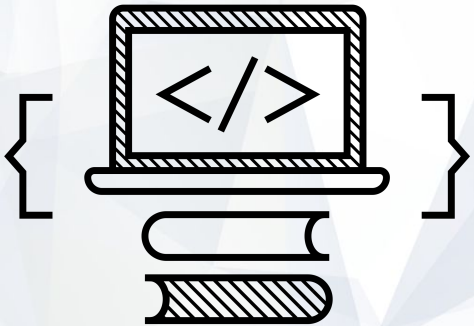
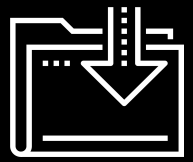


Computer Science for JavaScript

Coding Boot Camp

Module 17



What is computer science?

What Is Computer Science?

01

Computer science is formally known as “the study of computers and computational systems.”

02

The study of the design, development, and analysis of software systems.

03

The art of telling a computer what to do through a set of instructions.

How will understanding computer science make us better programmers?

The Benefits of Understanding Computer Science

These concepts often appear during technical interviews.

When inheriting large codebases, you might be asked to optimize code efficiency.

Understanding what makes up a program will help us design software more efficiently.

Discovering how a program executes can tell us how to avoid performance issues.

Computational thinking refers to the formulation of a solution using logic, patterns, and problem-solving methods.

Which computer science concepts will we learn?

Computer Science Concepts

Algorithms

We will learn about algorithms and how to gauge their performance.

Data Structures

We will learn about data structures and how they are implemented in JavaScript.

What is Big O notation?

Big O Notation

Big `O` notation helps us describe how runtime will scale when we increase input size `(n)`.

It is denoted with a capital `O` alongside the growth factor (in parentheses).

Example: The time complexity of simple search is `O(n)`

What is a data structure?

What is a Data Structure?

01

A way of storing data so that it can be used efficiently by the computer or browser.

02

It is built upon simpler primitive data types (like variables).

03

It is non-opinionated, in the sense that it is only responsible for holding the data.

Example Data Structure: Arrays

```
var favFoods = ["Pickles", "Onions", "Carrots"]
```

Why is computer science a common category for technical interview questions?

Computer Science in Technical Interviews

01

Understanding computer science fundamentals shows that you know how a program functions and how to make it faster and more efficient.

02

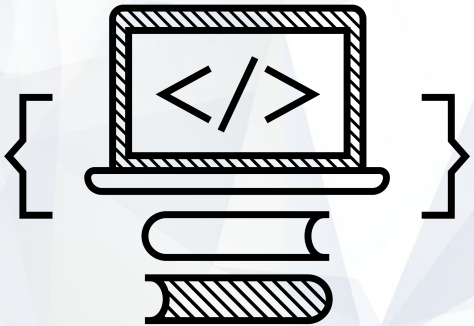
Many candidates are computer science graduates.

03

Comprehending the use cases of data structures and the algorithms that manipulate them will help you create more performant applications.

Instructor Demonstration

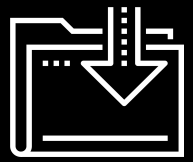
Mini-Project



Computer Science for JavaScript

Coding Boot Camp

Module 17



What is computer science?

What Is Computer Science?

01

Computer science is formally known as “the study of computers and computational systems.”

02

The study of the design, development, and analysis of software systems.

03

The art of telling a computer what to do through a set of instructions.

How will understanding computer science make us better programmers?

The Benefits of Understanding Computer Science

These concepts often appear during technical interviews.

When inheriting large codebases, you might be asked to optimize code efficiency.

Understanding what makes up a program will help us design software more efficiently.

Discovering how a program executes can tell us how to avoid performance issues.

Computational thinking refers to the formulation of a solution using logic, patterns, and problem-solving methods.

Which computer science concepts will we learn?

Computer Science Concepts

Algorithms

We will learn about algorithms and how to gauge their performance.

Data Structures

We will learn about data structures and how they are implemented in JavaScript.

What is Big O notation?

Big O Notation

Big `O` notation helps us describe how runtime will scale when we increase input size `(n)`.

It is denoted with a capital `O` alongside the growth factor (in parentheses).

Example: The time complexity of simple search is `O(n)`

Big O Notation

Which function is more efficient?

Which has fewer instructions?

```
function list_items (list) {
  for (var i = 0; i < list.length; i += 1) {
    // Log each item in the array
    console.log(list[i]);
  }
}

function head (list) {
  // Return first item of a list
  return list[0];
}
```

Count the instructions!

Count Instructions

`head` = 1 instruction

Count Instructions

`list_items = n` instructions

…

`(n = list.length)`

The Verdict

`head` is more efficient.

The Verdict

But `list_items` isn’t bad.

Quantifying Efficiency

head **always** executes one instruction…

Quantifying Efficiency

…no matter how long the array is.

Quantifying Efficiency

head takes the same amount of time on any input

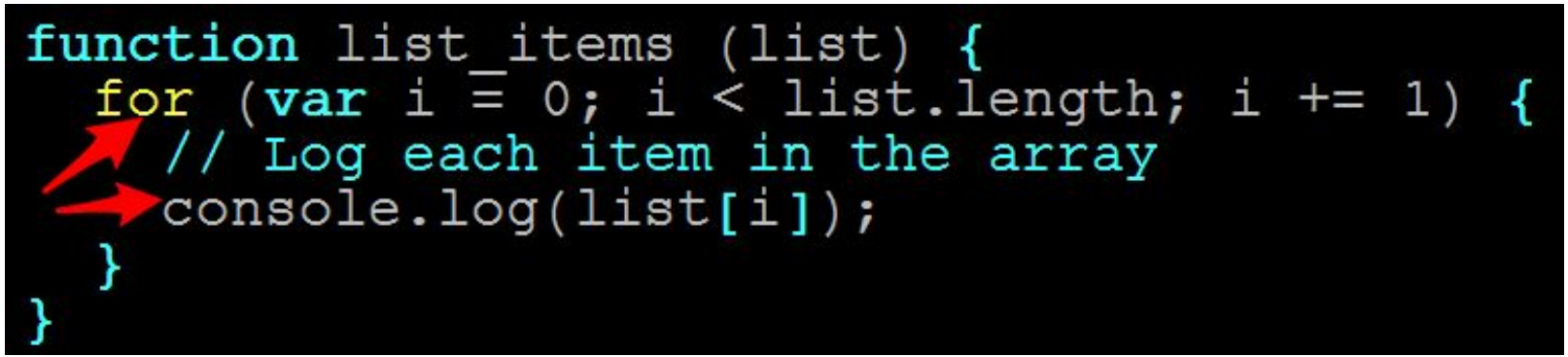
```
// Three elements...
var names = ['Gogol', 'Pushkin', 'Dostoevsky'];

// One thousand elements...
var huge_array = generate_array(1000);

// ...But these statements take
//    the same amount of time.
console.log( head(names) );
console.log( head(huge_array) );
```

Quantifying Efficiency

`list_items` needs `n` instructions

Quantifying Efficiency

One `console.log` per item:



```
function list_items (list) {
  for (var i = 0; i < list.length; i += 1) {
    // Log each item in the array
    console.log(list[i]);
  }
}
```

Quantifying Efficiency

`console.log` is fast…

Quantifying Efficiency

…but not free.

Quantifying Efficiency

Longer arrays = more time

Quantifying Efficiency

Double array length = Double time
Triple array length = Triple time

Quantifying Efficiency

In other words…

Quantifying Efficiency

The running time of `head` and `list_items` scale differently.

Time complexity =
the rate at which algorithm **slows**
as input **grows**.

Quantifying Efficiency

`head` is **always** one instruction.

Running time **does not** slow for larger inputs.

Quantifying Efficiency

In other words…

Quantifying Efficiency

The running time of `head` is **constant**.

Quantifying Efficiency

`list_items` takes `n` instructions

Running time **depends on array**.

Quantifying Efficiency

Double array length,
double time,
etc…

Quantifying Efficiency

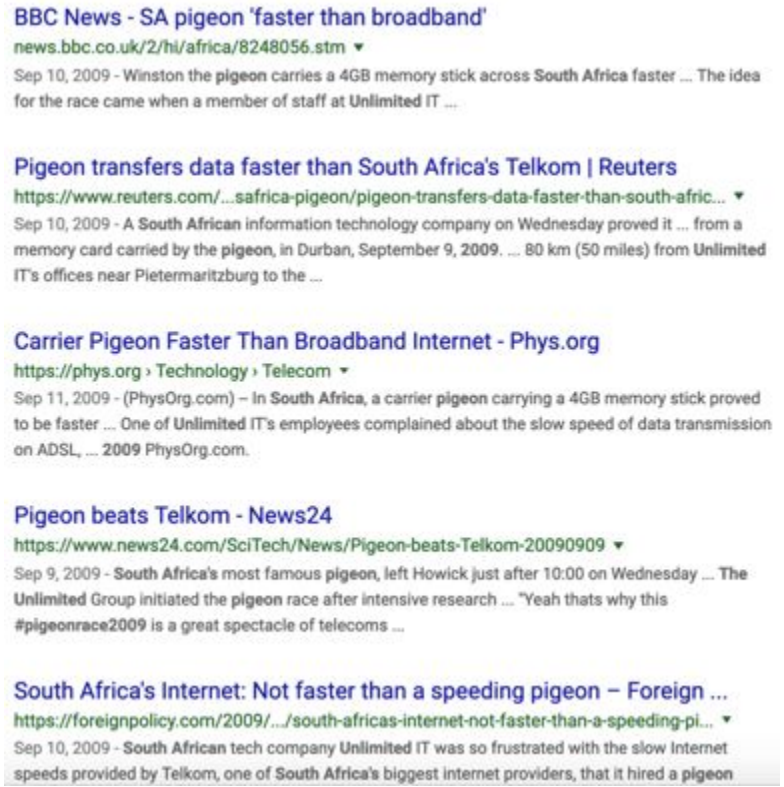
Running time **increases linearly** with array length.

Big O Notation

Big O - Meet “Winston The Carrier Pigeon”

The Unlimited - South Africa - 2009

Big O - “Pigeon Wins!”



BBC News - SA pigeon 'faster than broadband'
news.bbc.co.uk/2/hi/africa/8248056.stm ▾
Sep 10, 2009 - Winston the **pigeon** carries a 4GB memory stick across **South Africa** faster ... The idea for the race came when a member of staff at **Unlimited** IT ...

Pigeon transfers data faster than South Africa's Telkom | Reuters
https://www.reuters.com/...safrica-pigeon/pigeon-transfers-data-faster-than-south-afric... ▾
Sep 10, 2009 - A **South African** information technology company on Wednesday proved it ... from a memory card carried by the **pigeon**, in Durban, September 9, **2009**. ... 80 km (50 miles) from **Unlimited** IT's offices near Pietermaritzburg to the ...

Carrier Pigeon Faster Than Broadband Internet - Phys.org
https://phys.org › Technology › Telecom ▾
Sep 11, 2009 - (PhysOrg.com) – In **South Africa**, a carrier **pigeon** carrying a 4GB memory stick proved to be faster ... One of **Unlimited** IT's employees complained about the slow speed of data transmission on ADSL, ... **2009** PhysOrg.com.

Pigeon beats Telkom - News24
https://www.news24.com/SciTech/News/Pigeon-beats-Telkom-20090909 ▾
Sep 9, 2009 - **South Africa's** most famous **pigeon**, left Howick just after 10:00 on Wednesday ... **The Unlimited** Group initiated the **pigeon** race after intensive research ... "Yeah thats why this **#pigeonrace2009** is a great spectacle of telecoms ...

South Africa's Internet: Not faster than a speeding pigeon – Foreign ...
https://foreignpolicy.com/2009/.../south-africas-internet-not-faster-than-a-speeding-pi... ▾
Sep 10, 2009 - **South African** tech company **Unlimited** IT was so frustrated with the slow Internet speeds provided by Telkom, one of **South Africa's** biggest internet providers, that it hired a **pigeon**

What is a data structure?

What is a Data Structure?

01

A way of storing data so that it can be used efficiently by the computer or browser.

02

It is built upon simpler primitive data types (like variables).

03

It is non-opinionated, in the sense that it is only responsible for holding the data.

Example Data Structure: Arrays

```
var favFoods = ["Pickles", "Onions", "Carrots"]
```

Why is computer science a common category for technical interview questions?

Computer Science in Technical Interviews

01

Understanding computer science fundamentals shows that you know how a program functions and how to make it faster and more efficient.

02

Many candidates are computer science graduates.

03

Comprehending the use cases of data structures and the algorithms that manipulate them will help you create more performant applications.

Instructor Demonstration

Mini-Project