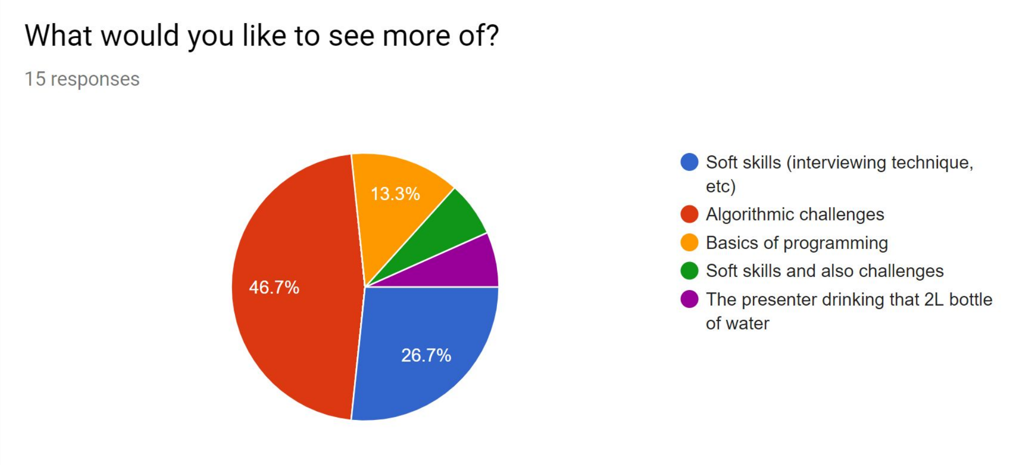
Technical Skills Challenge

Week 2: Asking Questions about questions about questions about questions about qu

Before We start...

Thanks for your feedback!

Here’s what people told us...

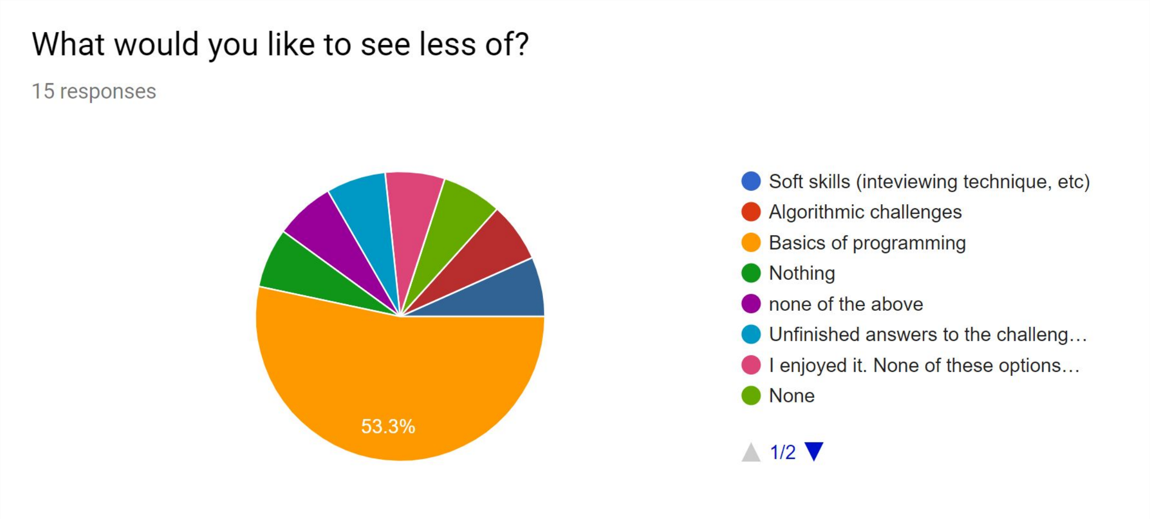


What would you like to see more of?

15 responses

Thanks for your feedback!

Here’s what people told us...



What would you like to see less of?

15 responses

Thanks for your feedback!

What went well…

- The friendly, non-judgmental atmosphere
- Working on stuff together
- Granular, easy to understand explanation of concepts

But we could improve on…

- Having harder questions as well for people with a bit more experience
- Being a bit better prepared!

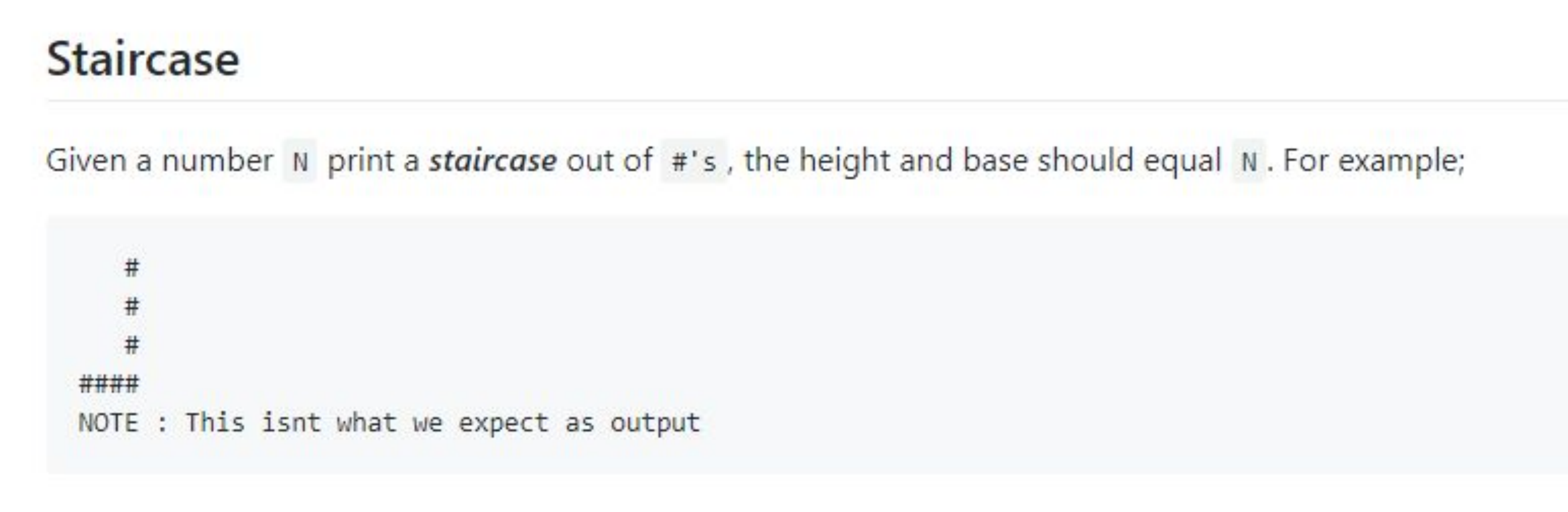
And people wanted to see us cover technical skills outside of algorithm stuff.

So what are we doing in response?

- We’re going to do this bi-weekly instead of weekly
  - (Next session will be the 30th of October, the night before Halloween. Spooky 👻).
  - That’ll give us more time to prepare this well around our jobs and courses.
- We’re trying out a new format that should make it easier to prepare without compromising on the usefulness, and have made some internal efficiency changes. We’ll be able to include new Committee Members in the process.
- There is a take home question which is a *little* more challenging, and an example of a google interview question in the github repo for a real challenge.
- If we are finding we’re able to reasonably prepare this bi-weekly, we’re going to look into running wider ranging technical/employability workshops on the unfilled Tuesdays (how to use git, basic tools, clean code principles, having a good LinkedIn profile, etc.).

Question 1: A Step Above the Rest

The question as asked...



Staircase

Given a number `N` print a ***staircase*** out of `#'s` , the height and base should equal `N` . For example;

```
   #
   #
   #
####
NOTE : This isnt what we expect as output
```

What Questions Would you Ask?

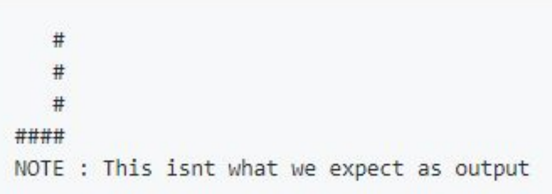
(2 minutes)

Things to think about:

- Is anything unclear?
- Are there any edge cases that aren't fully addressed?
- What data do you need?
- What qualifies a correct answer?
- Do you need to think about extensibility?
- What format should the returned data be in?
- Does your answer need to work for big inputs?

Print a staircase of base N and height N
E.g.

```
   #
  ##
 ###
####
```



```
   #
   #
   #
####
NOTE : This isnt what we expect as output
```

- What DO you expect as output? [ABOVE]
- Does the staircase have to face left? [YES]
- Is my font going to be uniform width? [YES]
- Does it have to be lined up like that? [YES]
- Does it have to have the largest at the bottom? [YES]
- Will it always step in by 1 hash? [YES]
- Does this string need to be stored or can it be printed? [PRINTED IS FINE]

How would you start answering this?

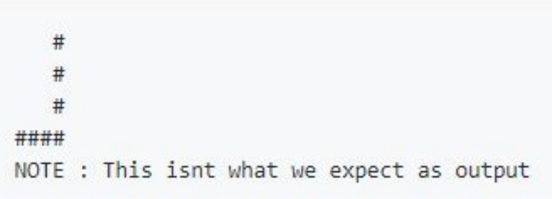
(3 minutes)

Things to think about:

- Do any problems/quick solutions jump out to you?
- What are your first assumptions? Are they correct?
- Are there any processes that are repeated?
- What code structures could this use? (If/Else, Loops, Recursion…)
- What datatypes are the input and output going to be? What format are they in?

Print a staircase of base N and height N
E.g.

```
   #
  ##
 ###
####
```



```
   #
   #
   #
####
NOTE : This isnt what we expect as output
```

- We have a repeated process of a known size so a For loop is probably most appropriate
- If this was going left to right this would be a slightly easier problem. How would we do that?
- How do we know how many spaces to type?

Let’s write some pseudocode!

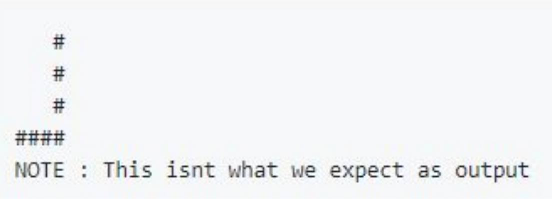
(10 Minutes)

Things to think about:

- Can you put any reused or difficult processes into their own functions so you can think about them separately?
- Are you trying to calculate exactly what the question is asking for? If not, is there an easier way to get that answer?
- Are there any edge cases that your algorithm doesn’t account for?

Print a staircase of base N and height N
E.g.

```
   #
  ##
 ###
####
```



```
   #
   #
   #
####
NOTE : This isnt what we expect as output
```

THINGS TO REMEMBER:

- We should probably use a for loop
- Our staircase needs to face left, not right
- We can make use of printing to make this easier

IF YOU HAVE ANY MORE QUESTIONS, ASK!

Now let’s write some real code!

(10 Minutes)

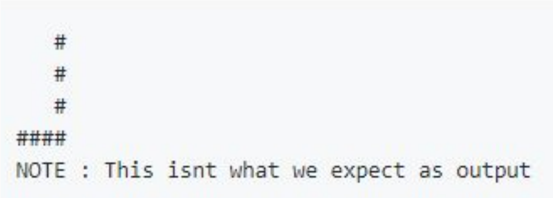
If you’re not happy with your pseudocode yet, keep trying! Ask others at your table, too.

Things to think about:

- Any plain english in your pseudocode can probably become the name of a function
- You might have to go back to your pseudocode if things you didn’t think about crop up.

Print a staircase of base N and height N
E.g.

```
   #
  ##
 ###
####
```



```
   #
   #
   #
####
NOTE : This isnt what we expect as output
```

THINGS TO REMEMBER:

- We should probably use a for loop
- Our staircase needs to face left, not right
- We can make use of printing to make this easier

IF YOU HAVE ANY MORE QUESTIONS, ASK!

Time!

Anybody figure it out (or close to) in pseudocode?

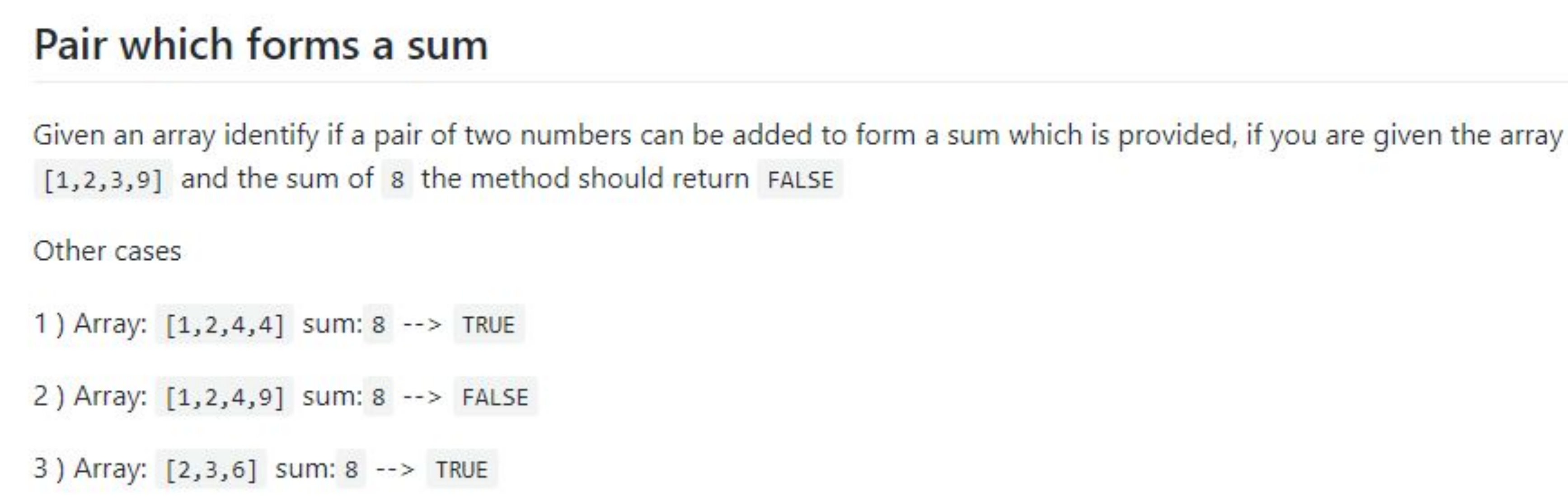
Anybody get any interesting revelations?

Anybody get it working (or close to) in real code?

Anybody want to show theirs off?

Question 2: Destined Pairs

The question as asked...



Pair which forms a sum

Given an array identify if a pair of two numbers can be added to form a sum which is provided, if you are given the array `[1,2,3,9]` and the sum of `8` the method should return `FALSE`

Other cases

1) Array: `[1,2,4,4]` sum: `8` --> `TRUE`

2) Array: `[1,2,4,9]` sum: `8` --> `FALSE`

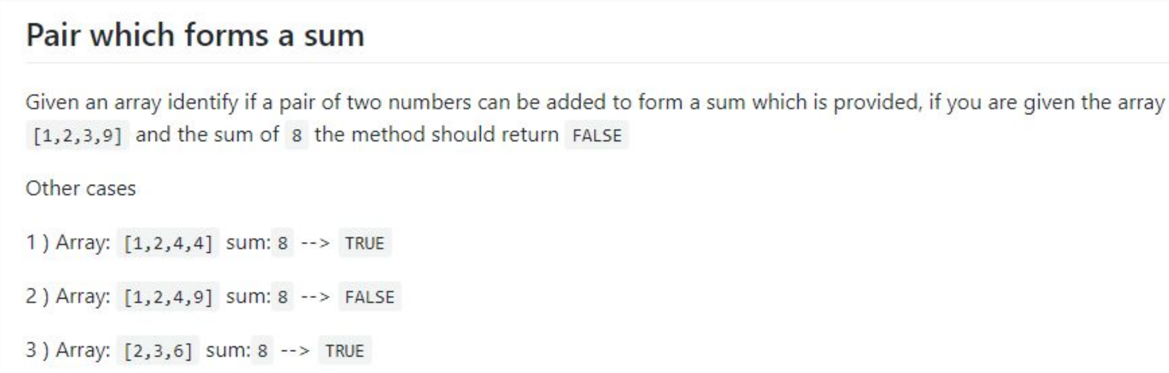
3) Array: `[2,3,6]` sum: `8` --> `TRUE`

What Questions Would you Ask?

(2 minutes)

Things to think about:

- Is anything unclear?
- Are there any edge cases that aren't fully addressed?
- What data do you need?
- What qualifies a correct answer?
- Do you need to think about extensibility?
- What format should the returned data be in?
- Does your answer need to work for big inputs?



Pair which forms a sum

Given an array identify if a pair of two numbers can be added to form a sum which is provided, if you are given the array `[1,2,3,9]` and the sum of `8` the method should return `FALSE`

Other cases

1) Array: `[1,2,4,4]` sum: `8` --> `TRUE`

2) Array: `[1,2,4,9]` sum: `8` --> `FALSE`

3) Array: `[2,3,6]` sum: `8` --> `TRUE`

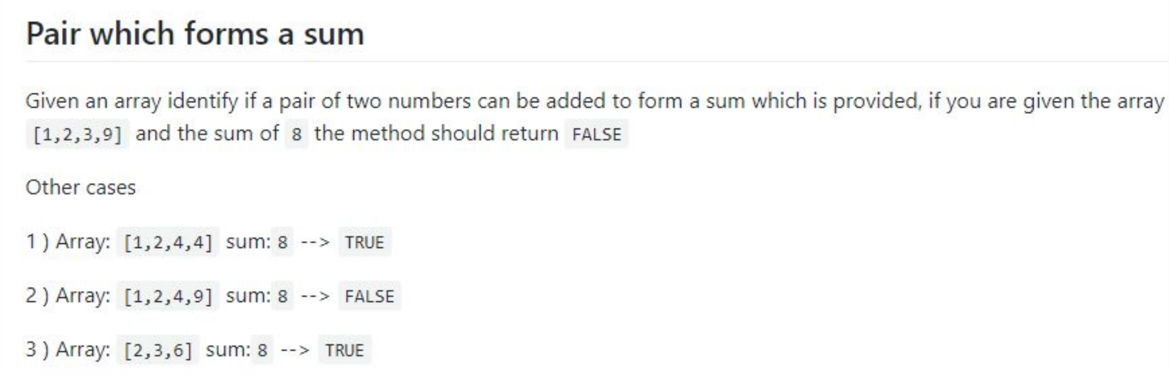
- Do we need to account for decimals? [NO]
- Do we need to account for negative numbers? [YEAH]
- What if the number is already in the array? [NEEDS TO BE A PAIR]
- Do we need to account for really large input arrays [PREFERABLY YEAH]
- Does the order of the array matter? [YES!! YOU CAN ASSUME THE ARRAY IS SORTED]

How would you start answering this?

(3 minutes)

Things to think about:

- Do any problems/quick solutions jump out to you?
- What are your first assumptions? Are they correct?
- Are there any processes that are repeated?
- What code structures could this use? (If/Else, Loops, Recursion…)
- What datatypes are the input and output going to be? What format are they in?



Pair which forms a sum

Given an array identify if a pair of two numbers can be added to form a sum which is provided, if you are given the array `[1,2,3,9]` and the sum of `8` the method should return `FALSE`

Other cases

1) Array: `[1,2,4,4]` sum: `8` --> `TRUE`

2) Array: `[1,2,4,9]` sum: `8` --> `FALSE`

3) Array: `[2,3,6]` sum: `8` --> `TRUE`

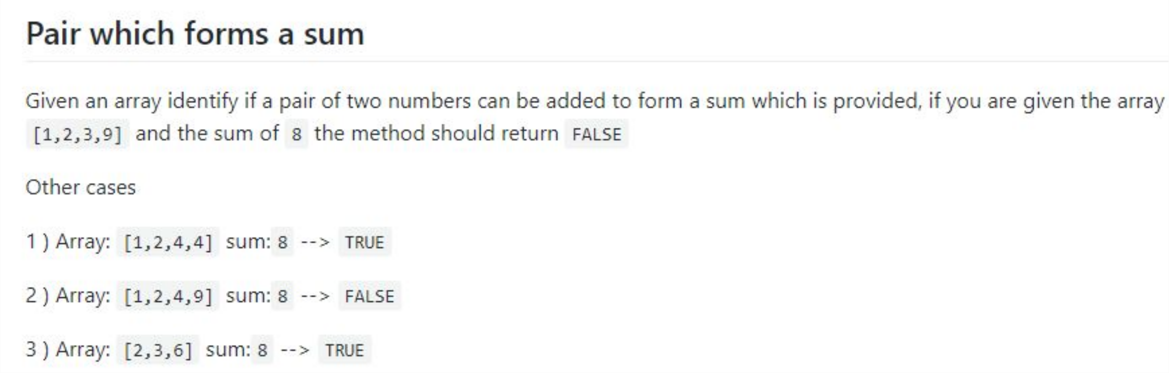
- We can probably use a for loop to check each element of this array
- How do we make sure we don’t add a number to itself?
- Can we use the value of our last result to inform what we do next?

Let’s write some pseudocode!

(15 Minutes)

Things to think about:

- Can you put any reused or difficult processes into their own functions so you can think about them separately?
- Are you trying to calculate exactly what the question is asking for? If not, is there an easier way to get that answer?
- Are there any edge cases that your algorithm doesn’t account for?



Pair which forms a sum

Given an array identify if a pair of two numbers can be added to form a sum which is provided, if you are given the array `[1,2,3,9]` and the sum of `8` the method should return `FALSE`

Other cases

1) Array: `[1,2,4,4]` sum: `8` --> `TRUE`

2) Array: `[1,2,4,9]` sum: `8` --> `FALSE`

3) Array: `[2,3,6]` sum: `8` --> `TRUE`

THINGS TO REMEMBER:

- The arrays are always in ascending order!!
- We don’t need to return the pair, just True/False
- Make sure you’re not counting the same number twice
- You don’t need to worry about running across the number itself.
- Can we use our last result to inform what we do next?

IF YOU HAVE ANY MORE QUESTIONS, ASK!

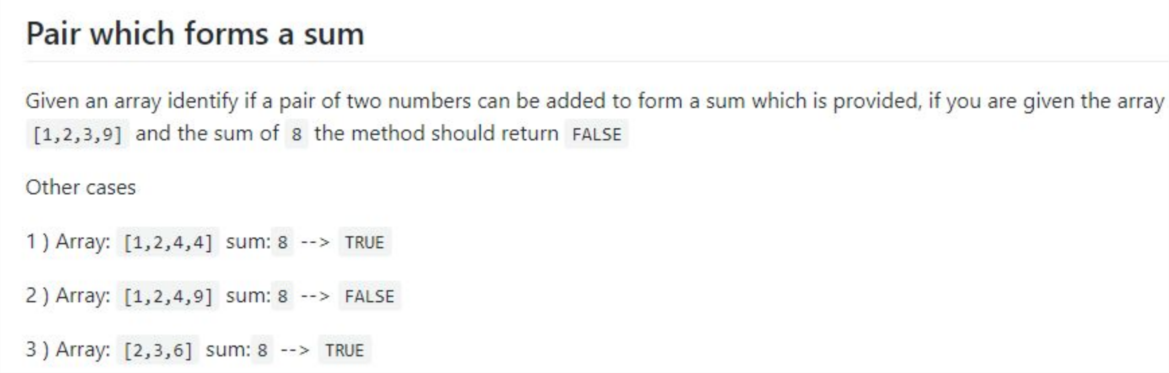
Now let’s write some real code!

(15 Minutes)

If you’re not happy with your pseudocode yet, keep trying! Ask others at your table, too.

Things to think about:

- Any verbs in your pseudocode can probably indicate a comparison or function call, nouns could become variables. Full sentences could become functions names.
- You might have to go back to your pseudocode if things you didn’t think about crop up.



Pair which forms a sum

Given an array identify if a pair of two numbers can be added to form a sum which is provided, if you are given the array `[1,2,3,9]` and the sum of `8` the method should return `FALSE`

Other cases

1) Array: `[1,2,4,4]` sum: `8` --> `TRUE`

2) Array: `[1,2,4,9]` sum: `8` --> `FALSE`

3) Array: `[2,3,6]` sum: `8` --> `TRUE`

THINGS TO REMEMBER:

- The arrays are always in ascending order!!
- We don’t need to return the pair, just True/False
- Make sure you’re not counting the same number twice
- You don’t need to worry about running across the number itself.
- Can we use our last result to inform what we do next?

IF YOU HAVE ANY MORE QUESTIONS, ASK!

Time!

Anybody figure it out (or close to) in pseudocode?

Anybody get any interesting revelations?

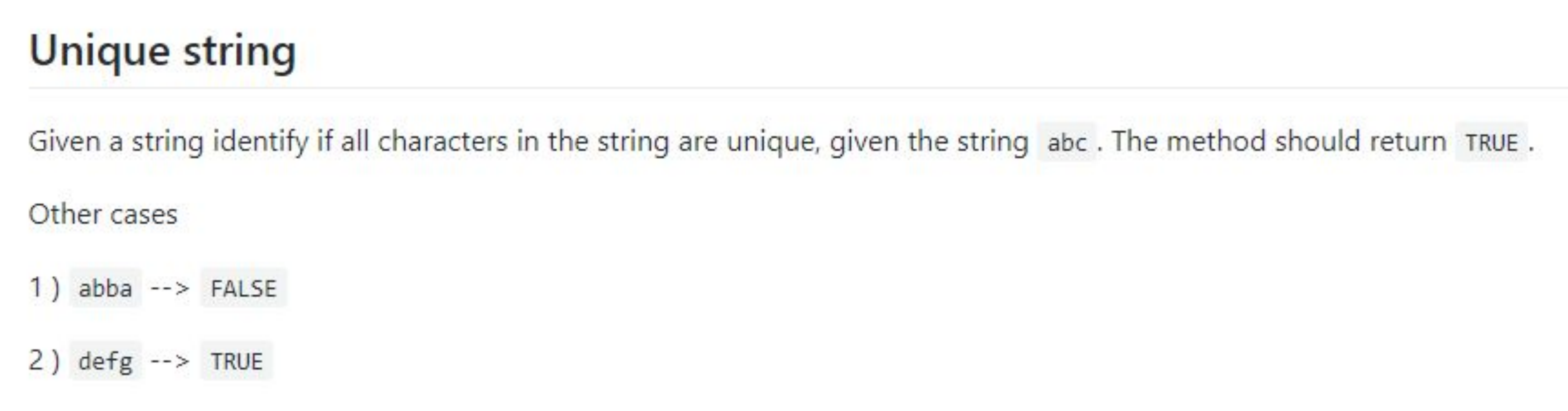
Anybody get it working (or close to) in real code?

Anybody want to show theirs off?

Finally, one to grow on...

Question 3: I am! The one and only!

The question as asked...



Unique string

Given a string identify if all characters in the string are unique, given the string `abc`. The method should return `TRUE`.

Other cases

1) `abba` --> `FALSE`

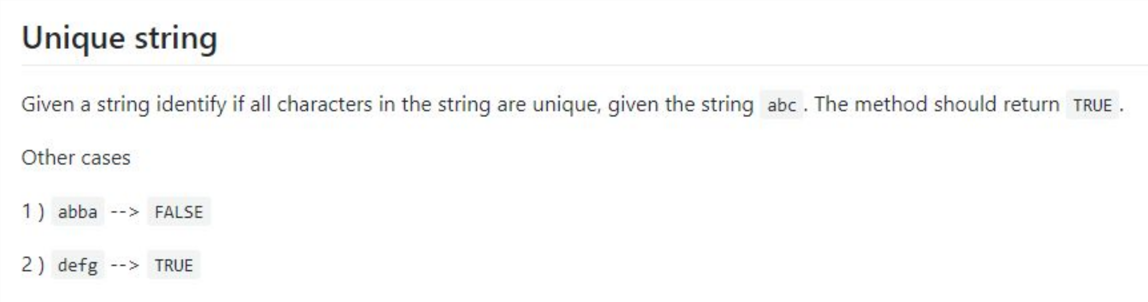
2) `defg` --> `TRUE`

What Questions Would you Ask?

(2 minutes)

Things to think about:

- Is anything unclear?
- Are there any edge cases that aren't fully addressed?
- What data do you need?
- What qualifies a correct answer?
- Do you need to think about extensibility?
- What format should the returned data be in?
- Does your answer need to work for big inputs?



Unique string

Given a string identify if all characters in the string are unique, given the string `abc` . The method should return `TRUE` .

Other cases

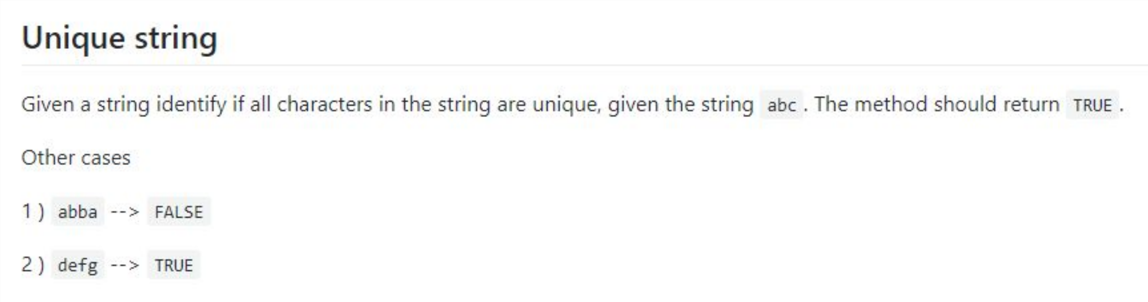
1) `abba` --> `FALSE`

2) `defg` --> `TRUE`

- Do you need to think about capital letters? [NO]
- Do you need to account for punctuation? [NO]
- Does it matter if they're adjacent? [NO]

Thank you for coming!

- Our next session will be **30th of October**
- You can access this and all previous sessions’ files at https://github.com/QCSQUB
- Like our facebook page: facebook.com/QCSQUB
- And like this event for updates.



Unique string

Given a string identify if all characters in the string are unique, given the string `abc`. The method should return `TRUE`.

Other cases

1) `abba` --> `FALSE`

2) `defg` --> `TRUE`

- Do you need to think about capital letters? [NO]
- Do you need to account for punctuation? [NO]
- Does it matter if they’re adjacent? [NO]

Register for the ICPC before tomorrow and flex your skills this saturday!

https://icpc.baylor.edu/regionals/finder/ukiepc-2018