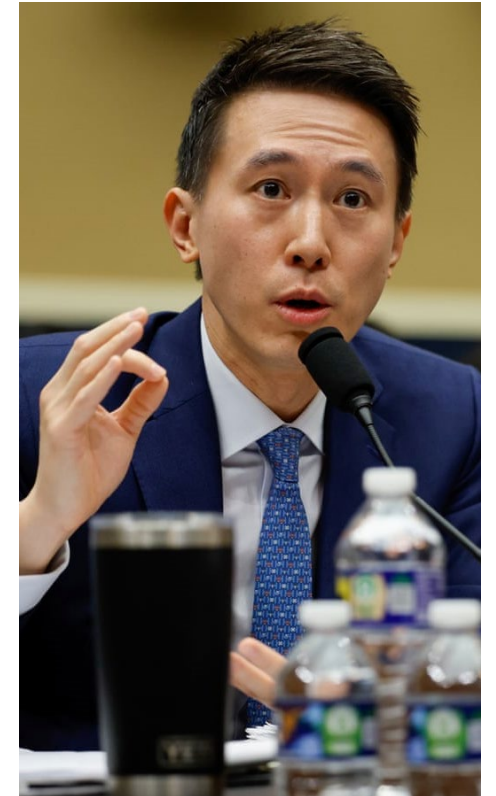


survey: which ones have you watched



COMP1511 Week 7!

William's Flex Week: Assignments + Suits S1-3

The Agenda

Assignment 1

Assignment 1 Debrief!

- How was Assignment 1?
 - 2D arrays vibes post-Assignment 1?
 - Assignment 2 tips 'n tricks? Share!
- Help Sessions still running!
- Check-in week this week!

Functions and Pointers

Let's code, but first...

- Problems with editing multiple variables in a helper function:
 - Cannot return multiple variables
- Solutions:
 - Pass a **struct** and copy from its values:
 - Create **struct**
 - Pass **struct**
 - Return **struct**
 - Copy from **struct**
 - Pass a pointer and edit directly:
 - Pass pointer
 - Profit!

Malloc and Arrays

Let's code, but first...

- What does **malloc** stand for?
- What does **sizeof** stand for?
- What does **malloc** take as an argument?
- How is memory allocated with **malloc**?

Pointers

How do pointers work?

```
int n = 42;
int *p;
int *q;
p = &n;
*p = 5;
*q = 17;
q = p;
*q = 8;
```

Let's step
through this
together!

Struct Pointers

Let's code, but first...

- Does ***** (dereference) work on **structs**?
- Does ***** work on field accessing?
 - i.e. ***my_struct.field**
- **->** is shorthand!

EOF Loops

Let's code, but first...

- What's the difference between **scanf** and **fgets**?
- What does **fgets** stand for?
- How does **fgets** read strings?

Assignment 1 Debrief!

- How was Assignment 1?
 - 2D arrays vibes post-Assignment 1?
 - Assignment 2 tips 'n tricks? Share!
- Help Sessions still running!
- Check-in week this week!

How do pointers work?

```
int n = 42;  
int *p;  
int *q;  
p = &n;  
*p = 5;  
*q = 17;  
q = p;  
*q = 8;
```

Let's step
through this
together!



Let's code, but first...

- Problems with editing multiple variables in a helper function:
 - Cannot return multiple variables
- Solutions:
 - Pass a **struct** and copy from its values:
 - Create **struct**
 - Pass **struct**
 - Return **struct**
 - Copy from **struct**
 - Pass a pointer and edit directly:
 - Pass pointer
 - Profit!



Let's code, but first...

- Does * (dereference) work on **structs**?
- Does * work on field accessing?
 - i.e. `*my_struct.field`
- `->` is shorthand!



Let's code, but first...

- What does **malloc** stand for?
- What does **sizeof** stand for?
- What does **malloc** take as an argument?
- How is memory allocated with **malloc**?

Let's code, but first...

- What's the difference between **scanf** and **fgets**?
- What does **fgets** stand for?
- How does **fgets** read strings?