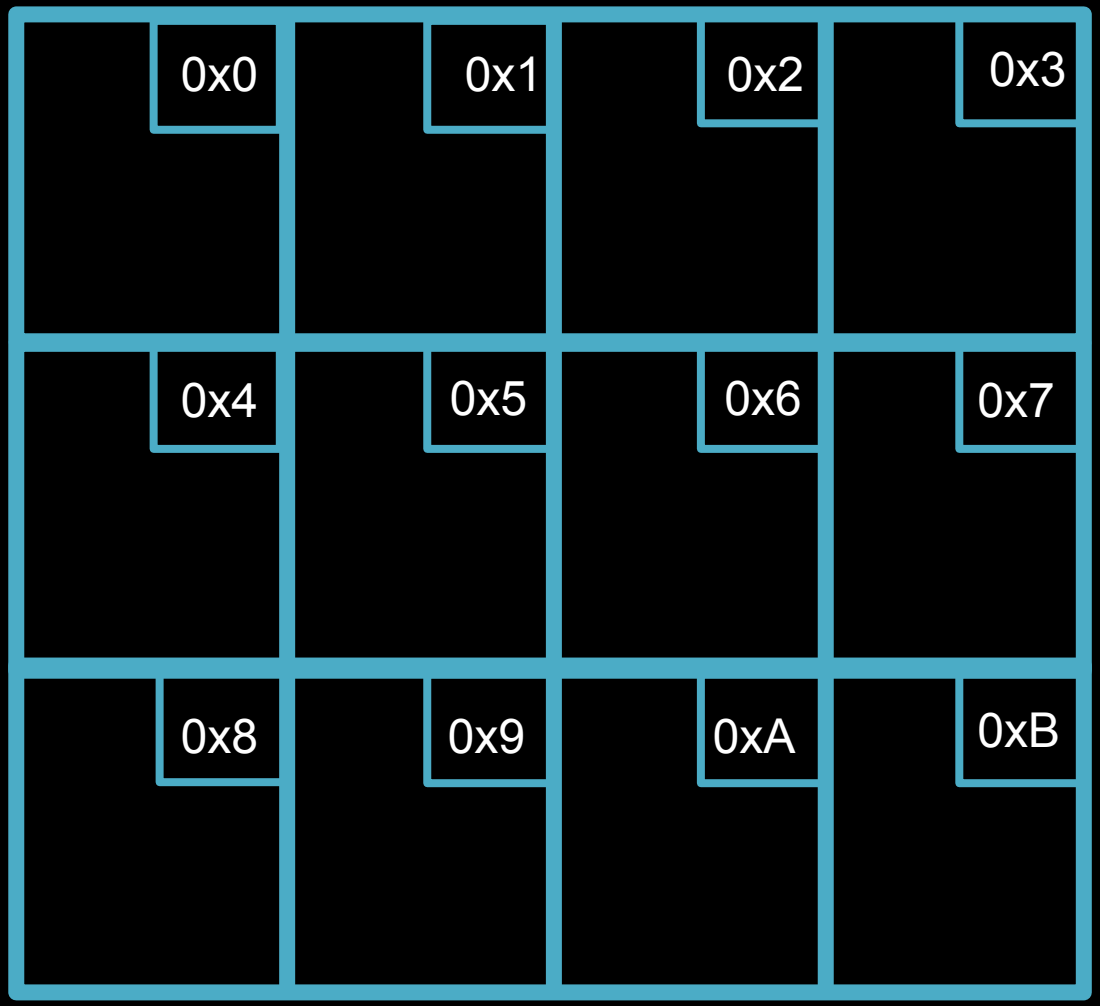
Pointers

Memory




MAN, I SUCK AT THIS GAME.
CAN YOU GIVE ME
A FEW POINTERS?
0x3A28213A
0x6339392C,
0x7363682E.
I HATE YOU.

Creating Pointers

Declaring pointers:
<type>* <variable name>

Examples:
int* x;
char* y;
float* z;

Referencing and Dereferencing

Referencing:
&<variable name>

Dereferencing:
***<pointer name>**

Under the hood...

```
int x = 5;

int* ptr = &x;

int copy = *ptr;
```

Variable	Address	Value
x	0x04	5
ptr	0x08	0x04
copy	0x0C	5

Track the values

	x	ptr
int x = 5;	5	
int* ptr = &x;	5	&x
*ptr = 35;	35	&x

Test Yourself

	a	b	c	pa	pb	pc
1.						
2.						
3.						
4.						
5.						
6.						
7.						

Answers

```
int a = 3, b = 4, c = 5;
int* pa = &a, *pb = &b, *pc = &c;
```

	a	b	c	pa	pb	pc
a = b * c;	20	4	5	&a	&b	&c
a *= c;	100	4	5	&a	&b	&c
b = *pa;	100	100	5	&a	&b	&c
pc = pa;	100	100	5	&a	&b	&a
*pb = b * c;	100	500	5	&a	&b	&a
c = (*pa) * (*pc);	100	500	10000	&a	&b	&a
*pc = a * (*pb);	50000	500	10000	&a	&b	&a

Pointer Arithmetic

Adding/subtracting `n` adjusts the pointer by

`n * sizeof(<type of the pointer>)` bytes

	x	y
int x = 5;	5	
int* y = &x;	5	0x04
y += 1;	5	0x08

What will print?

```
int main(void)
{
    char* str = "happy cat";
    int counter = 0;

    for (char* ptr = str; *ptr != '\0'; ptr++)
    {
        counter++;
    }

    printf("%d\n", counter);
}
```

Pointers and Arrays

```
int array[3];

*array = 1;
*(array + 1) = 2;
*(array + 2) = 3;
```