

# Interviewing for Frontend Engineers

Jem Young

@jemyoung

Senior Software Engineer





**FRONT END  
HAPPY HOUR**

Why is interviewing so  
difficult?

**Programming is math**

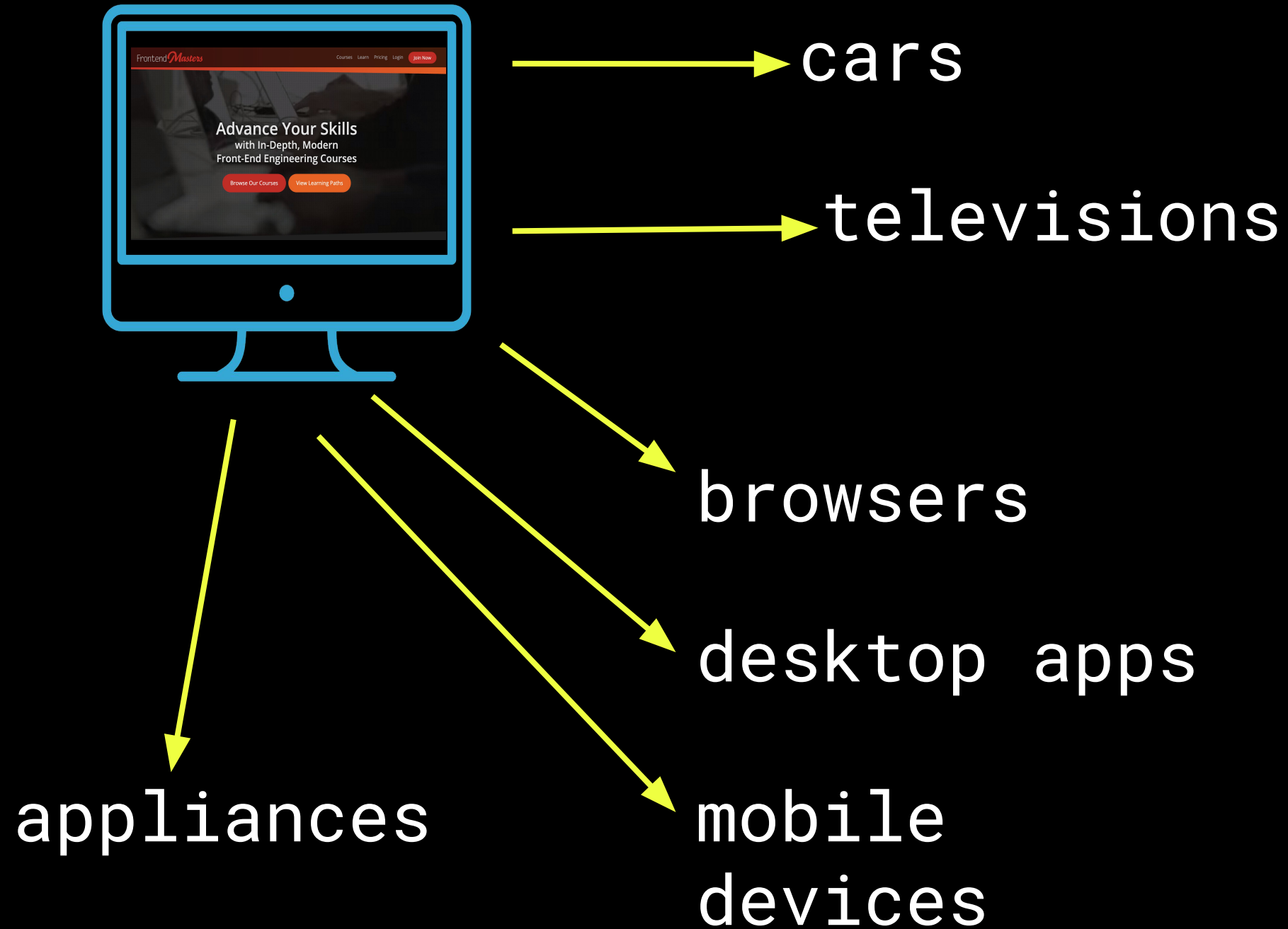
**Programming is for  
controlling hardware**

**Programming is for  
building things**

**3 types of  
programmers**

Why is interviewing for  
Frontend so difficult?

# Frontend







**Jem Young**

@JemYoung



In JavaScript, how would `_you_` define scope?

1:20 PM - 30 Oct 2019

**3** Retweets **39** Likes



34



3



39



# Interviewing for Frontend Engineers?

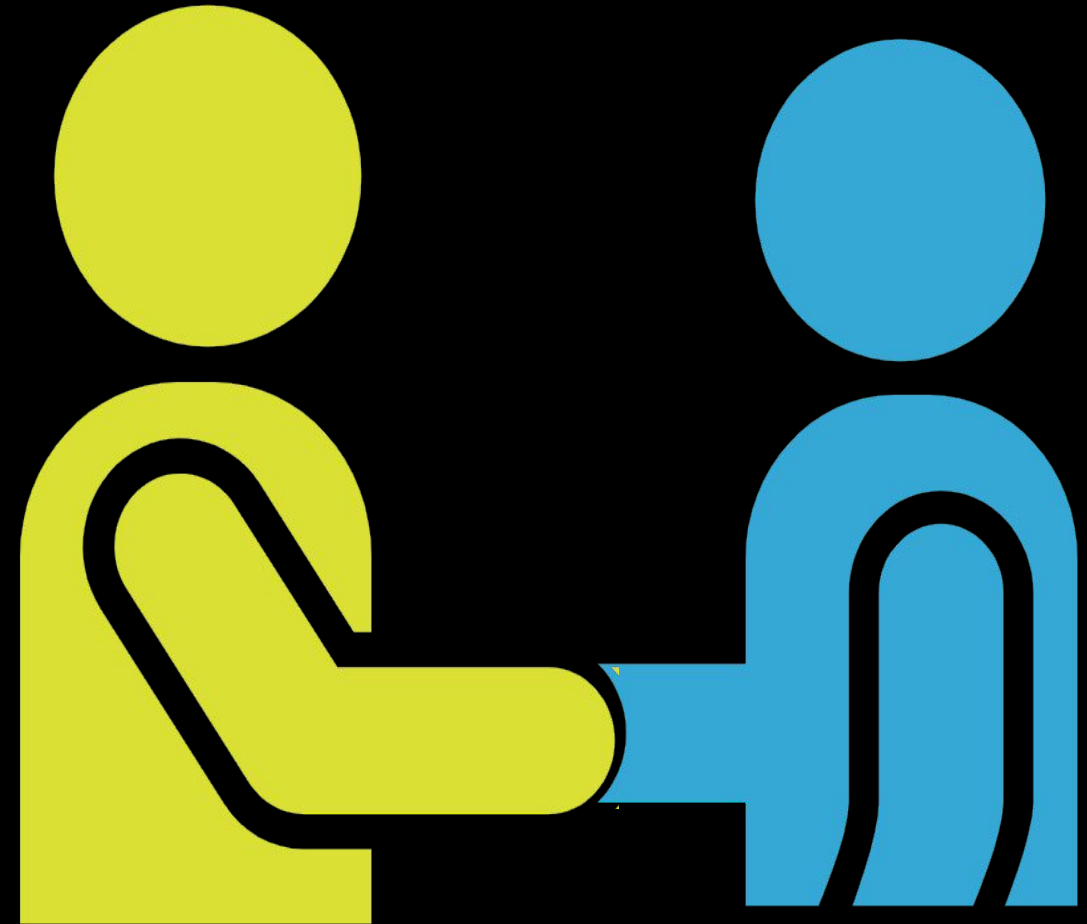
**candidate**

**interviewer**

0.300

0.501

- ❏ Application
- ❏ Initial call
- ❏ Code test
- ❏ Phone screen
- ❏ On-site
- ❏ Result



# Application



What do you like about your current job?

What sort of things are you looking for  
in your next role?

Which companies have appealing engineering cultures? What makes it appealing?

Application

preparation

- One page
- Skills
- Experience
- Technology familiarity
- Education
- Accomplishments

<div>Jem Young</div> <div><div>Address</div><div>T: ###-###-####</div><div>E: email@###.com</div><div>W: <a href="#">github.com/young</a></div></div> <div>JY</div>	
Professional Experience	<div><div>TITLE, COMPANY</div><div>LOCATION — DATE STARTED - DATE ENDED</div><div><div><ul style="list-style-type: none"><li>What you worked on, responsibilities, description of role.</li></ul></div><div>Technology/software used</div></div><div>Notable Projects/Skills/Accomplishments</div><div><div>Senior Software Engineer, Netflix</div><div>Los Gatos, California — January 2016 - present</div><div><ul style="list-style-type: none"><li>Led feature development and creation of A/B tests (<b>React/LESS/NodeJS</b>) for the Netflix non-member experience.</li><li>Leader of several working groups and infrastructure projects ranging from website performance to testing ergonomics. Member of UI Core infrastructure team.</li><li>Member of TC39</li></ul></div></div><div><div>Notable Projects</div><div><ul style="list-style-type: none"><li><i>Atticus</i>: A stateless caching proxy (<b>NodeJS/Redis</b>). Increased automated end-to-end test stability from 88% to 99%.</li><li><i>Our Story</i>: A templating framework enabling rapid creation and iteration of server driven A/B tests.</li></ul></div></div></div>
Education	<div><div>SCHOOL</div><div>DEGREE — DATE</div></div>

What do I do if I have little/no experience?

preparation

Application



- Which roles does this person fit?
- What sort of experience is required?
- Is there any bias in your selection process?



☒ Application

☐ Initial call

☐ Code test

☐ Phone screen

☐ On-site

☐ Result

**Initial call**

**preparation**

1. What do you do currently?
2. What are some projects you've worked on recently?
3. What are you looking for in your next role?
4. Why do you want to work for \_\_\_\_\_?
5. What is your availability for the next steps?

**Initial call**

**preparation**

1. How many steps are in the interview process? How long does it generally take?
2. How big is your engineering team?
3. Which team would I be interviewing for?
4. What is the culture like?
5. Who are you competitors?
6. What sort of projects would I work on?

- What is the difference between const, let, and var?
- Explain prototypical inheritance
- What is 'this' mean in JavaScript?
- What is the data structure of the DOM?
- What is a Stack and a Queue? How would you create those data structures in JavaScript?
- How can you tell if an image element is loaded on a page?
- What is call() and apply()?
- What is event delegation and what are the performance tradeoffs?
- What is a Worker? When would you use one?

# Prescreen



What is the difference  
between const, let, and var?

Explain prototypical  
inheritance

What does 'this' mean in  
JavaScript?

What is the data structure  
of the DOM?

What is a Stack? What is a Queue? How would you create those data structures in JavaScript?

How can you tell if an  
image element is loaded on  
a page?

What are `call()` and `apply()`?



What is event delegation?

What is a Worker? When  
would you use one?

TRIVIA



☒ Application

☒ Initial call

☐ Code test

☐ Phone screen

☐ On-site

☐ Result

Code test

preparation

- Make your code as readable as possible
  - Comment your code
  - Don't over complicate the architecture
- Don't import too many libraries
- If you have time, add unit tests
- Ask questions!

## Code test tips

Code test

preparation

- Make the problem as straightforward as possible
- Be honest with the time constraints
- Have a code review checklist

# Code test tips



Average	Good	Exceptional
Application starts properly.  3/5 requirements complete.	Code is well documented.  All requirements are complete.  No errors are thrown in the console.	Modular architecture designed for extensibility.  Created unit and integration tests.

Knowledge: Big-0

```
function search(arr, num) {  
  for (let i = 0; i < arr.length; i++) {  
    if (arr[i] === num) {  
      return true;  
    }  
  }  
  
  return false;  
}
```

Big Omega  $\Omega$  “*best case*”

Big Theta  $\Theta$  “*average case*”

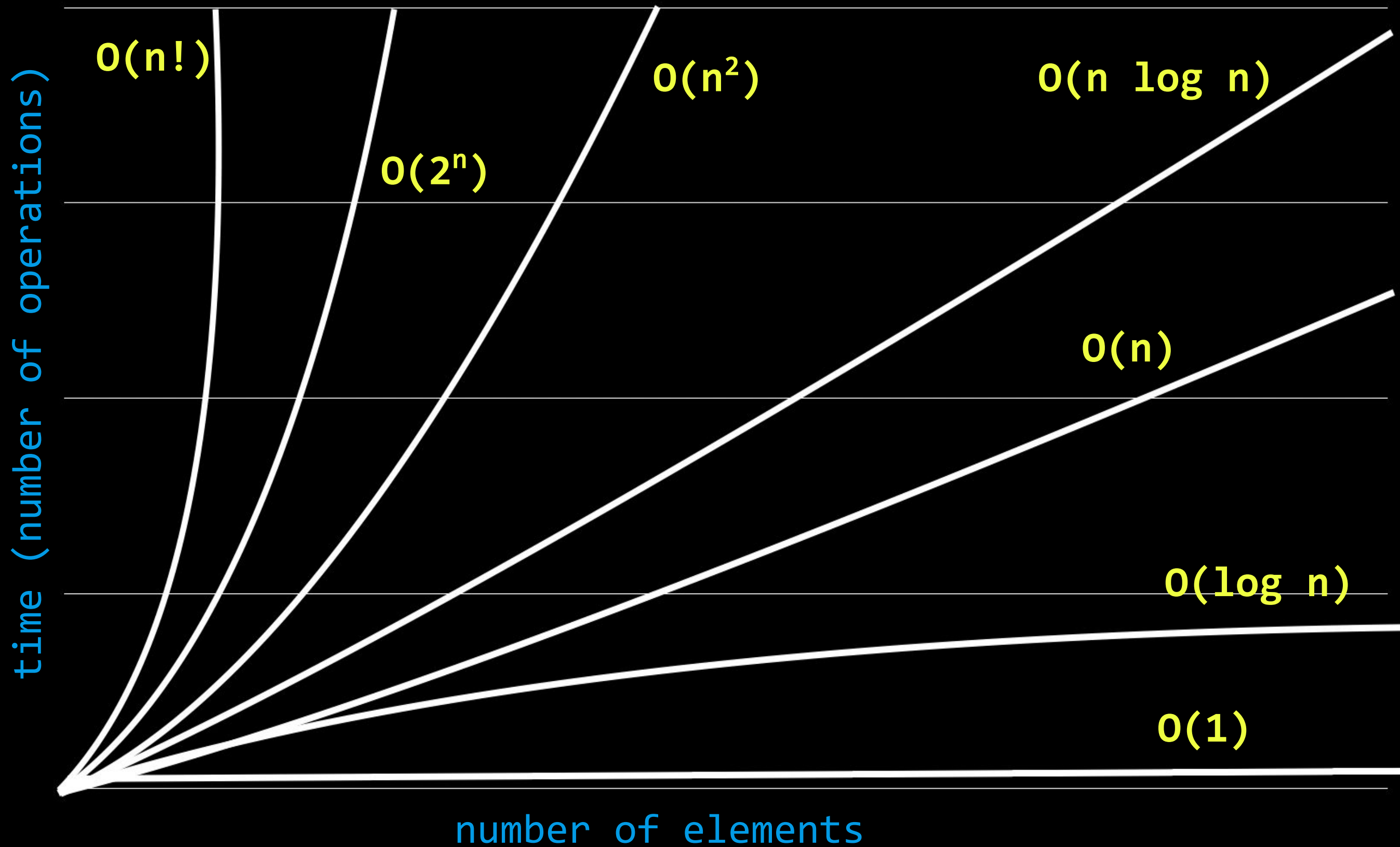
Big O  $O$  “*worst case*”

$\Omega(1)$

$\Theta(n)$

$O(n)$

```
function search(arr, num) {  
  for (let i = 0; i < arr.length; i++) {  
    if (arr[i] === num) {  
      return true;  
    }  
  }  
  
  return false;  
}
```



$\Omega(1)$

$\Theta(n)$

$O(n)$

```
function search(arr, num) {  
  for (let i = 0; i < arr.length; i++) {  
    if (arr[i] === num) {  
      return true;  
    }  
  }  
  
  return false;  
}
```

```
for (let i = 0; i < arr.length; i++) {  
  for (let x = 0; x < arr2.length; x++) {  
    for (let y = 0; y < arr3.length; y++) {  
      }  
    }  
  }  
}
```



```
arr.map(item => {  
  arr.forEach(i => {  
    arr.filter( x => {  
      //  
    })  
  })  
})
```

☒ Application

☒ Initial call

☒ Code test

☐ Phone screen

☐ On-site

☐ Result

Phone screen

preparation

- Ask questions
- Talk out your solution
- Get comfortable with the environment

**preparation**

**Phone screen**

- Are you in a quiet area?
- Is the problem well worded?
- Does the candidate know what the requirements and restrictions are?
- Did you leave time for questions at the end?

Problem: phone screen

☒ Application

☒ Initial call

☒ Code test

☒ Phone screen

☐ On-site

☐ Result



On-site

preparation

- Practice writing code without a computer
- Go over general sample problems
- Ask your friends to test you
- Try to ask what the style of technical questions will be

On-site

preparation

- Make sure questions are relevant to the role
- Have backup interviewers
- Book the room for the day
- Allow time for breaks (bathroom, lunch, etc)

**On-site**

Knowledge: strings

`.split()`

`.toLowerCase()`

`.substring()`

`.startsWith()`

**Useful  
methods**

Problem: reverse



Knowledge: arrays

`Object.entries()`

`Array.from()`

`[...item]`

`.isArray()`   `.pop()`

`.filter()`   `.push()`

`.reduce()`   `.map()`

`.concat()`

`.join()`

**Useful  
methods**

Problem: duplicate strings

Problem: flatten array

Knowledge: scope

`.call()`  
`.apply()`

`.bind()`

Problem: bind



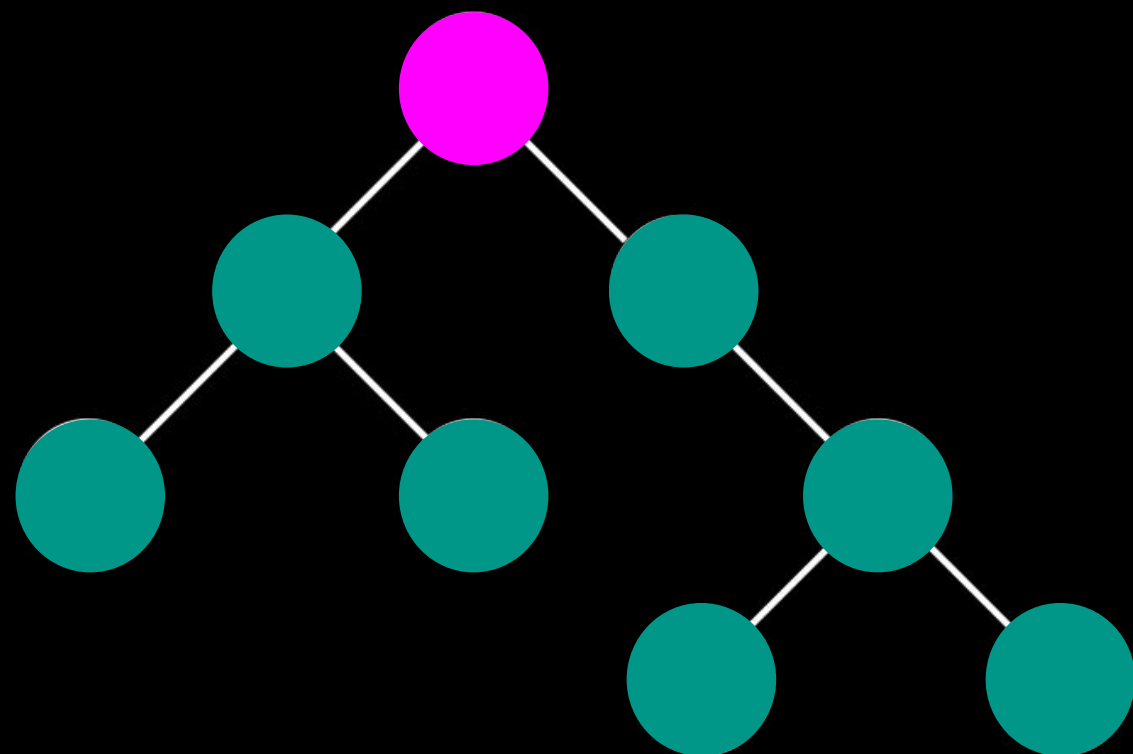
Knowledge: timing

setInterval()

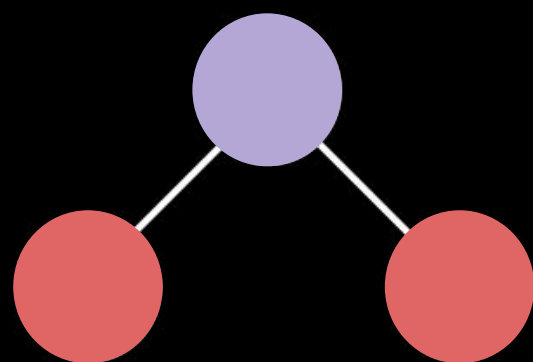
setTimeout()

Problem: debounce

Knowledge: trees



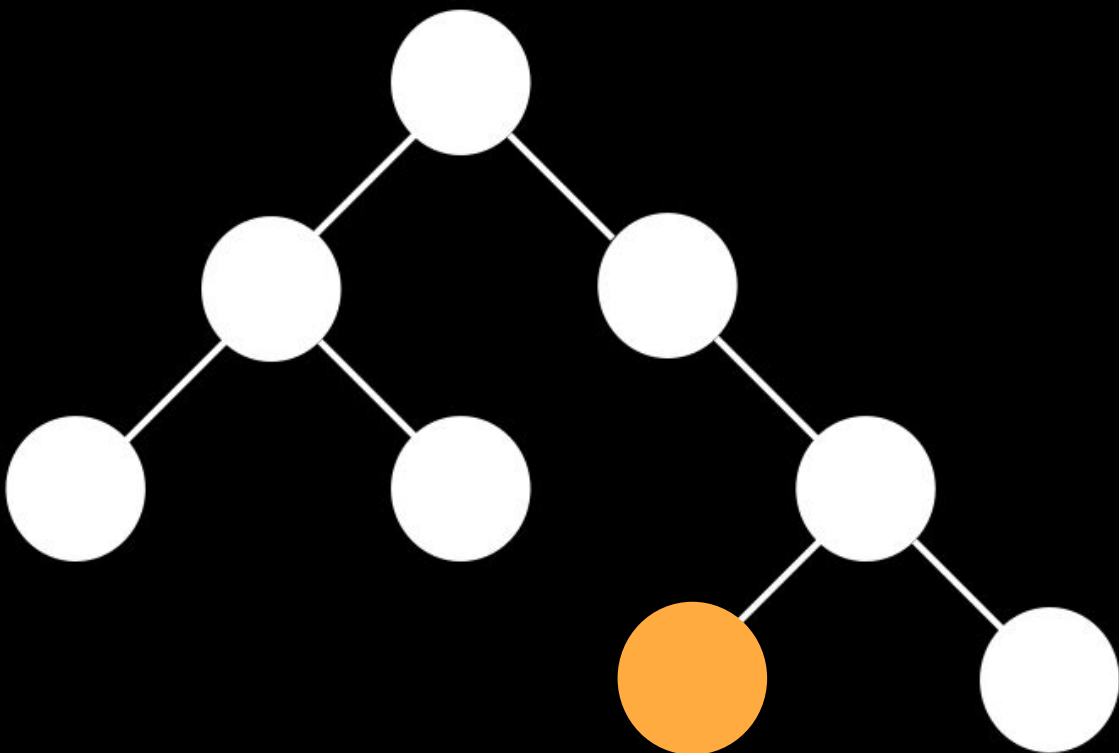
- Root
- Nodes



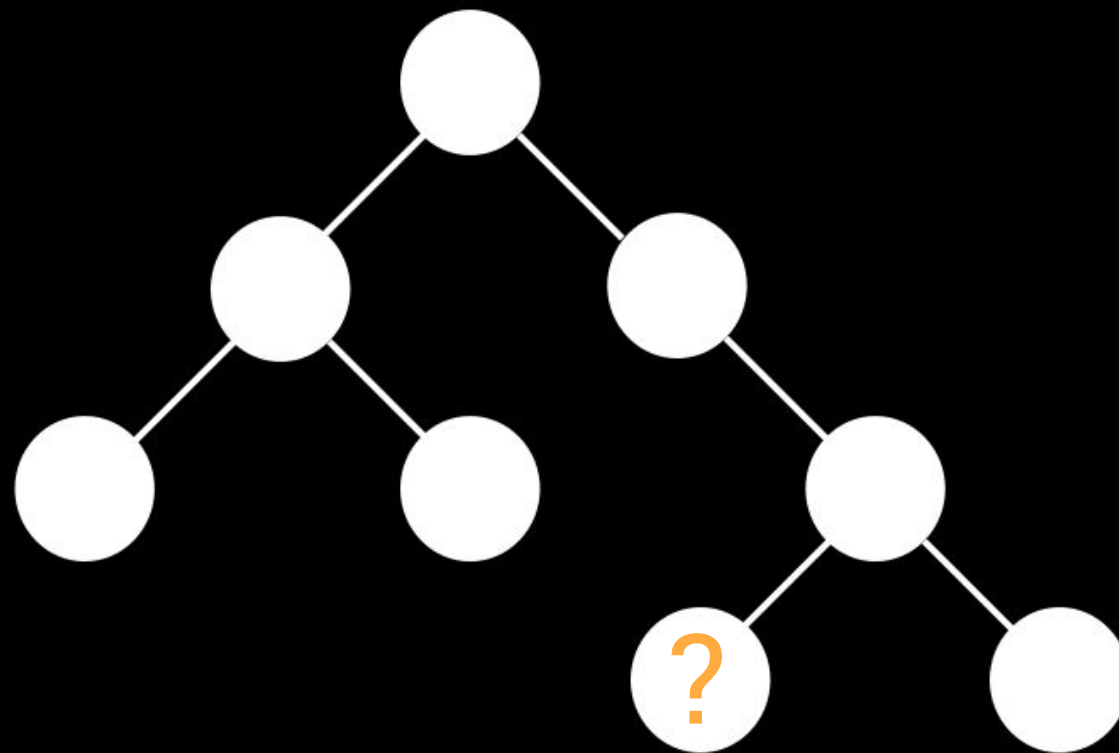
[ 0 , 1 ]

- Children
- Parent

A



B



Problem: domTree

Knowledge: rendering



requestAnimationFrame()

Problem: move element

Knowledge: promises

```
function returnApple(callback) {  
  |   setTimeout(() => {  
  |     |   callback('🍏')  
  |   }, 500);  
}
```

```
function returnApplePromise() {  
  return new Promise(function(resolve, reject) {  
    setTimeout(() => {  
      resolve('🍏');  
    }, 500);  
  });  
}
```

```
async function getApples() {  
  const apple = await returnApplePromise();  
  console.log(apple);  
  
  returnApple(apple2 => {  
    console.log(apple2)  
  });  
}
```

Problem: sleep

Problem: promisify



☒ Application

☒ Initial call

☒ Code test

☒ Phone screen

☒ On-site

☐ Result

**Result**

**preparation**

0.501

Result

preparation

- Give feedback
- Invite them to re-apply
- Stay in touch

**Thanks!**