



International
JavaScript
Conference

by  devmio

Algorithm Design

for Web Developers

Michael Dowden *(he/him)*
@mrzdowden





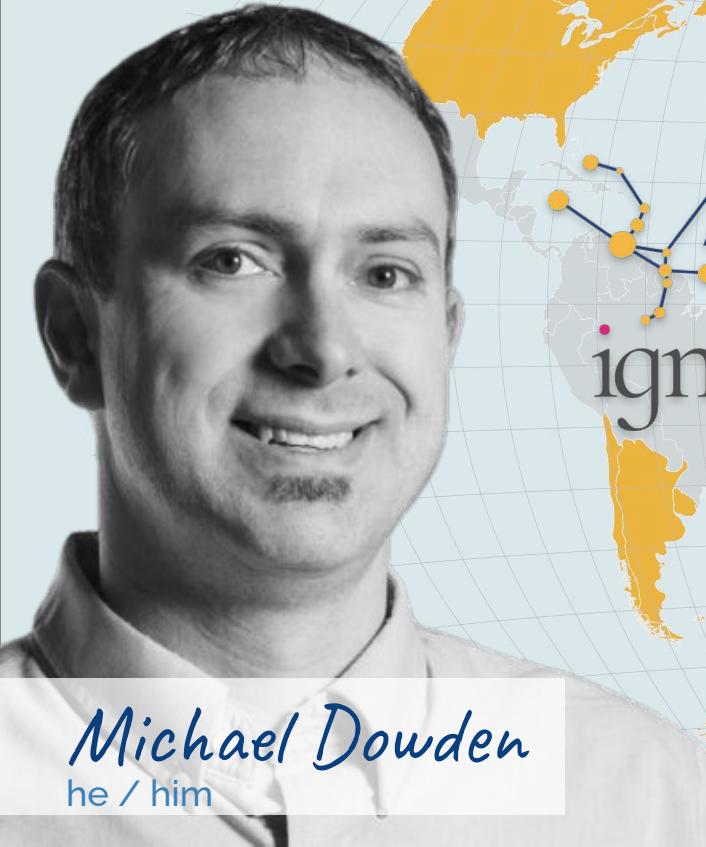
Michael Dowden

(he/him)

Presents

Algorithm Design

for Web Developers



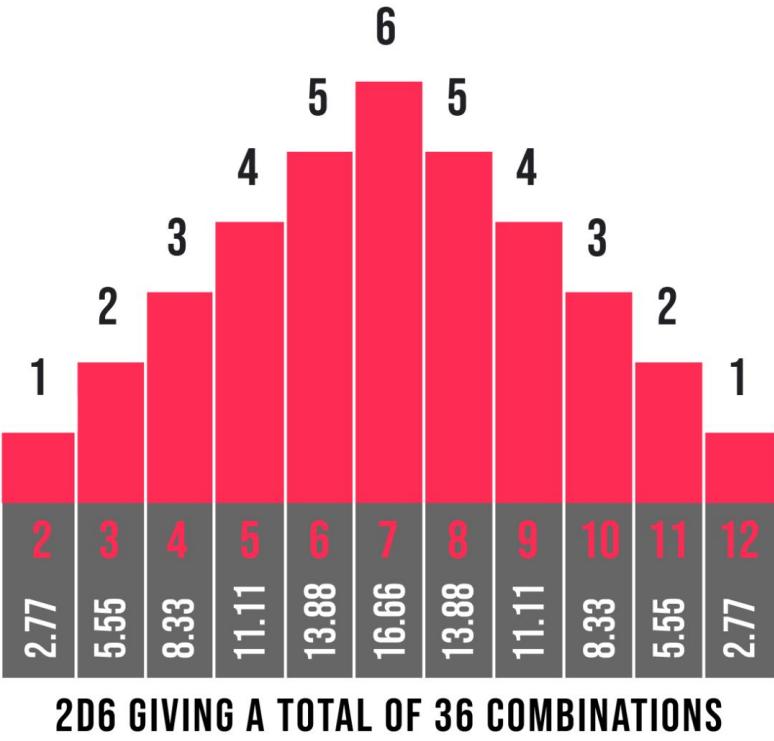
Michael Dowden
he / him







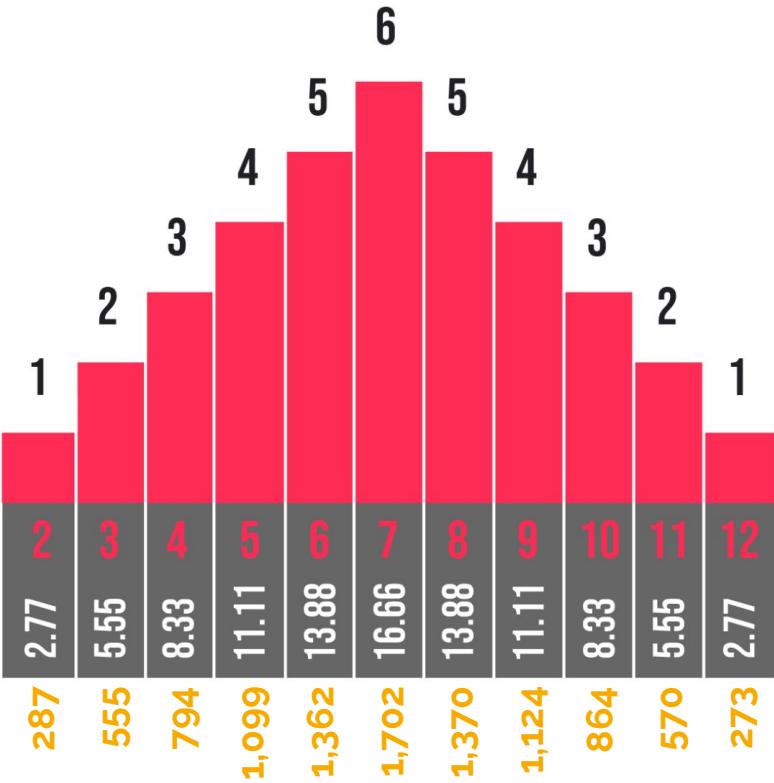




```
function randInt(min: number, max: number): number {  
    return Math.floor( Math.random() * (max - min + 1) + min)  
}
```

```
function roll2d6(): number {  
    return randInt(1, 6) + randInt(1, 6)  
}
```

```
function rollManyDice: number[] {  
    const qty = 10 * 1000  
    const results: number[] = []  
    for(let i = 0; i < qty; i++) {  
        results.push(roll2d6())  
    }  
}
```



```

type Counter = {
    value:number
    count:number
}
const stats: Counter[] = []
const qty = 10 * 1000

for(let d = 2; d <= 12; d++) {
    let count = 0
    for(let i = 0; i < qty; i++) {
        if(results[i] === d) count++
    }
    stats.push({ value: d, count })
}

console.table(stats)

```

Brute Force Method

(index)	value	count
0	2	287
1	3	555
2	4	794
3	5	1,099
4	6	1,362
5	7	1,702
6	8	1,370
7	9	1,124
8	10	864
9	11	570
10	12	273

```
// Array Count
const stats: Counter[]
= new Array(12+1)

for(let i = 2; i <= 12; i++) {
  stats[i] = { value:i, count:0 }
}

results.forEach(value => {
  stats[value].count++
})
```

```
// Reduce
const stats = results.reduce(
  acc: Counter[], n: number
) => {

  if(!acc[n]) {
    acc[n] = { value:n, count:1 }
  } else {
    acc[n].count++
  }

  return acc
}, [])
```

Comparing Algorithms

Algorithm	Operations	Elapsed Time	Memory Used
Brute Force	230,022	5.500ms	200 bytes
Array Count	10,022	1.400ms	184 bytes
Reduce	20,000	1.700ms	184 bytes

- Reduce requires no setup
- Reduce does not depend on mutability
- Reduce requires no knowledge of n (number of data points), or k (possible number of values)

Cyclomatic Complexity

Function Name	Start Line	End Line	Cyclomatic Complexity (Threshold: 10)	Lines of Code (Threshold: 50)	Parameter Count (Threshold: 4)
randInt	1	3	1	3	2
roll2d6	4	13	3	10	1
(anonymous)	53	56	1	4	1
(anonymous)	64	75	2	12	2
stats2d6	14	96	6	▲ 52	0
(anonymous)	107	118	2	12	2
(anonymous)	99	122	2	12	1
rollManyDice	97	123	1	4	0
sizeOf	127	154	▲ 11	28	1
formatByteSize	155	164	4	10	1
memorySizeOf	125	166	1	6	1

Source Code: f6248822-300e-11f0-8081-320933a50b16.js

Generated by [Codalyze](#) on 2025-05-13 15:28

Clear



Photo by [Rohan Makhecha](#) on [Unsplash](#)

Performant



Photo by [Kenny Leys](#) on [Unsplash](#)

Robust



Photo by [Christin Hume](#) on [Unsplash](#)



Gathering Metrics

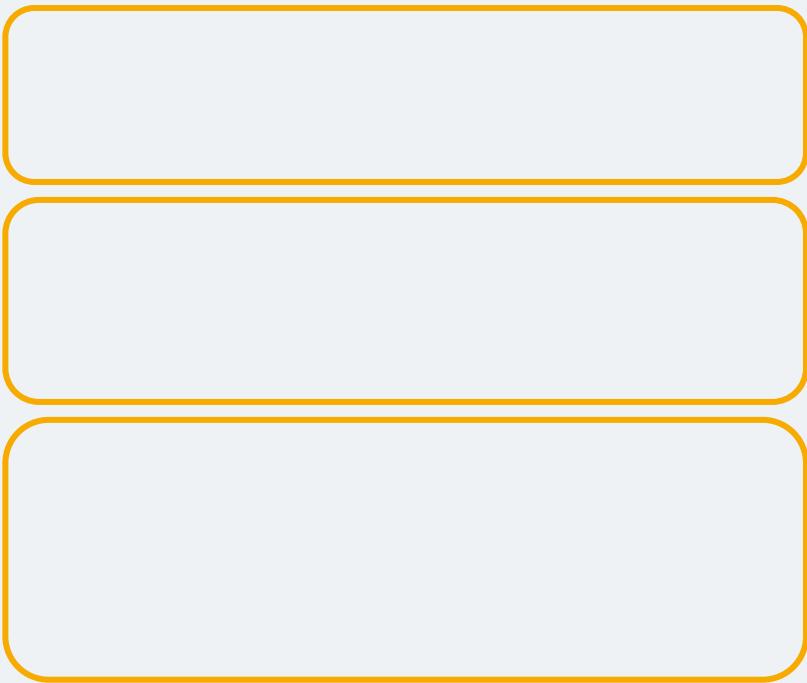
Calculating Memory

- Array - calculate based upon stored data
- Boolean - 4 bytes
- Number - 8 bytes (all are stored as doubles)
- Object - calculate based upon [own] properties
- String - 2 bytes/character (UTF-16)
- Other - convert to string and calculate accordingly
 - Includes BigInt, Symbol, etc.

<https://gist.github.com/rajinwonderland/36887887b8a8f12063f1d672e318e12e>



Measuring Time



Counting Operations

Brute Force Code

```
const qty = 10 * 1000  
  
for(let d = 2; d <= 12; d++) {  
    let count = 0  
    for(let i = 0; i < qty; i++) {  
        if(results[i] === d) {  
            count++  
        }  
    }  
    stats.push({ value: d, count })  
}
```

```
let t = 0 // Initialize Operation Counter
for(let d = 2; d <= 12; d++) {
    t++ // Initialize/increment d
    let count = 0
    t++ // Initialize count
    for(let i = 0; i < qty; i++) {
        t++ // Initialize/increment i
        if(results[i] === d) {
            count++
            t++ // Increment count
        }
        t++ // Compare d with current element
    }
    stats.push({ value: d, count })
}
```

The diagram illustrates the execution flow of the provided JavaScript code. It uses orange arrows pointing left to indicate the flow from right to left, which typically represents the direction of control flow in such visualizations. Annotations on the right side of the arrows provide specific values or counts for certain operations:

- An arrow from the first `t++` to the right is labeled **11**.
- An arrow from the second `t++` to the right is labeled **11**.
- An arrow from the third `t++` to the right is labeled **110,000**.
- An arrow from the fourth `t++` to the right is labeled **10,000**.
- An arrow from the fifth `t++` to the right is labeled **110,000**.



Asymptotic Notation

Asymptotic Notation

 Ω

Big Omega

 $\Omega(fn(n))$ Θ

Big Theta

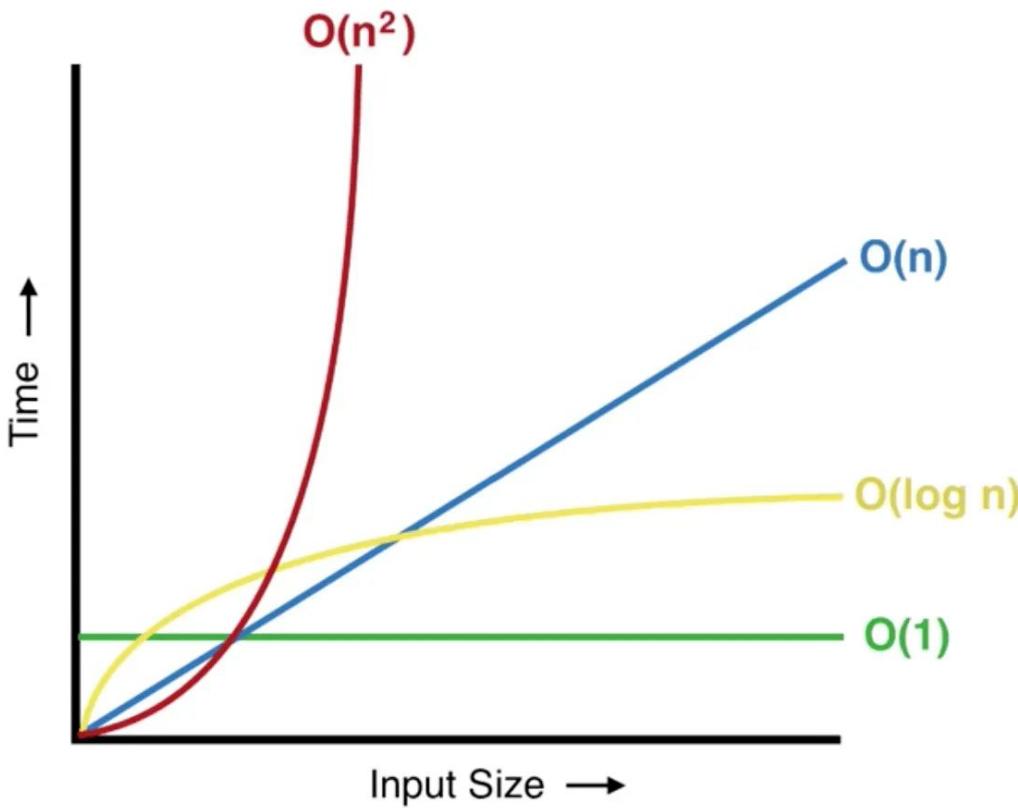
 $\Theta(fn(n))$ O

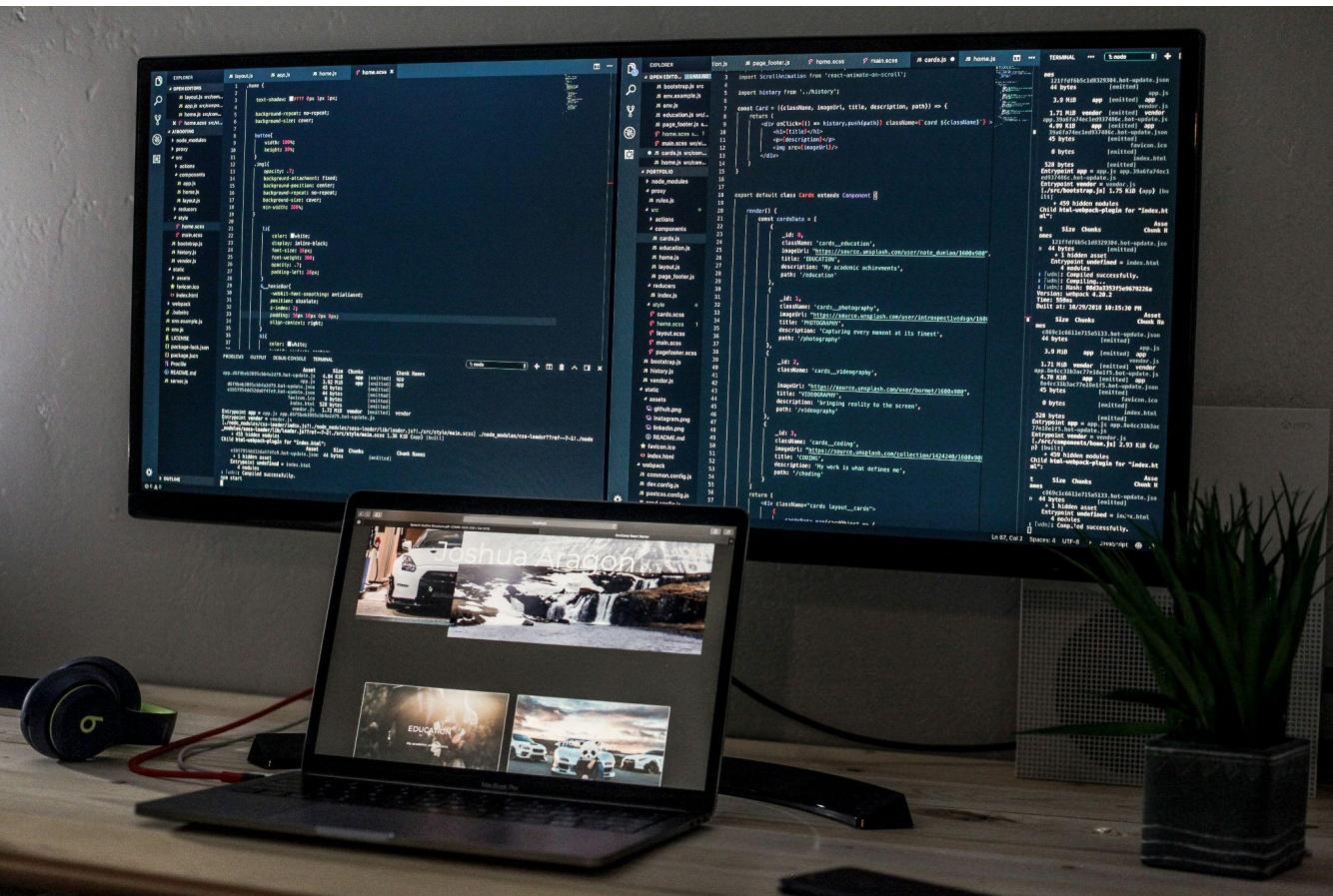
Big O

 $O(fn(n))$

Big O Notation

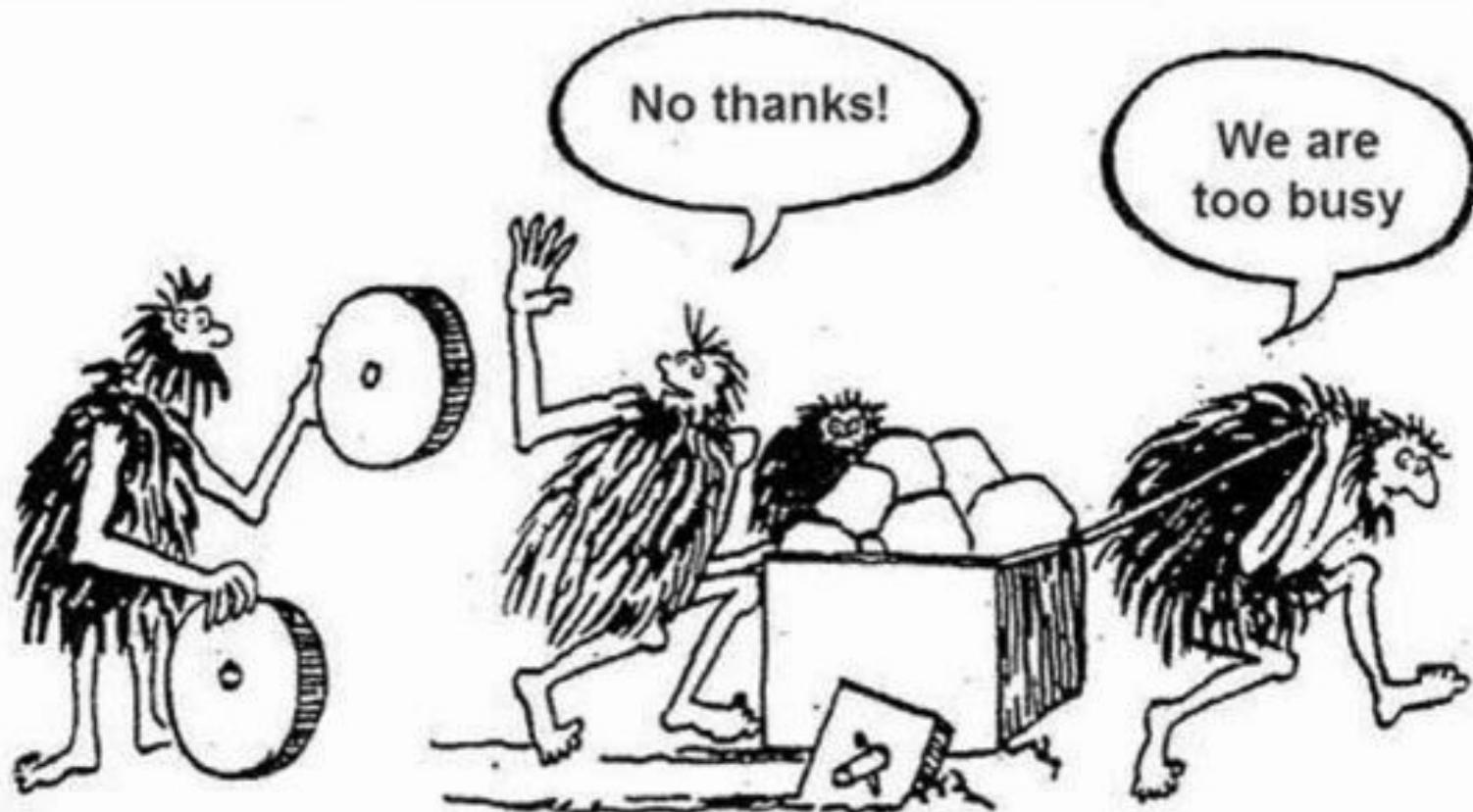
Diagram by [Maxwell Harvey Croy](#) on Medium





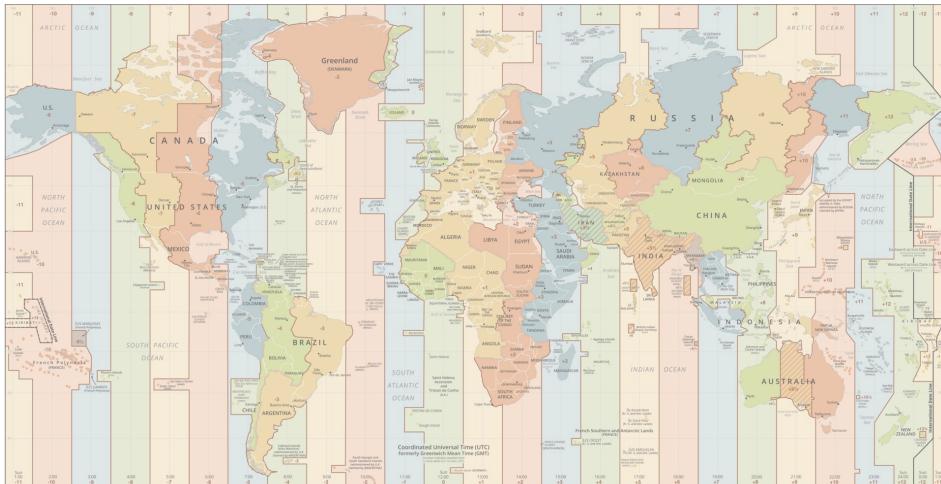


Not Invented Here



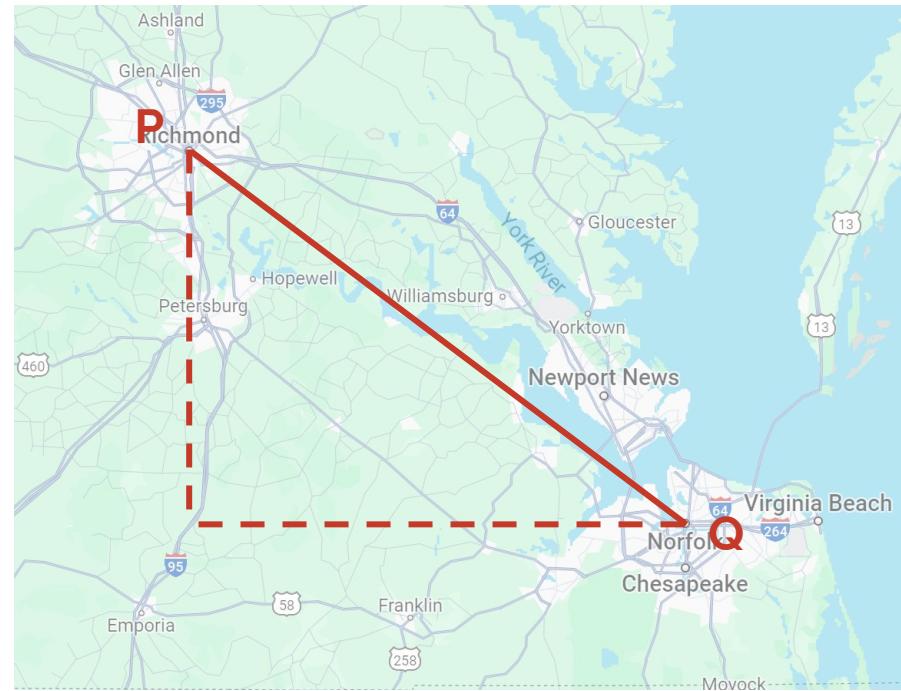
Time Zones

- Need today's date a year ago? Just subtract 365 days.
- *What about leap years?*
 - Add a day every year divisible by 4
- *What about non-leap centuries?*
 - Remove a day every 400 years
- *What about leap seconds?*
 -
- Just use [Luxon](#) (a DateTime Library)



Geographic Distance

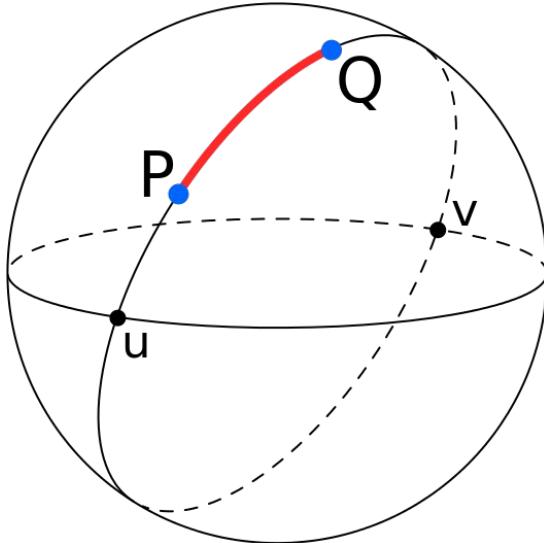
- Find the distance from P to Q
- λ = latitude, φ = longitude
- $P(\lambda_p, \varphi_p)$, $Q(\lambda_q, \varphi_q)$
- $a^2 + b^2 = c^2$
 - Pythagorean theorem
- $PQ = \sqrt{(\lambda_p - \lambda_q)^2 + (\varphi_p - \varphi_q)^2}$



Geographic Distance

Just use the Haversine formula

$$\text{haversin}\left(\frac{d}{r}\right) = \text{haversin}(\phi_2 - \phi_1) + \cos(\phi_1) \cos(\phi_2) \text{haversin}(\lambda_2 - \lambda_1)$$



Other Algorithms

- Sort
 - Insertion sort / Selection sort
 - Quick sort
 - Merge sort
- Search
 - Depth-first search (DFS)
 - Breadth-first search (BFS)
 - Binary search
 - Knuth–Morris–Pratt Algorithm (KMP)
- Data Compression
 - Huffman coding
 - DEFLATE



Web Considerations

API Considerations

- Are we hitting the API faster than we can get a reply?
- Are we causing accidental DDoS?
- Is our application tolerant of high latency?
- What is the volume of data involved in our application?

Photo by Annie Spratt on Unsplash

TURAL AND COMMON OBJECTS

Free-Arm and Industrial Drawing

cent circle into three equal parts from the second into the same number of parts, the alternating with those of the first circle; the three equal parts, the division lines similar with the second circle. The innermost circle in the Primaries, the second the Secondary. Scholars should learn how to construct these and white are not colours, the elements. White adds body, heightens its hue, black lessens it. Colours of prismatic colour are composed of all the primaries of colour; in which means an absence of any one of all the colours. In modern three-colour prints all effects are produced by colours superimposed, as in the printing of the three primaries of the other, generally in the Red, and Blue.

It would not, however, be very practicable to paint pictures with the three primaries, the process would be too tedious and intricate, so several intermediate pigments are admixed of varying proportions.

1

TECHNICAL POINTS

barbaric tribes in general. Lastly, by Japan in common with

The *Primaries* are :
The *Secondaries*
Orange, Green
The *Common Objects*
Crane, Burne-Jones, Morris, Helleu, Hassall, Rackham,
countries confirms this view of the importance of line. Re-
spective.
Hence we may console ourselves it cannot be all wrong
with the pencil when the form demands it and cannot be all wrong
produced first, it may be as far as the child is able to represent a
perfect and expressive as possible, shading or painting to give
work and technical drawing should therefore run along
and claywork. Outlines preparatory to shading or painting being
construction, &c.—Analysis of form and pair
great essentials; so are "construction lines" and co-
ellipses not always, but mostly.
with, radiation, convergence in perspective, etc.
be impressed unconsciously at every opportunity
natural forms. There may be
and as for mechanical aids the
they not also mechanical
so narrow of c

“massing” consists in building up objects can be built up from the starting-point, and the other shapes, and so on, it is quite

Free-Arm and Hand Exercises, as e.g.

Drawing "consists in building up forms, just as can be built up in clay. The "dot" is starting-point, and that is made to grow larger shapes, by enlargement and other processes. It is quite an evolutionary or accretive process. When the scholars grow older, they begin to draw a good outline, because they must needs make outline drawing and outline. They care

DOM Considerations

- Limited DOM size (elements on page)
 - Lighthouse warns at 800
 - Excessive at 1,400
 - Effectively stops working on most machines at 3,000
- Rendering & Painting
 - Cause broken-looking UI / flashing
 - Sluggish page performance



In Conclusion

Vocabulary

- Atomic
- Idempotent
- Orthogonal
- Composable
- Side Effects
- Mutability / Immutable
- Deterministic
- Pure Function

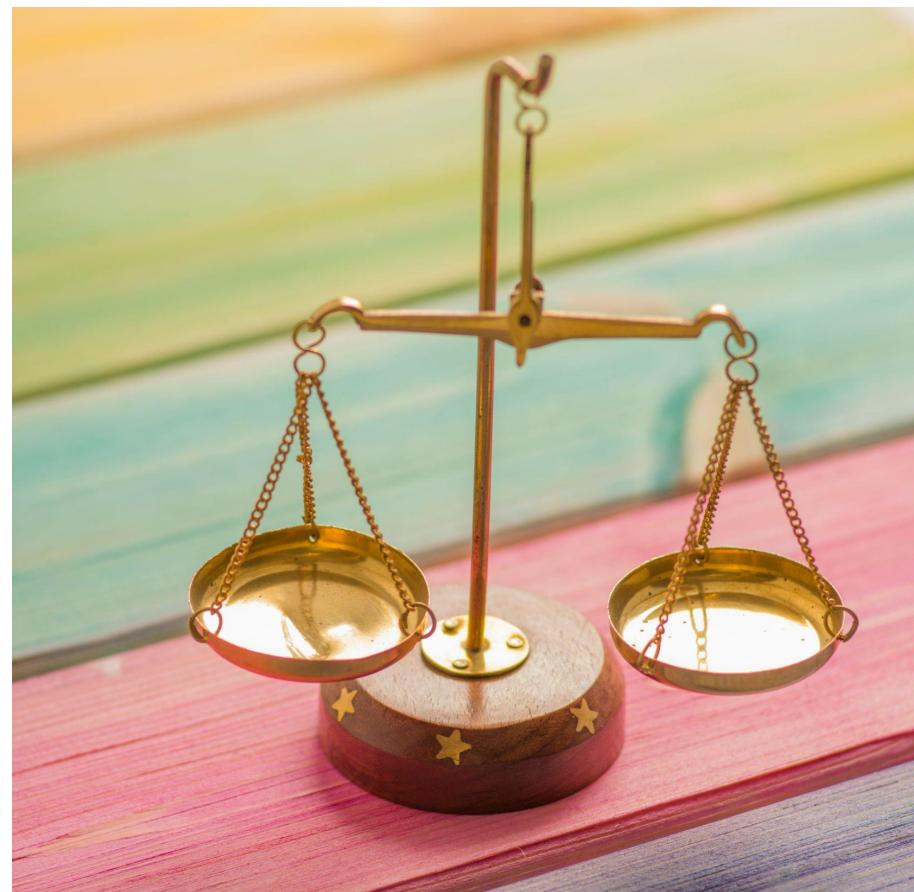
Clear



Performant



Robust



We ask for
your feedback!

PLEASE
VOTE
NOW!



Michael Dowden *(he/him)*

@mr dowden



mrdowden.com



Architecting CSS

The Programmer's Guide to Effective
Stylesheets

—
Martine Dowden
Michael Dowden

Apress

Approachable Accessibility

Planning for Success
—
Martine Dowden
Michael Dowden

Apress

Programming Languages

ABC++



Resources

- <https://github.com/M2D2co/algorithm-design>
- <https://gist.github.com/rajinwonderland/36887887b8a8f12063f1d672e318e12e>
- https://developer.mozilla.org/en-US/docs/Web/API/Performance_API
- <https://developer.mozilla.org/en-US/docs/Web/API/Performance>
- <https://dev.to/shevchenkonik/memory-size-of-javascript-boolean-3mlj>
- <https://www.khanacademy.org/computing/computer-science/algorithms/asymptotic-notation/a/asymptotic-notation>
- https://en.wikipedia.org/wiki/Big_O_notation
- <https://www.igismap.com/haversine-formula-calculate-geographic-distance-earth/>
- <https://web.dev/articles/dom-size-and-interactivity>

Media

- Photo by [Joshua Hoehne](#) on [Unsplash](#)
- Photo by [Galen Crout](#) on [Unsplash](#)
- Photo by [Brett Jordan](#) on [Unsplash](#)
- Diagram from [The Dark Fortress](#)
- Photo by [Rohan Makhecha](#) on [Unsplash](#)
- Photo by [Kenny Leys](#) on [Unsplash](#)
- Photo by [Christin Hume](#) on [Unsplash](#)
- Diagram by [Maxwell Harvey Croy](#) on [Medium](#)
- Photo by [Elena Mozhvilo](#) on [Unsplash](#)
- Photo by [Annie Spratt](#) on [Unsplash](#)
- Photo by [Pawel Janiak](#) on [Unsplash](#)
- Photo by [Joshua Aragon](#) on [Unsplash](#)