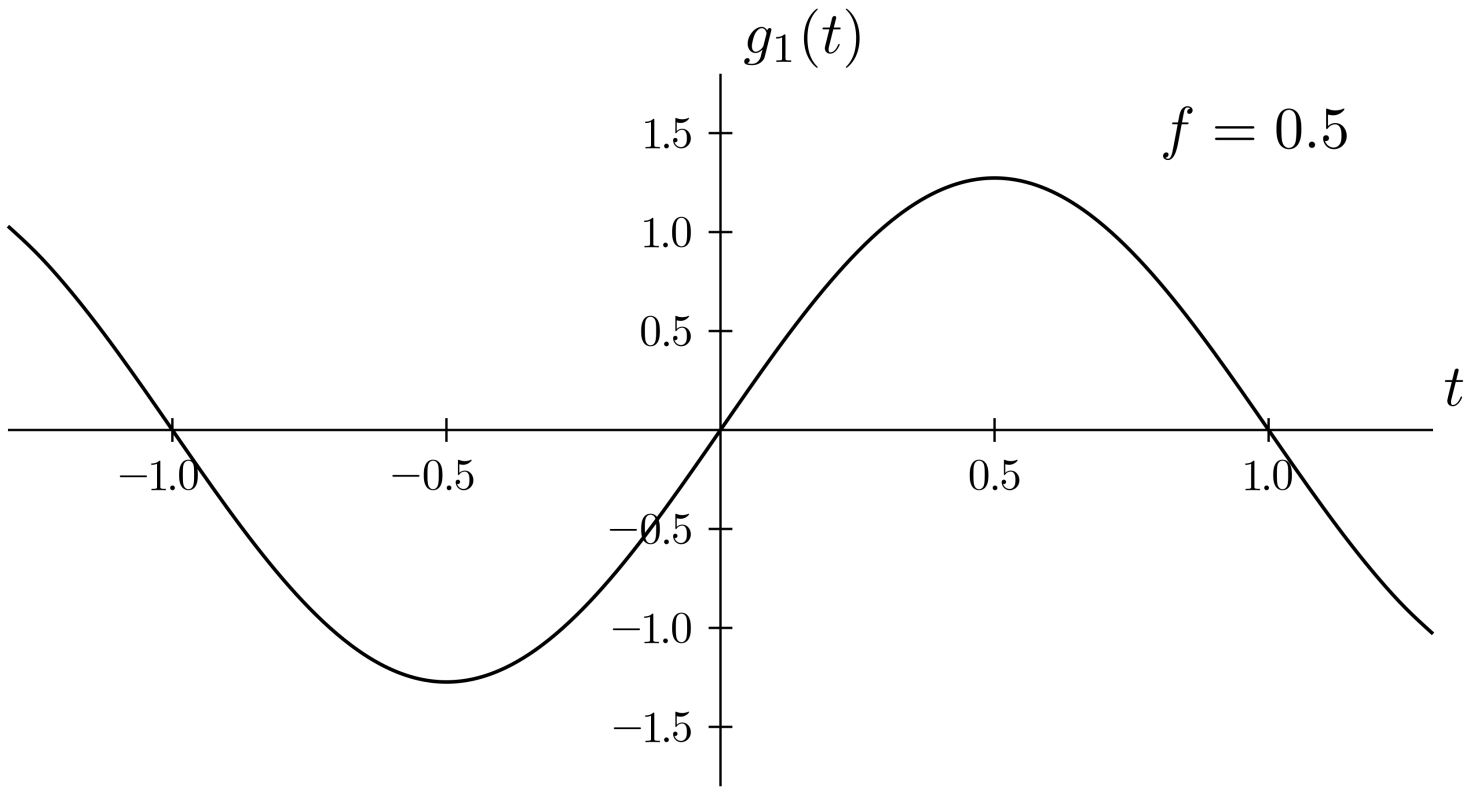
$$g_1(t) = \frac{4}{\pi} \cdot \frac{\sin(2\pi(2(1)-1)ft)}{2(1)-1}$$

$$= \frac{4}{\pi} \cdot \frac{\sin(2\pi(1)ft)}{1}$$

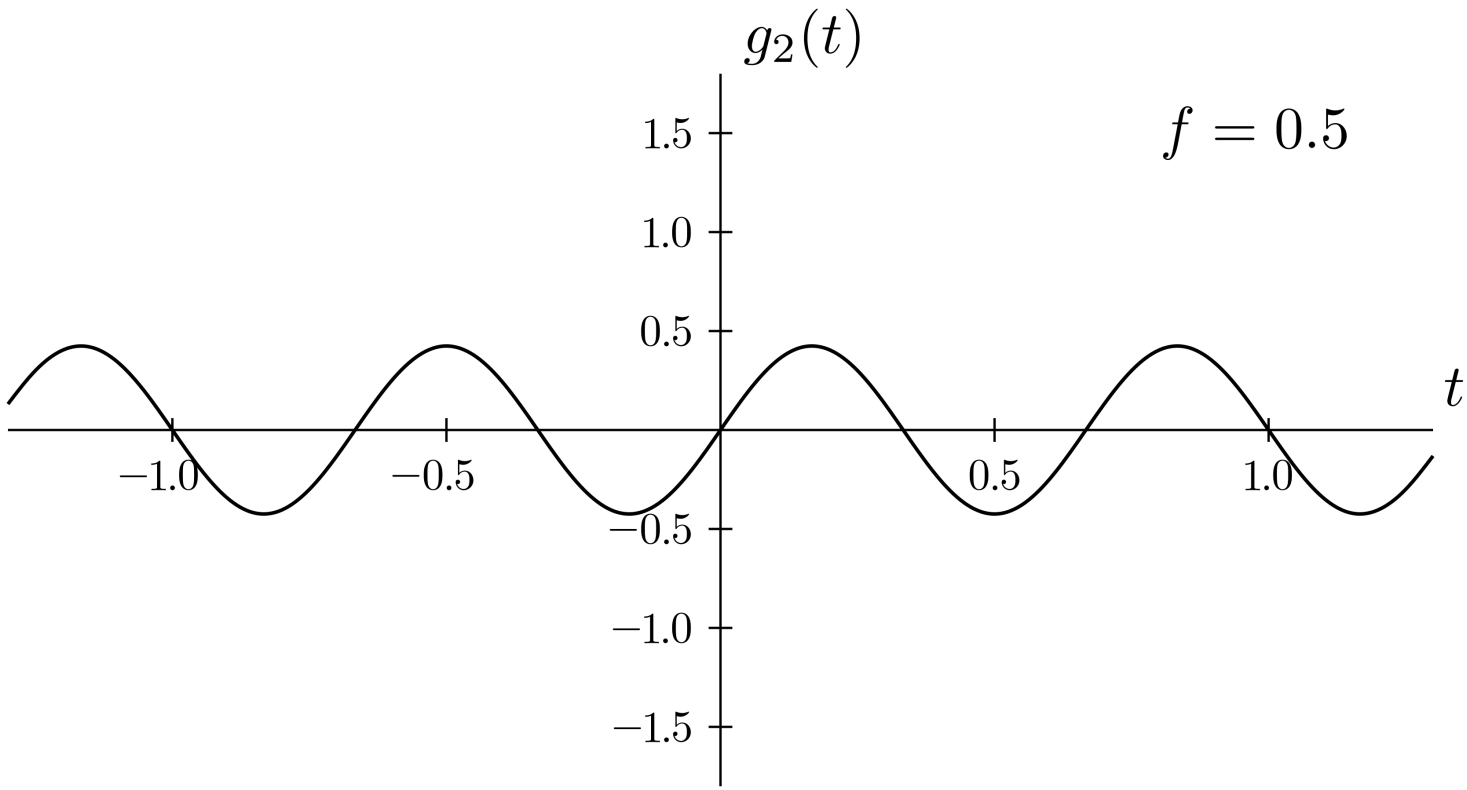
$$= \frac{4}{\pi} \cdot \sin(2\pi ft)$$

$$g_1(t) = \frac{4}{\pi} \sin(2\pi ft)$$

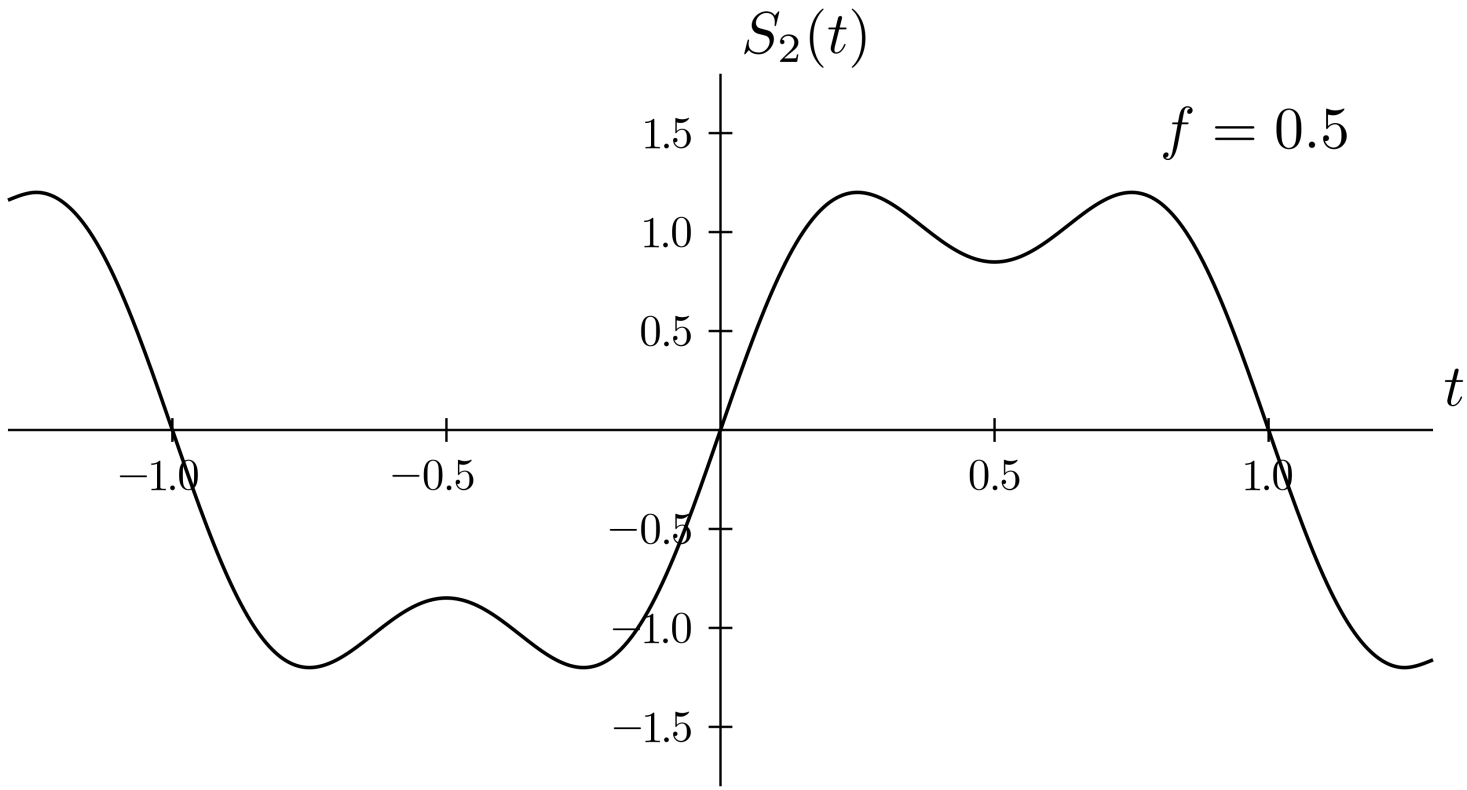
breaking down a scary looking function



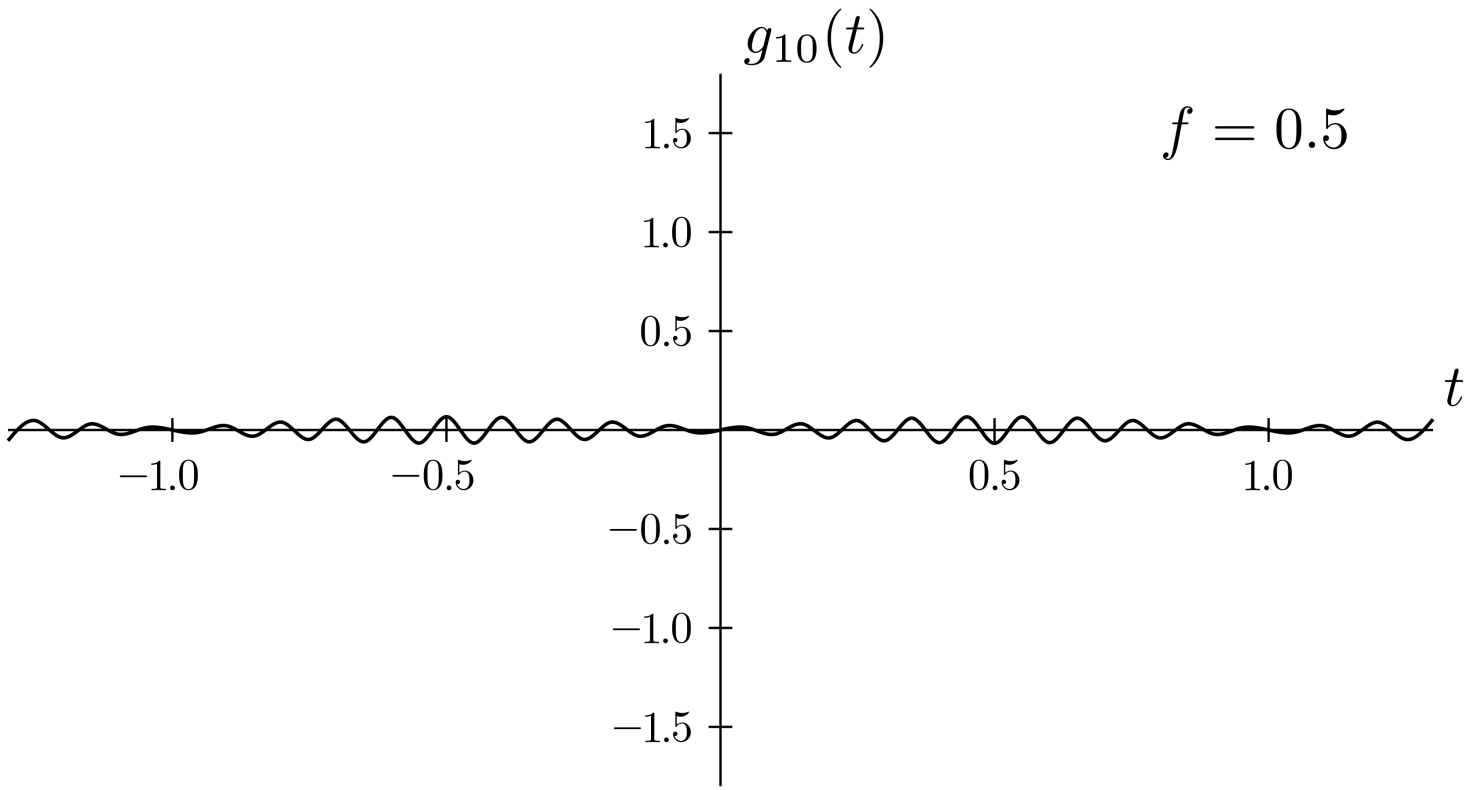
breaking down a scary looking function



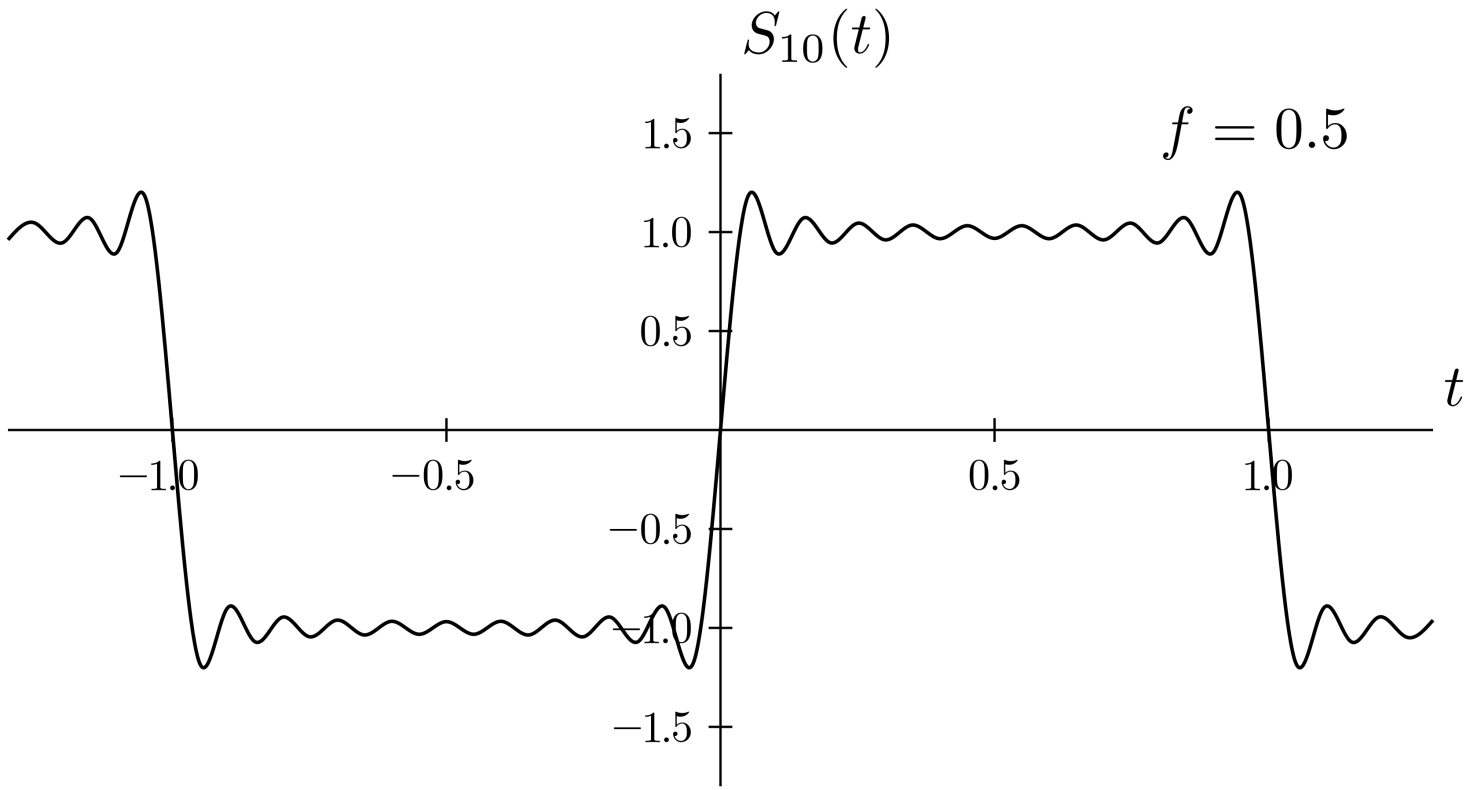
breaking down a scary looking function



breaking down a scary looking function



breaking down a scary looking function



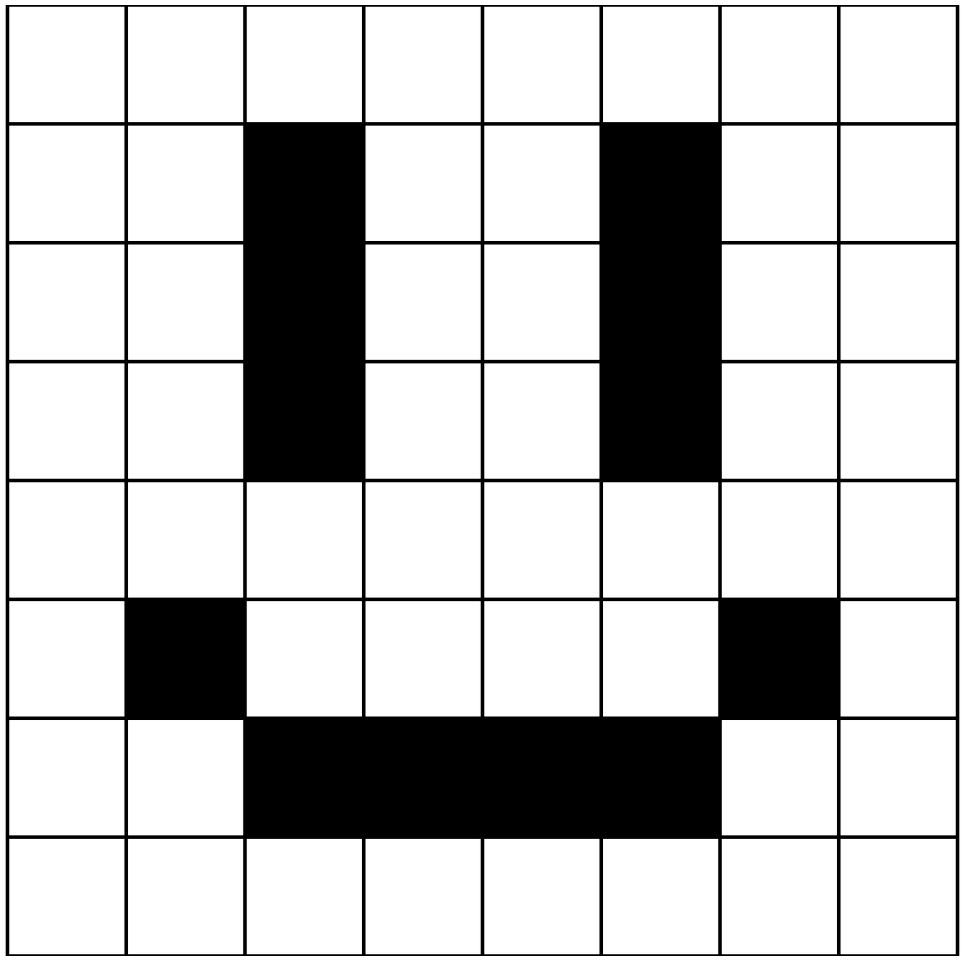
vectors (not the math kind), i.e. arrays

- set of pairs $\Rightarrow$ index and value
- elements (pairs) are conventionally of same memory size

vectors (not the math kind), i.e. arrays

```
one_d_array = [0, 0, 1, 0, 1, 0, 1, 1]
```

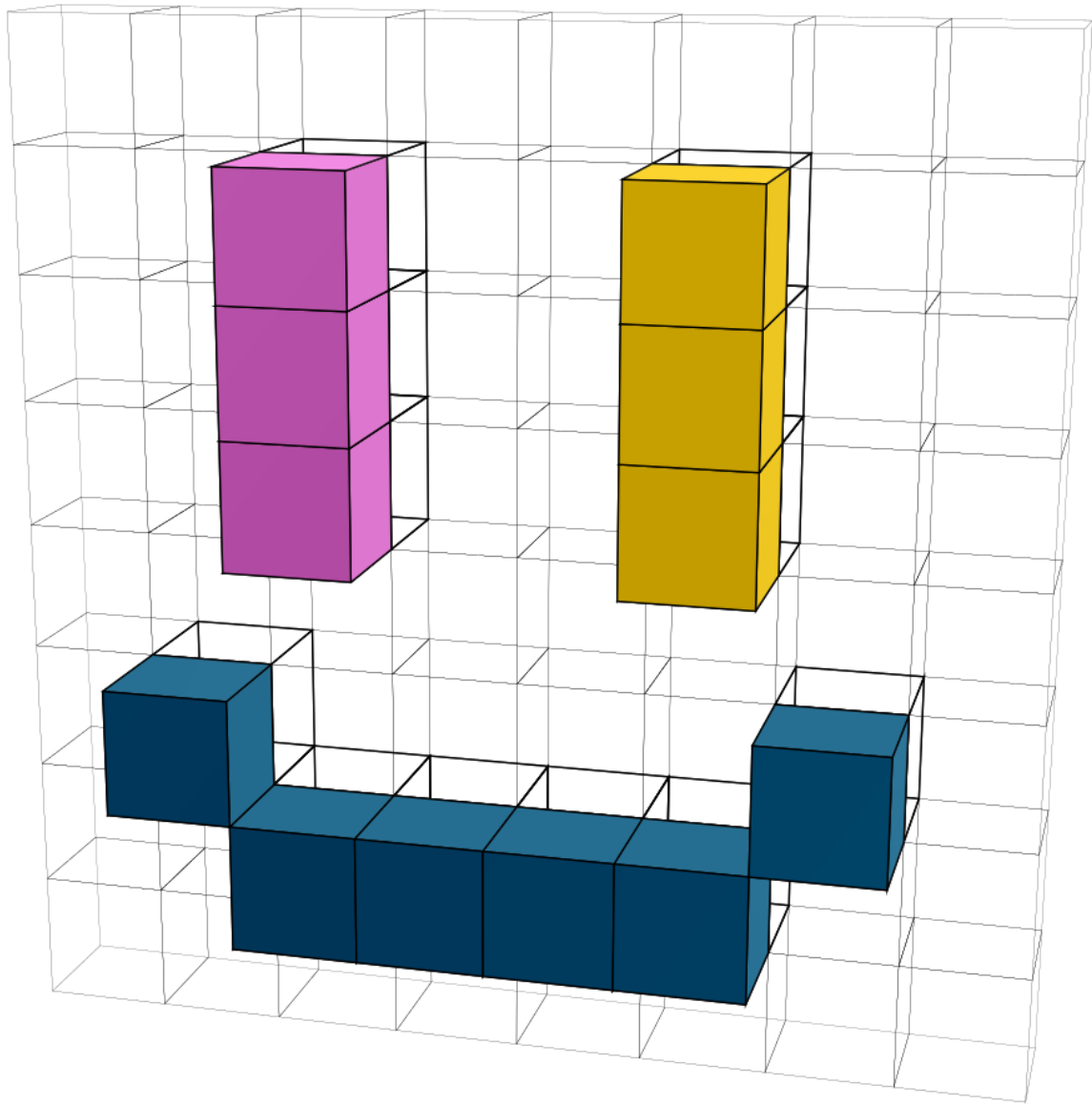
vectors (not the math kind), i.e. arrays



vectors (not the math kind), i.e. arrays

```
two_d_array = [
[0, 0, 0, 0, 0, 0, 0, 0],
[0, 0, 1, 0, 0, 1, 0, 0],
[0, 0, 1, 0, 0, 1, 0, 0],
[0, 0, 1, 0, 0, 1, 0, 0],
[0, 0, 0, 0, 0, 0, 0, 0],
[0, 1, 0, 0, 0, 0, 1, 0],
[0, 0, 1, 1, 1, 1, 0, 0],
[0, 0, 0, 0, 0, 0, 0, 0]
]
```

vectors (not the math kind), i.e. arrays



binary tree

data structure expressed as a figurative tree

one root node

nodes can only have one parent node and at most two children nodes (left and right)

foundation for more complex data structures