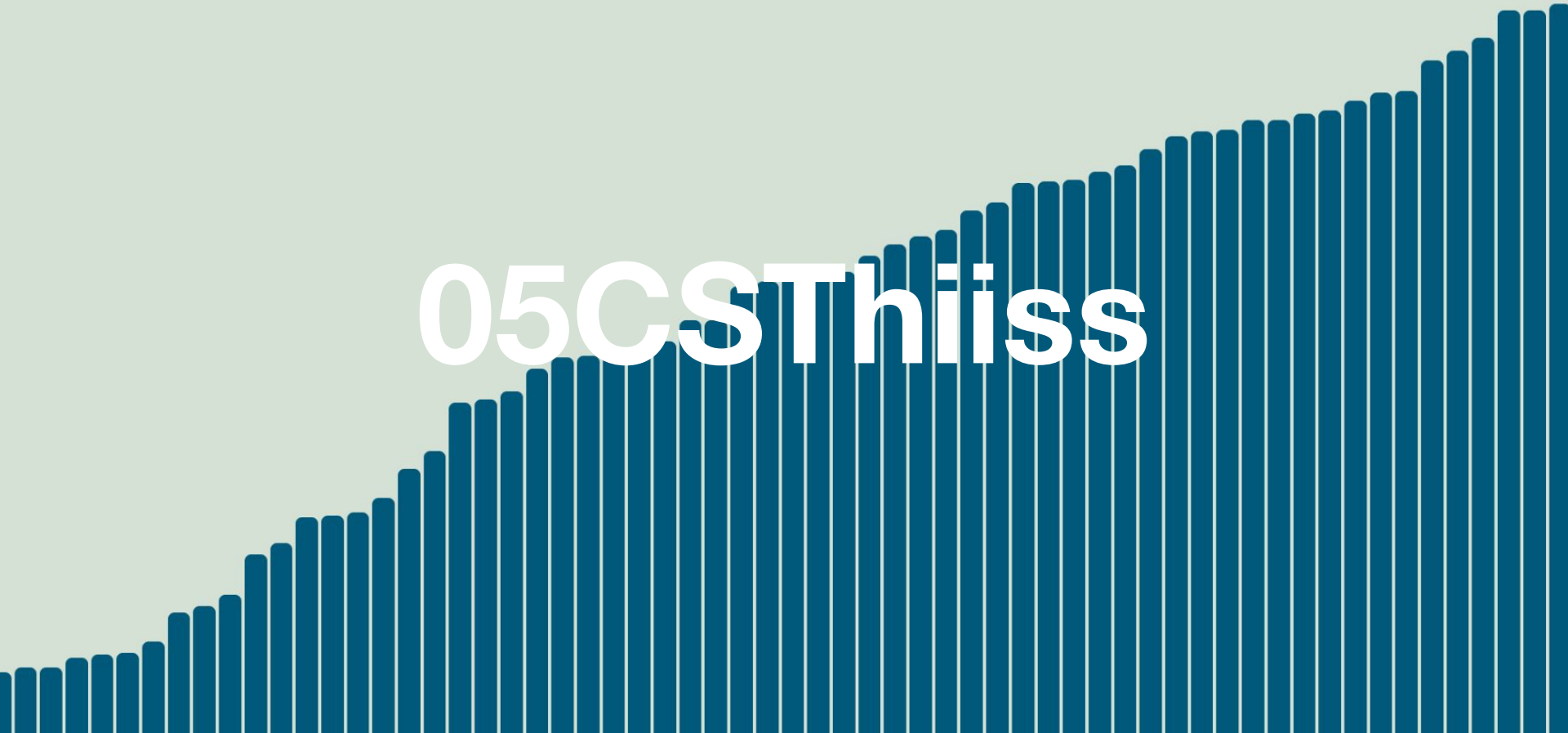

05CSThiiss

Think.
Pair.
Share.

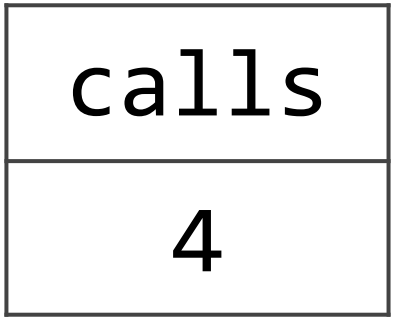
cs50.ly/questions

- What are **pointers**, and how can we become familiar with their **syntax**?
- How can we **read** and **write** data from a file?
- What is **dynamic memory**, and how should we use it?
- How does a computer modify the **volume** of audio?

Pointers

Variables

```
int calls = 4;
```



Variables

```
int calls = 4;
```

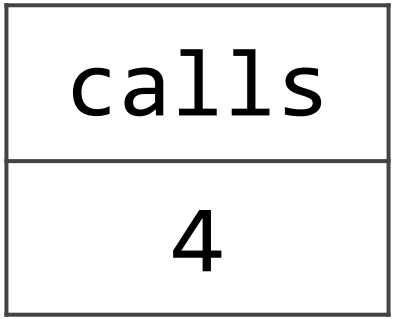
name

calls
4

Variables

```
int calls = 4;
```

type



Variables

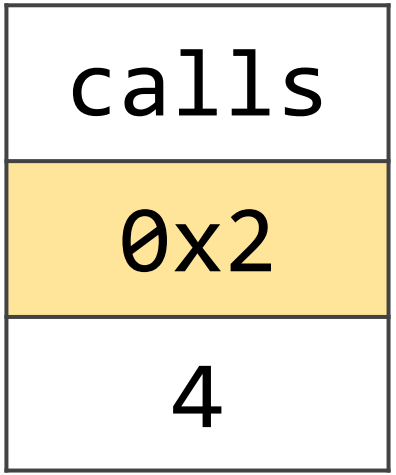
```
int calls = 4;
```

value

calls
4

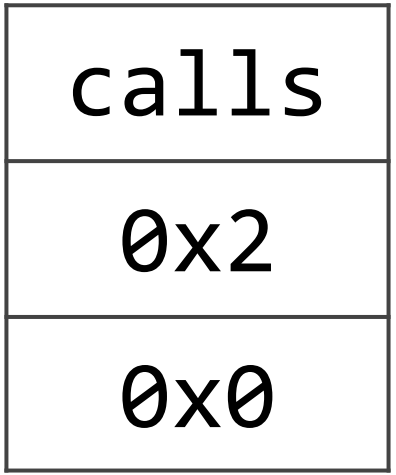
Variables

```
int calls = 4;
```



Pointers

```
int *calls = 0x0;
```



Pointers

```
int *calls = 0x0;
```

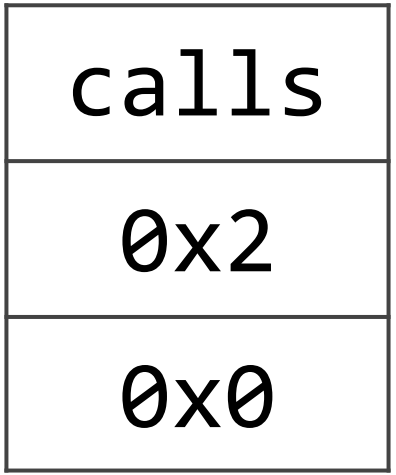
name

calls
0x2
0x0

Pointers

```
int *calls = 0x0;
```

type



Pointers

```
int *calls = 0x0;
```

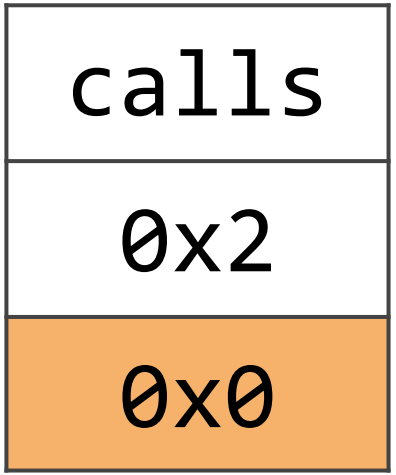
value

calls
0x2
0x0

Pointer Syntax

```
calls;
```

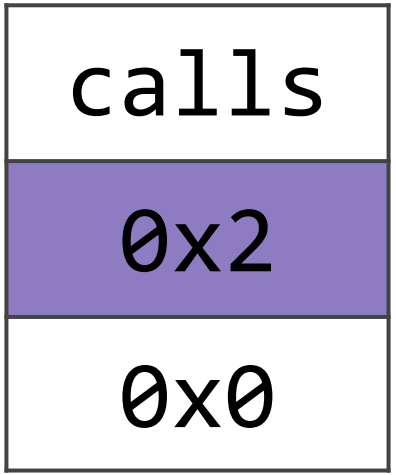
"value of"



Pointer Syntax

`&calls;`

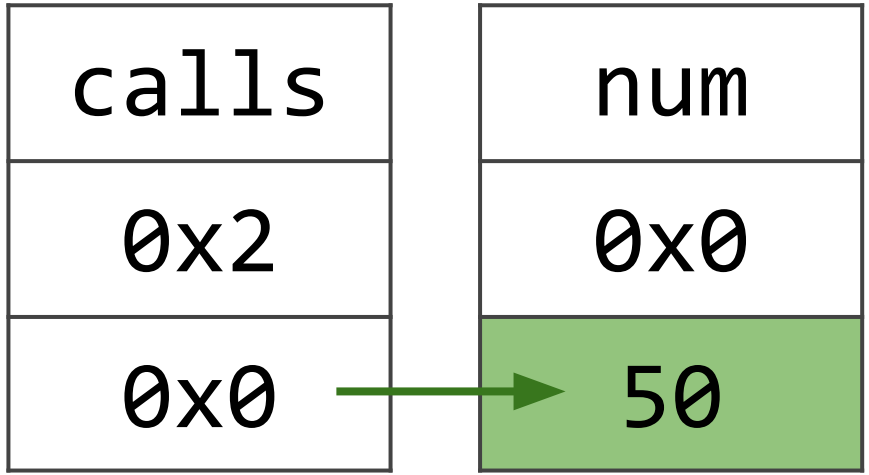
"address of"



Pointer Syntax

```
*calls;
```

"**go to** address stored in calls"



`type *` is a pointer that stores the address of a **`type`**.

`*x` takes a pointer **`x`** and goes to the address stored at that pointer.

`&x` takes **`x`** and gets its address.

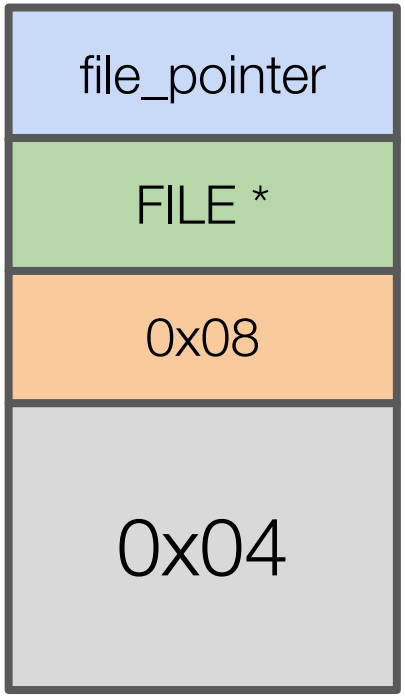
Pointer Prediction Exercise

Go to **cs50.ly/pointer-exercise**.

Visualize the code on the left, step by step. How do the values of the variables and pointers evolve? It's okay to use made-up addresses.

What will the final values for each variable or pointer be? Download, compile, and run **pointers.c** in VS Code to find out.

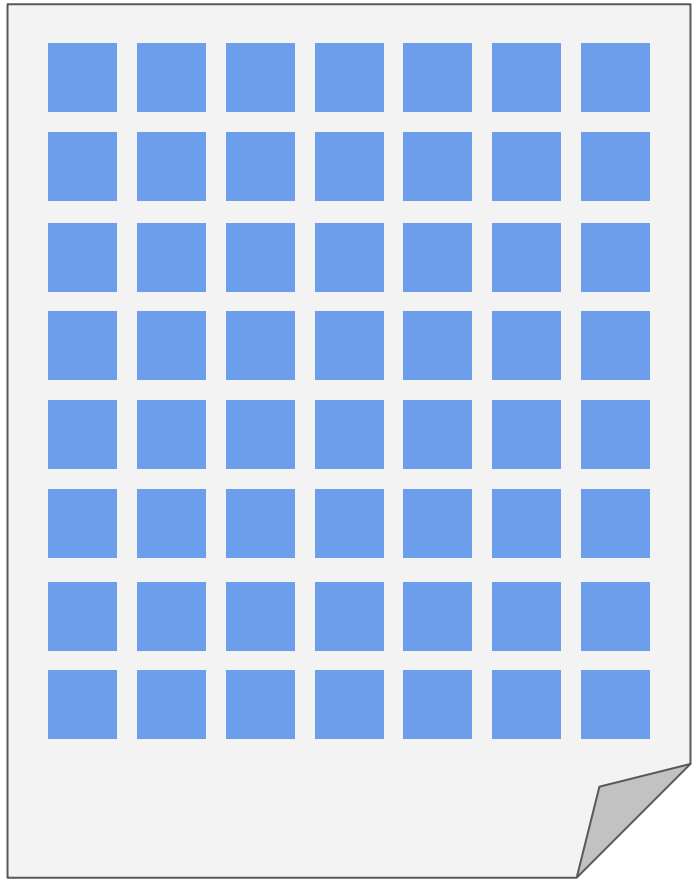
File I/O




```
FILE *file_pointer =
fopen("test.txt", "r");
```

file_pointer

file_pointer

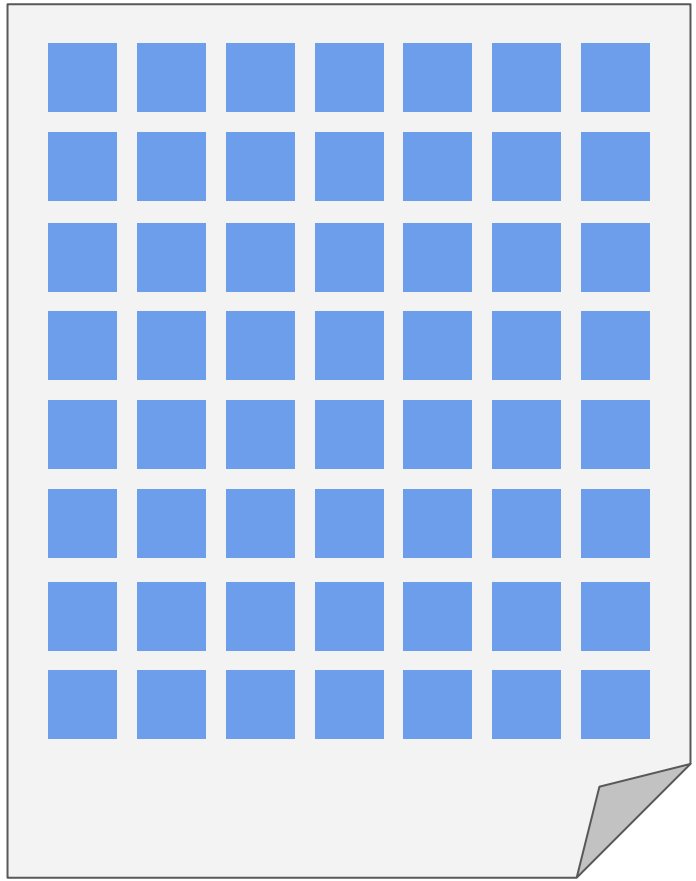


```
fread(buffer, 1, 4, file_pointer);
```

```
fread(buffer, 1, 4, file_pointer);
```

Location to read from

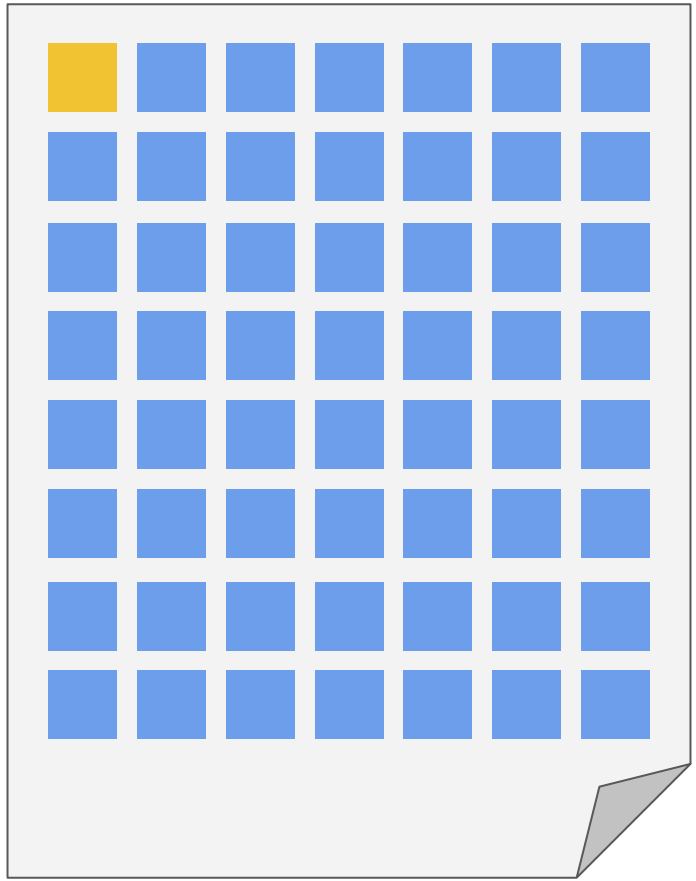
file_pointer



```
fread(buffer, 1, 4, file_pointer);
```

Size of blocks to read (in bytes)

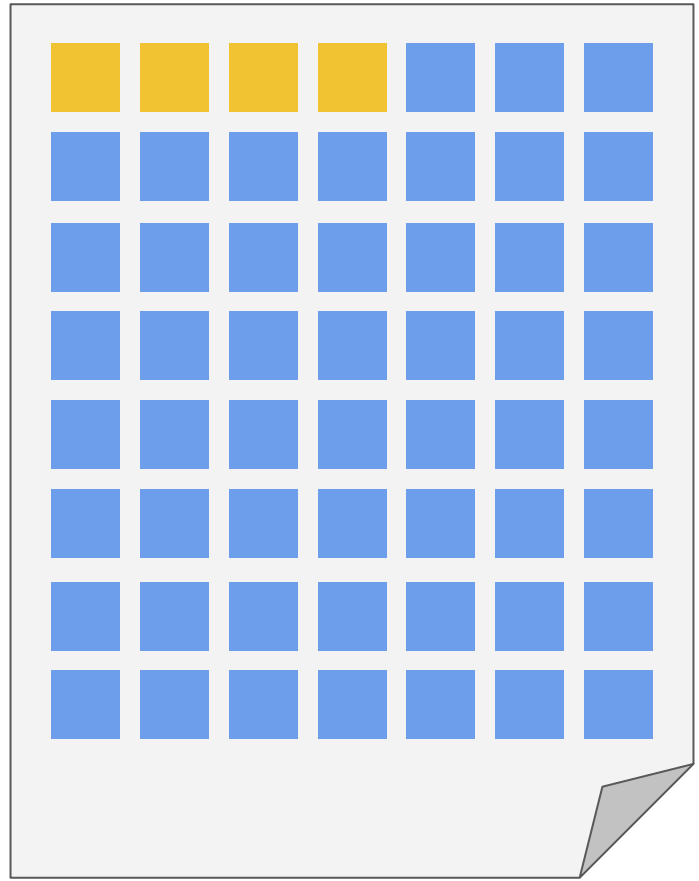
file_pointer



```
fread(buffer, 1, 4, file_pointer);
```

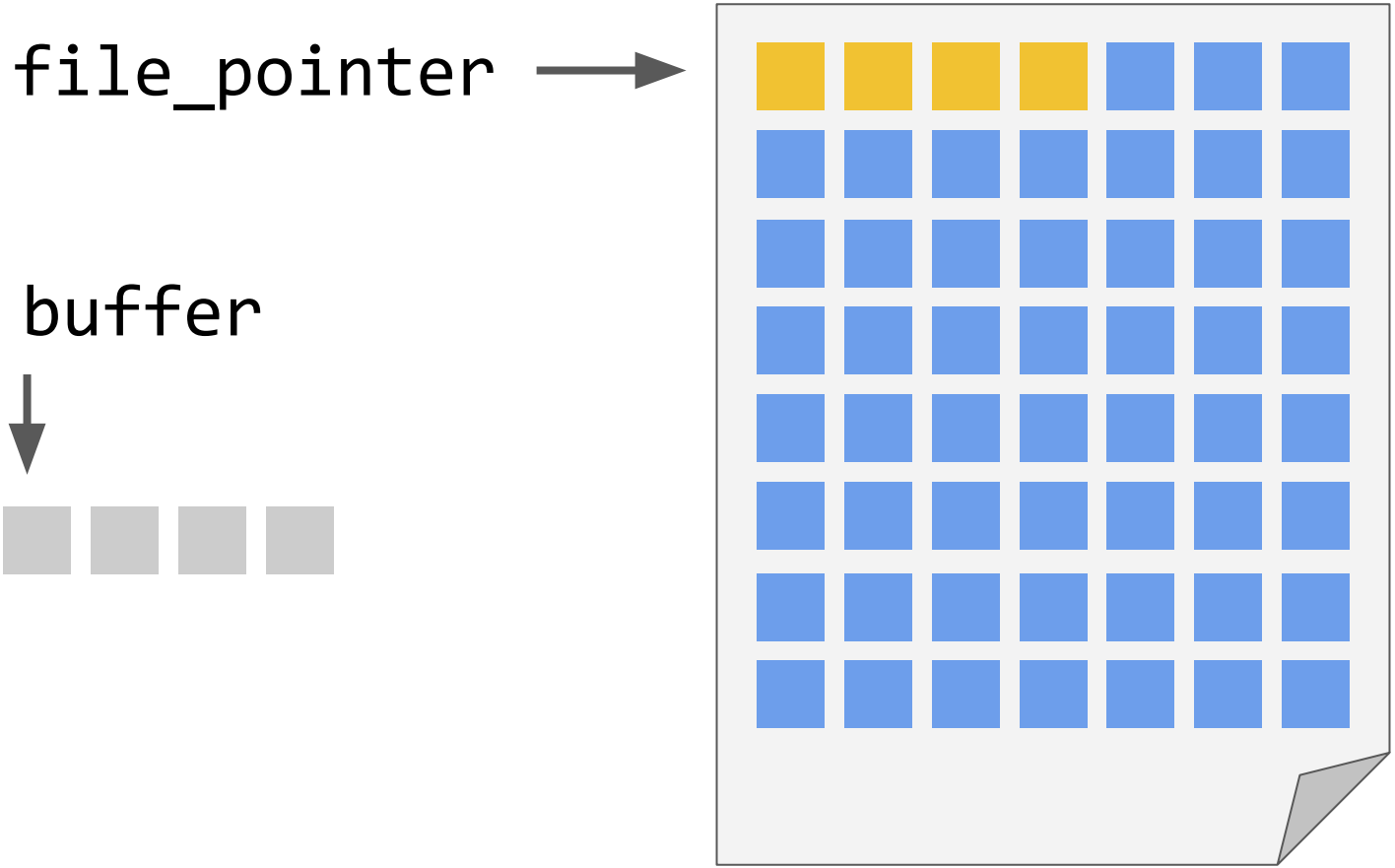
How many blocks to read

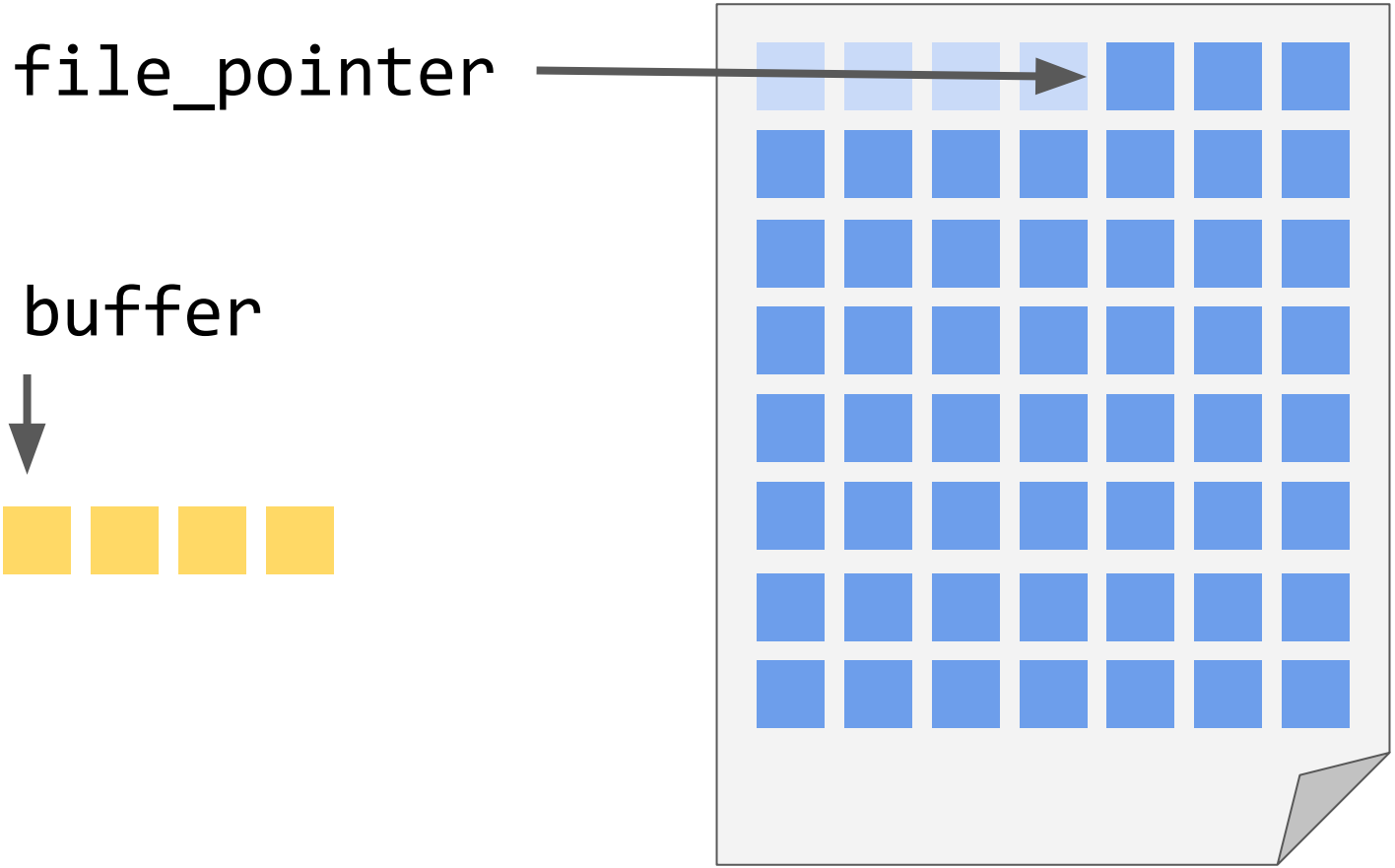
file_pointer

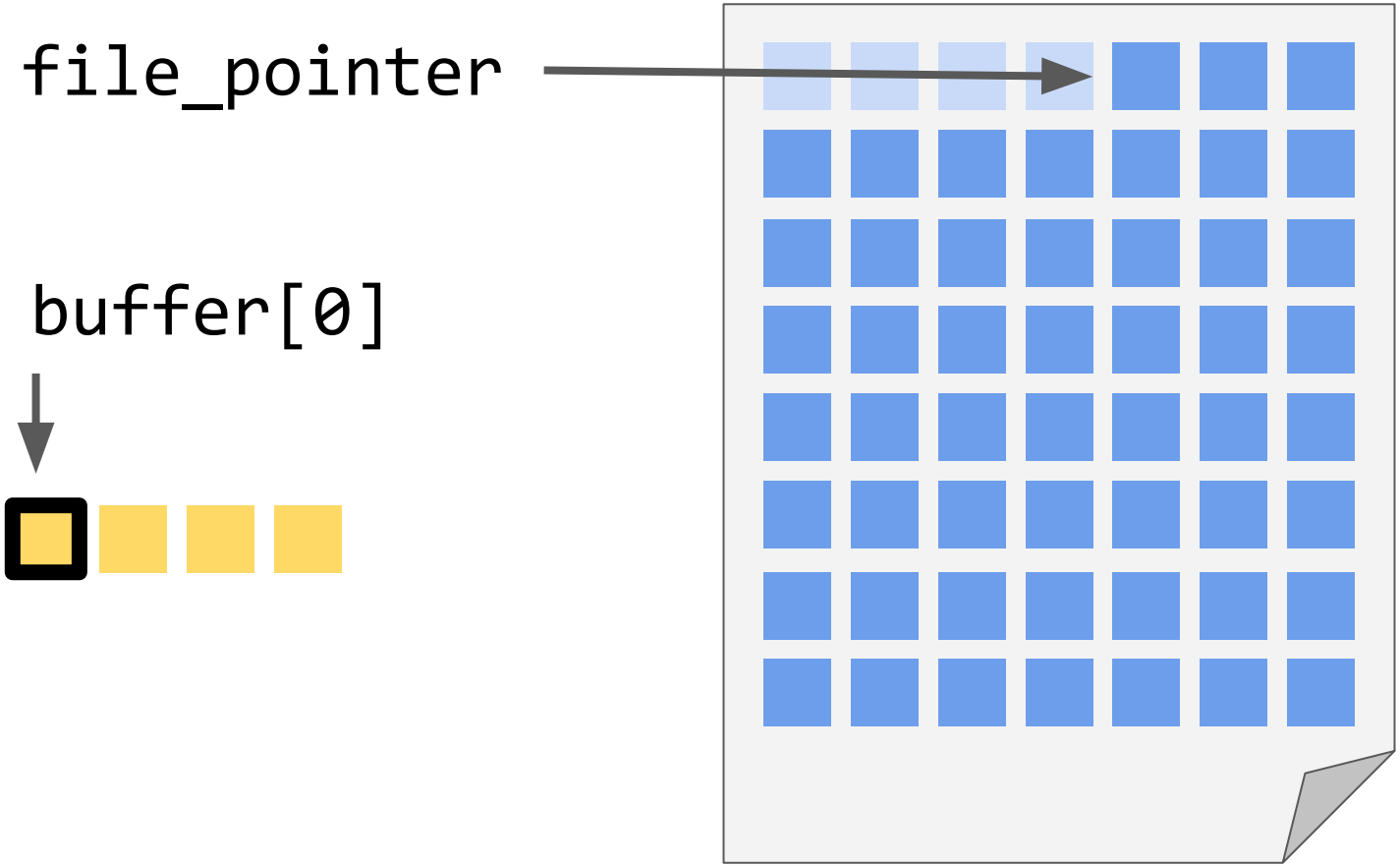


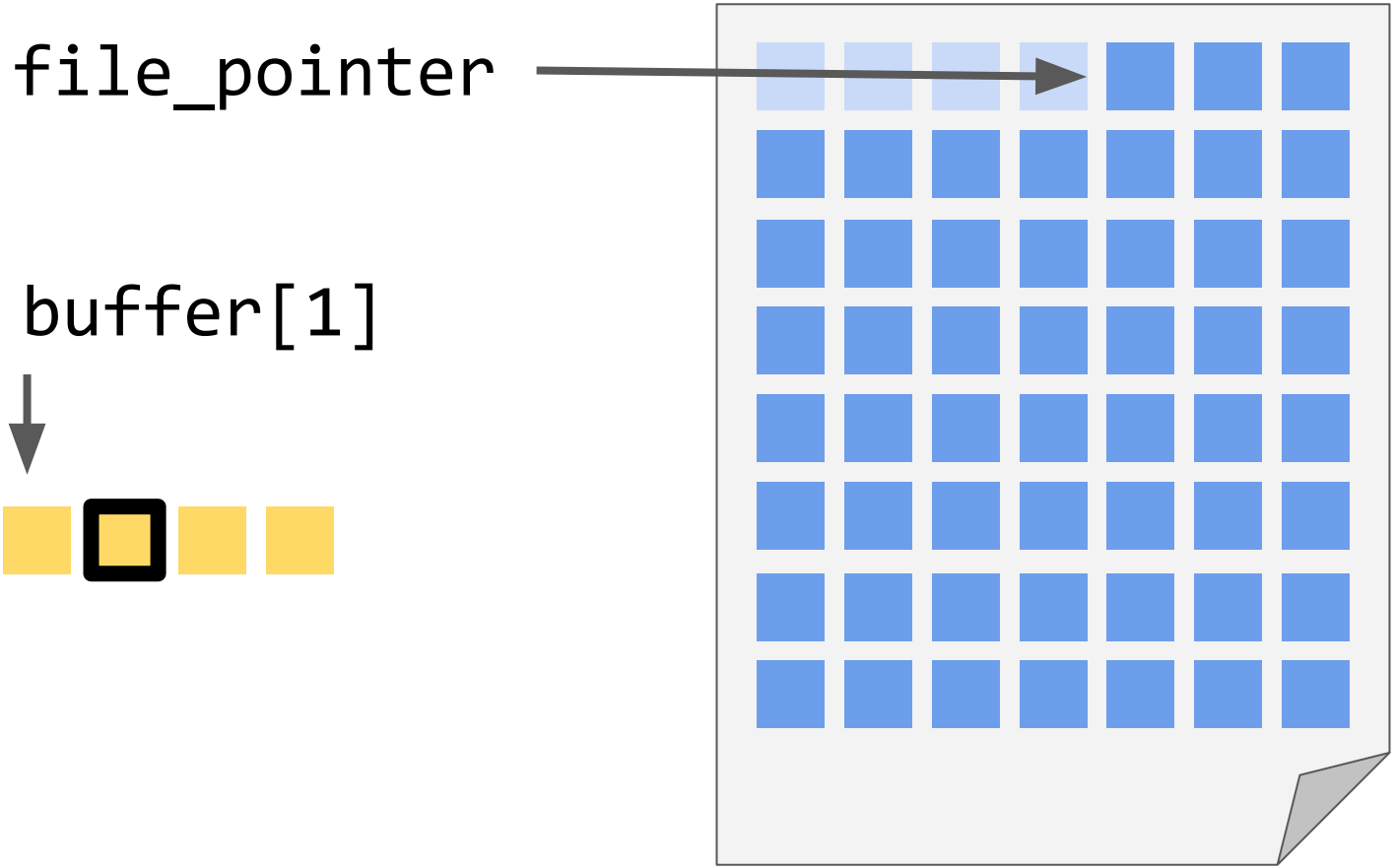
```
fread(buffer, 1, 4, file_pointer);
```

Location to store blocks


file_pointer
buffer

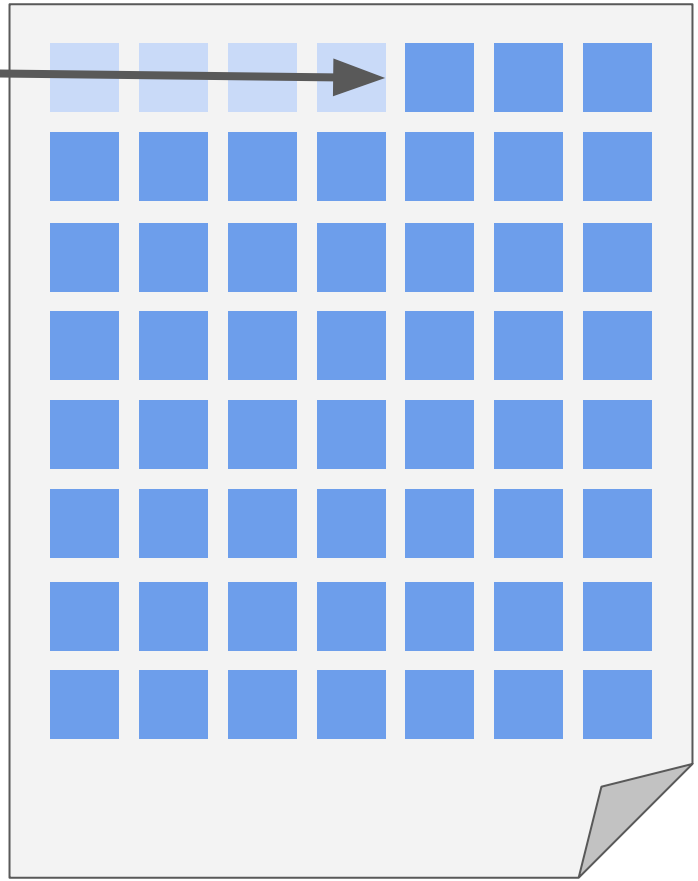

file_pointer
buffer


file_pointer
buffer[0]


file_pointer
buffer[1]

file_pointer

buffer[2]



```
fwrite(buffer, 1, 4, output_file);
```


output_file
buffer


output_file
buffer

File Reading Exercise

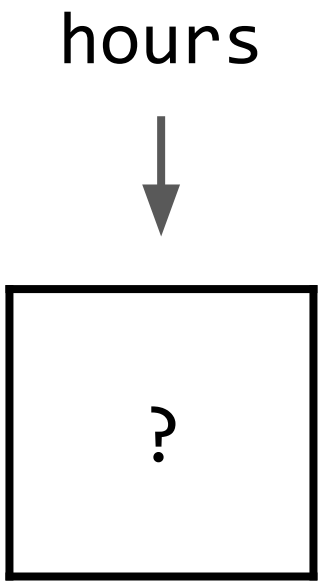
Create a program, **pdf.c**, that checks whether a file, passed in as a command-line argument, is a PDF. All PDFs will begin with a four byte sequence:

```
0x25 0x50 0x44 0x46
```

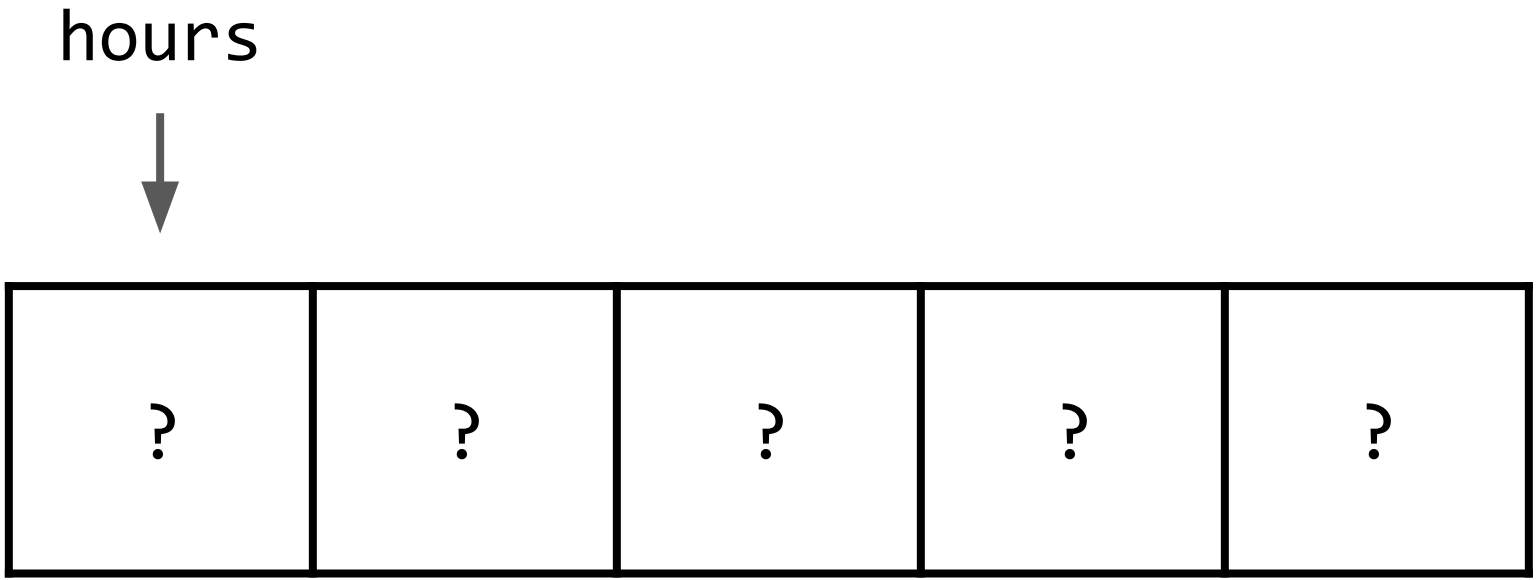
Use the **.pdf** and **.jpg** files in the section resources page to check your work.

Dynamic Memory

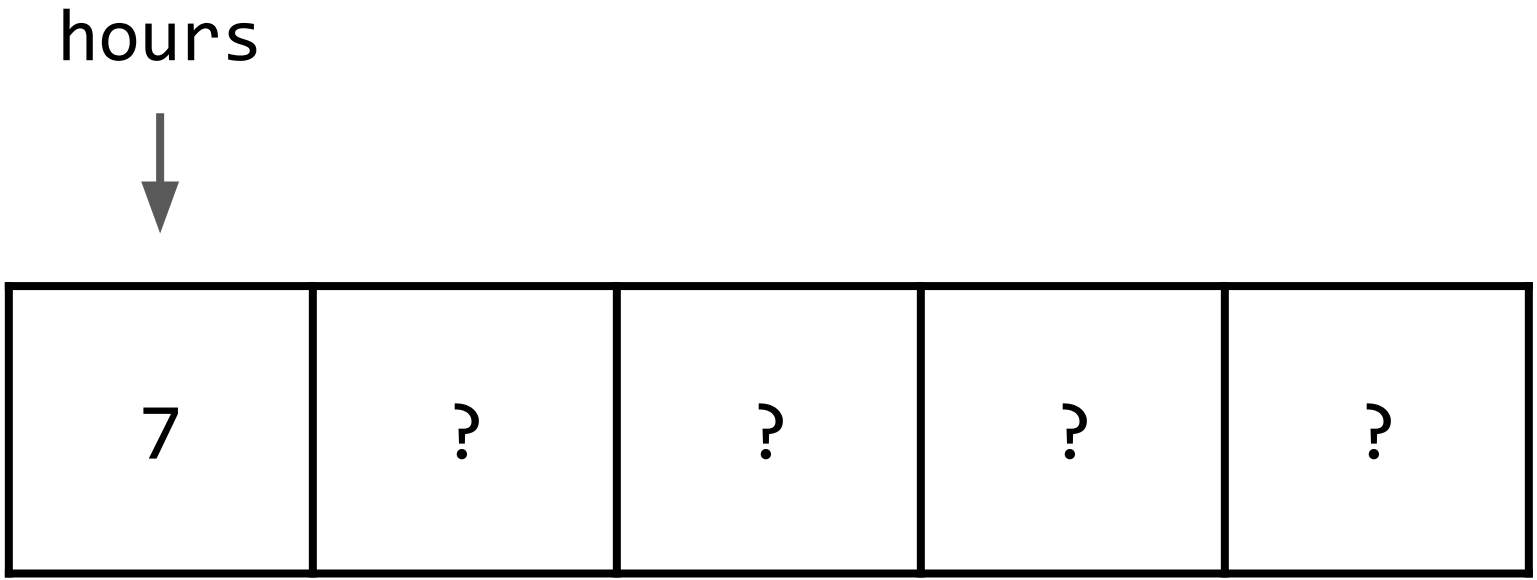
```
int *hours = malloc(sizeof(int));
```



```
int *hours = malloc(sizeof(int) * 5);
```

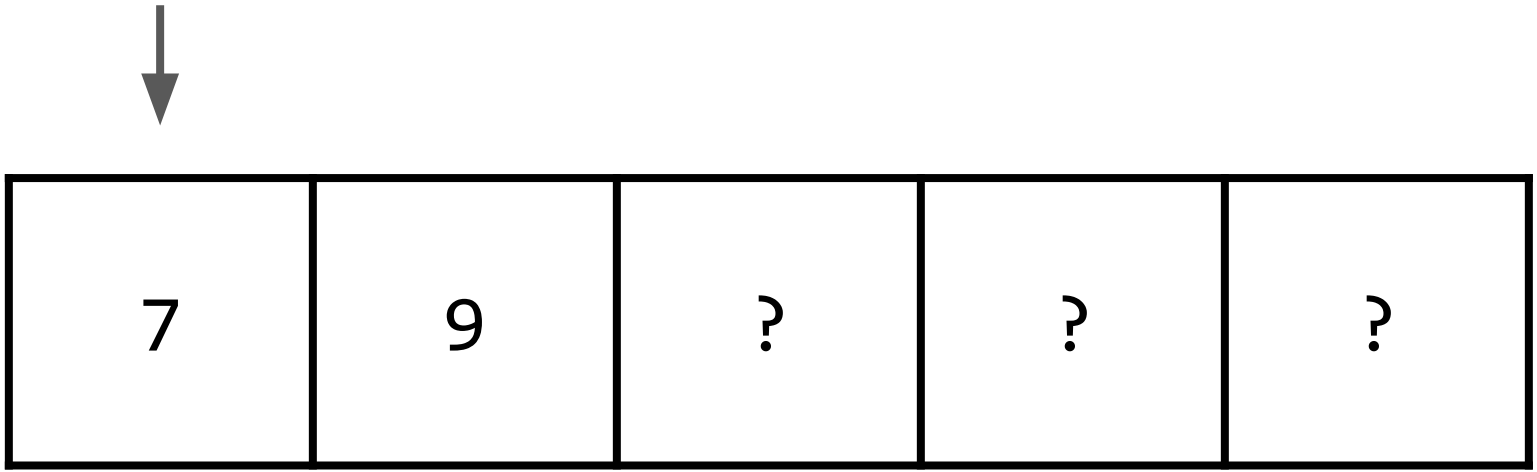


```
*hours = 7;
```

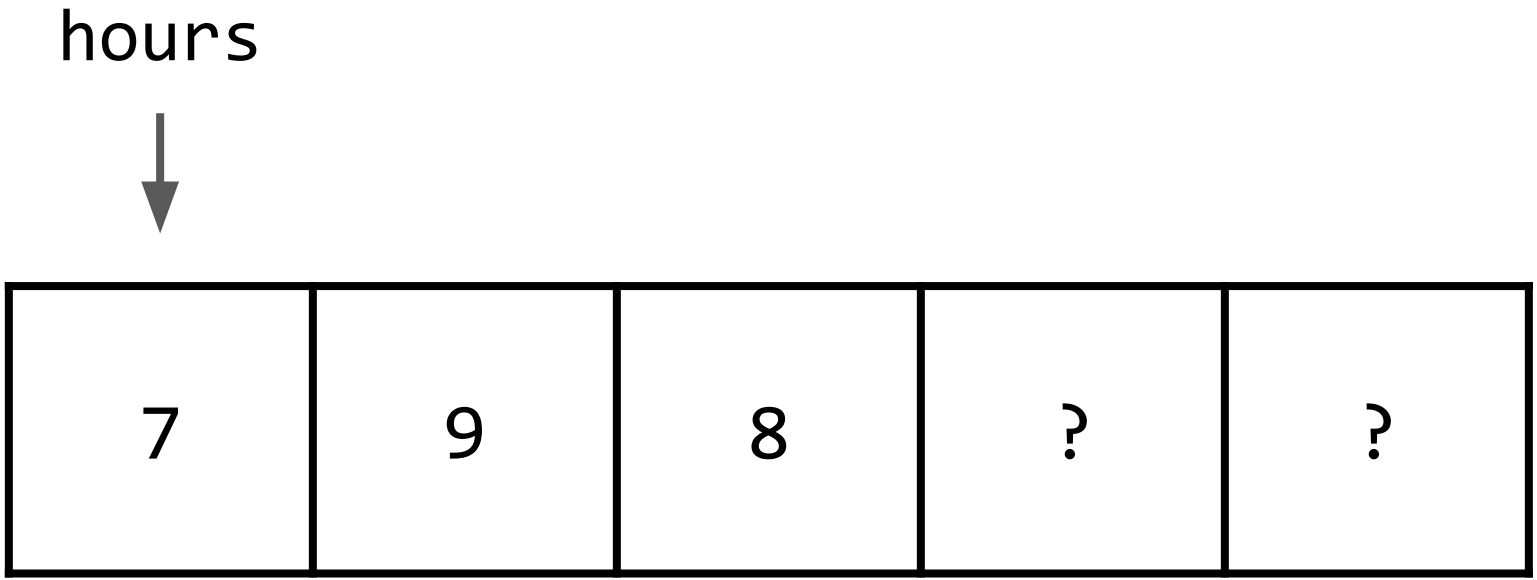


```
*hours = 7;
*(hours + 1) = 9;
```

hours

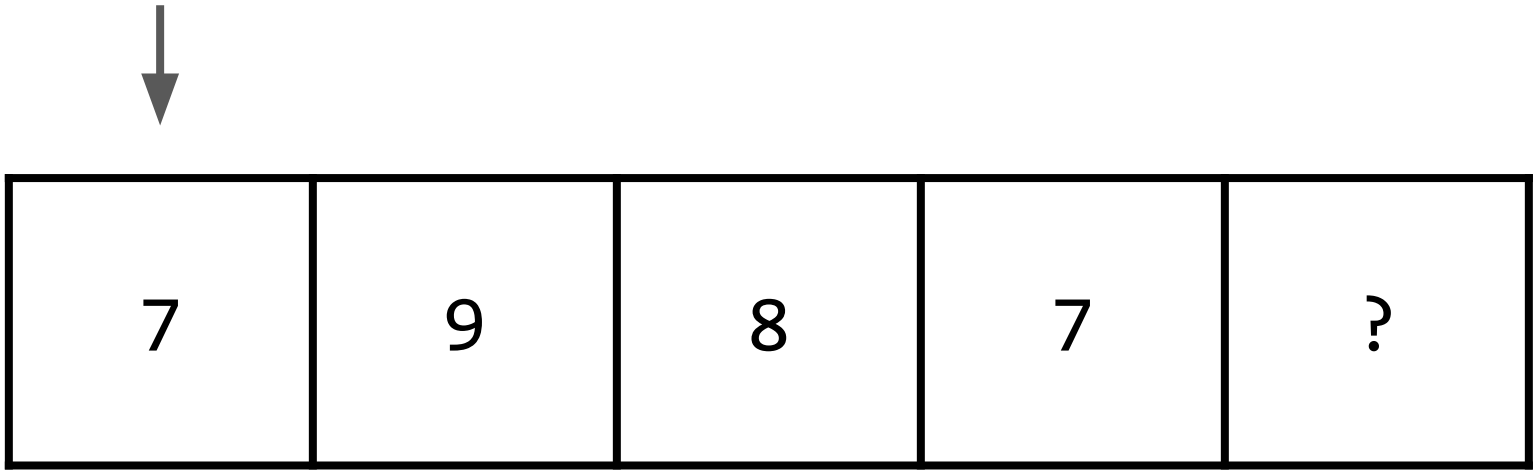


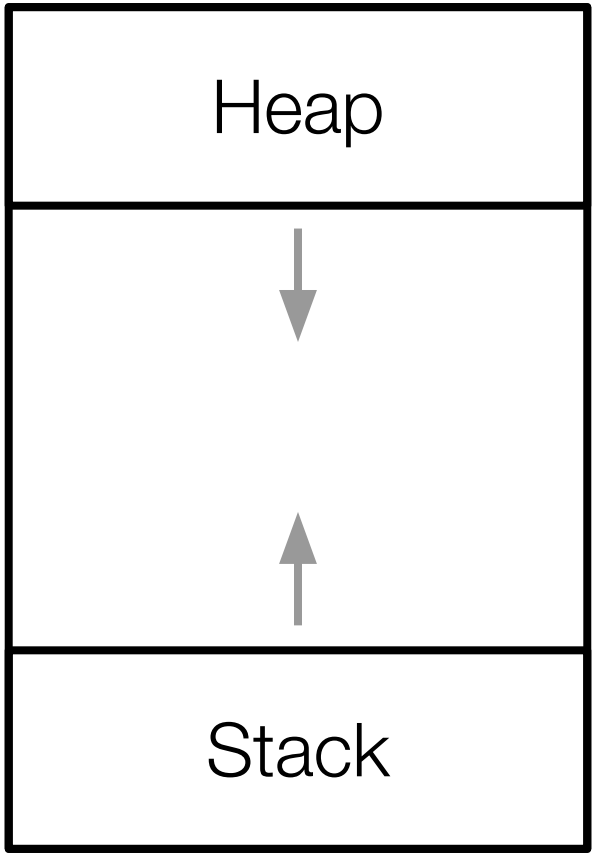
```
hours[2] = 8;
```



```
hours[2] = 8;
hours[3] = 7;
```

hours




Heap
Stack

Common memory errors

Failing to **free** every block of memory which we've **malloc**'d.

Failing to **fclose** every file we've **fopen**ed.

Using more memory than we've allocated.

Debugging Memory Exercise

Debug a program, **create.c**, that creates the file given as input at the command-line. For example,

```
./create test.c
```

will create a file, **test.c**. But our code has three memory errors! Can you find and fix them? Try running the below to check:

```
valgrind ./create test.c
```

Lab

Copying Header Bytes

We're given two file pointers, **input** and **output**. First, we need to copy the bytes, in the format **uint8_t**, from the input file to the output file.

What kind of "intermediary" structure should we use?

What functions might we use to pass data to this structure and then to our file?

Copying and Modifying Samples

Now we need to copy and modify samples, in the format **int16_t**, from the input file to the output file.

What kind of "intermediary" structure should we use?

What functions might we use to pass data to this structure and then to our file?

How should we modify the sample before writing it to the new file?

Tutorials

Office Hours

cs50.ly/attend


wssihaTSC50