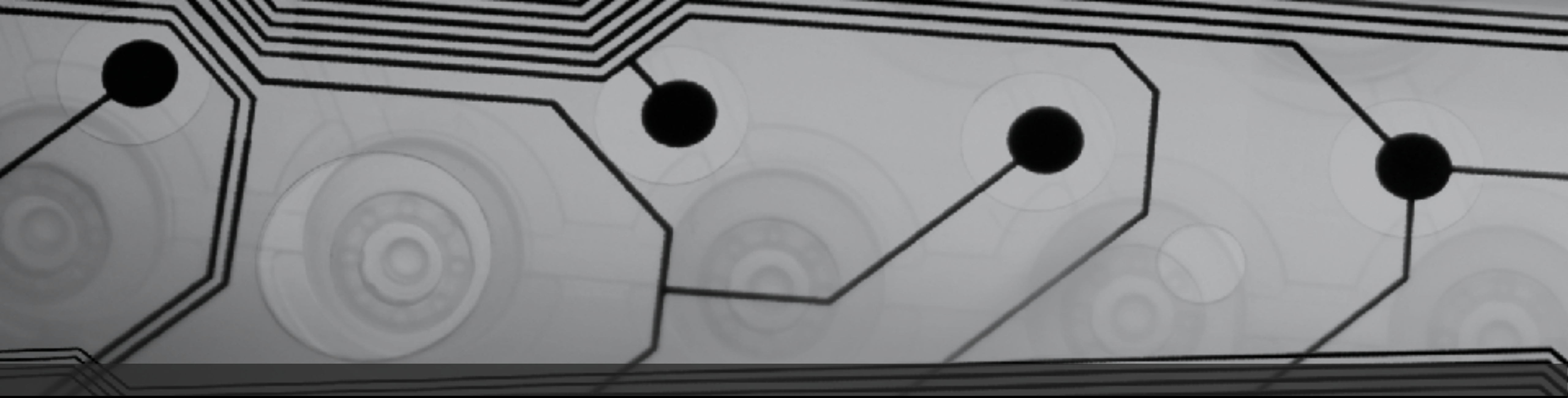

CMSC 131

Fall 2018

More memory diagrams with Arrays

Example:

```
int[] x = new int[5];
x[2] = 7;
x = new int[5];
```

Copying Arrays

Does this make a copy?

```
int[] a = new int[1000];
…
int[] b = a;
```

How can we write code that copies an array?

1. Let’s write the code
2. Other ways (we won’t cover them):
   - `clone`
   - `System.arraycopy`
   - `Arrays.copyOf`

Resizing an Array

Once array is instantiated, can you change it’s size?

What can we do? Fake it.

Let’s code this up:

Suppose we have an array, arr.

- Create a temporary array that is bigger than arr
- Copy existing data from arr over to new array
- Add additional data
- Re-assign arr so that it refers to new array

Alternate Style

Usually preferred:

```
int[] x, y;
int a, b;
```

This also works:

```
int x[], y[], a, b;
```

But why would you ever use this?

Arrays of References

Suppose I have a class called Cat and I want to store a sequence of Cats.

Let’s draw the memory diagram for this:

```
Cat[] x;
x = new Cat[4];    // how many Cats have I
made?
x[0] = new Cat(“Fluffy”);
x[1] = new Cat(“Princess”);
x[2] = new Cat(“Spot”);
x[3] = new Cat(“Steve”);
```

Initializing an array When Constructed

```
char[] arr = {‘x’, ‘@’, ‘A’, ‘!’};

double[] values = {3.1, 62.79, 5.88, 6.1, 7.55};

Cat[] kitties = {new Cat(“Felix”), new Cat(“Tom”),
                 new Cat(“Sylvia”), new Cat(“Oscar”)};
```