

CSCI 132:

Basic Data Structures and Algorithms

Arrays

Reese Pearsall
Fall 2023

Announcements

Program 1 due **tonight** at 11:59 PM

Lab 4 due on **Thursday** at 11:59 PM

```
Roses are Red,  
Violets are Blue.
```

```
Unexpected '{' on line 32.
```




What do you need to dig a hole?






What do you need to dig a hole?






What do you need to dig a hole?

	Pros	Cons
		
		
		




What do you need to dig a hole?

	Pros	Cons
	<ul style="list-style-type: none">• Cheap• Precise• No Training• Availability	<ul style="list-style-type: none">• Slow• Labor
		
		




What do you need to dig a hole?

	Pros	Cons
	<ul style="list-style-type: none">• Cheap• Precise• No Training• Availability	<ul style="list-style-type: none">• Slow• Labor
	<ul style="list-style-type: none">• Fast• Labor	<ul style="list-style-type: none">• Expensive• Training
		

What do you need to dig a hole?

	Pros	Cons
	<ul style="list-style-type: none">• Cheap• Precise• No Training• Availability	<ul style="list-style-type: none">• Slow• Labor
	<ul style="list-style-type: none">• Fast• Labor	<ul style="list-style-type: none">• Expensive• Training
	<ul style="list-style-type: none">• Really good at digging	<ul style="list-style-type: none">• Takes up a lot of garage space

What do you need to dig a hole?




	Pros	Cons
	<ul style="list-style-type: none">• Cheap• Precise• No Training• Availability	<ul style="list-style-type: none">• Slow• Labor
	<ul style="list-style-type: none">• Fast• Labor	<ul style="list-style-type: none">• Expensive• Training
	<ul style="list-style-type: none">• Really good at digging	<ul style="list-style-type: none">• Takes up a lot of garage space

Best tool for the job?

Burying your pet goldfish



What do you need to dig a hole?




	Pros	Cons
	<ul style="list-style-type: none">• Cheap• Precise• No Training• Availability	<ul style="list-style-type: none">• Slow• Labor
	<ul style="list-style-type: none">• Fast• Labor	<ul style="list-style-type: none">• Expensive• Training
	<ul style="list-style-type: none">• Really good at digging	<ul style="list-style-type: none">• Takes up a lot of garage space

Best tool for the job?

Building Express tunnel to Bridger Bowl



What do you need to dig a hole?




	Pros	Cons
	<ul style="list-style-type: none">• Cheap• Precise• No Training• Availability	<ul style="list-style-type: none">• Slow• Labor
	<ul style="list-style-type: none">• Fast• Labor	<ul style="list-style-type: none">• Expensive• Training
	<ul style="list-style-type: none">• Really good at digging	<ul style="list-style-type: none">• Takes up a lot of garage space

Best tool for the job?

Creating the foundation for a house



What do you need to dig a hole?

	Pros	Cons
	<ul style="list-style-type: none">• Cheap• Precise• No Training• Availability	<ul style="list-style-type: none">• Slow• Labor
	<ul style="list-style-type: none">• Fast• Labor	<ul style="list-style-type: none">• Expensive• Training
	<ul style="list-style-type: none">• Really good at digging	<ul style="list-style-type: none">• Takes up a lot of garage space

Best tool for the job?

Digging a Well for water



What do you need to dig a hole?



Best tool for the job?

Digging a Well for water



ensive
ning

s up a lot of
ge space

What do you need to dig a hole?



We can't use the best tool for the job unless we know that tool exists!






Best tool for the job?

Digging a Well for water



What do you need to dig a hole?

	Pros	Cons
	<ul style="list-style-type: none">• Cheap• Precise• No Training• Availability	<ul style="list-style-type: none">• Slow• Labor
	<ul style="list-style-type: none">• Fast• Labor	<ul style="list-style-type: none">• Expensive• Training
	<ul style="list-style-type: none">• Really good at digging	<ul style="list-style-type: none">• Takes up a lot of garage space

Best tool for the job?

Creating the foundation for a house



What do you need to dig a hole?

Pros

Cons



Best tool for the job?

Creating the foundation for a house



ve

a lot of
ace



What do you need to dig a hole?

Pros

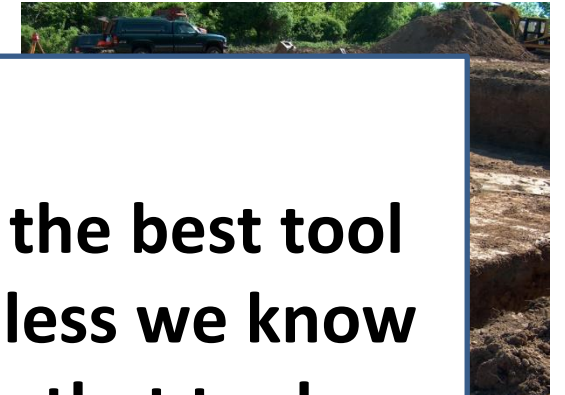
Cons



- Slow
- Labor

Best tool for the job?

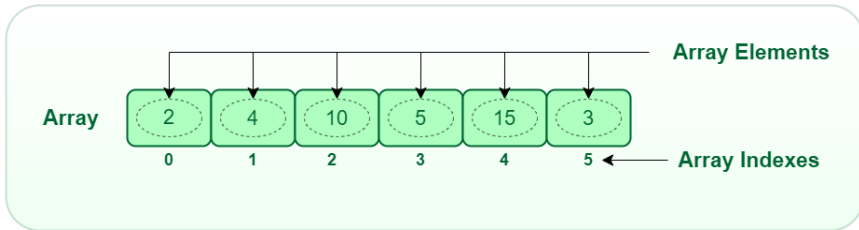
Creating the foundation for a house



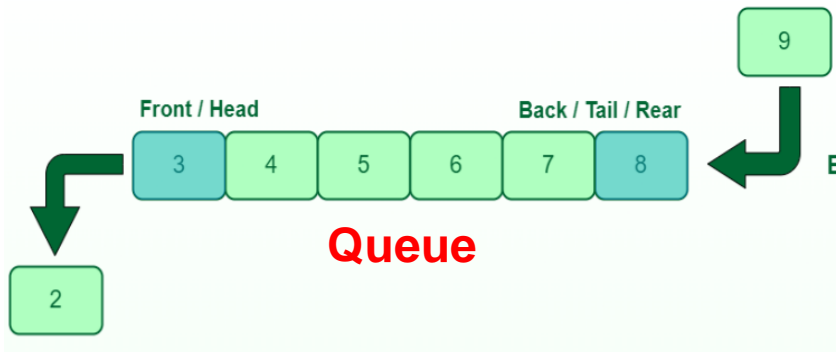
We can't use the best tool for the job unless we know how to use that tool

garage space

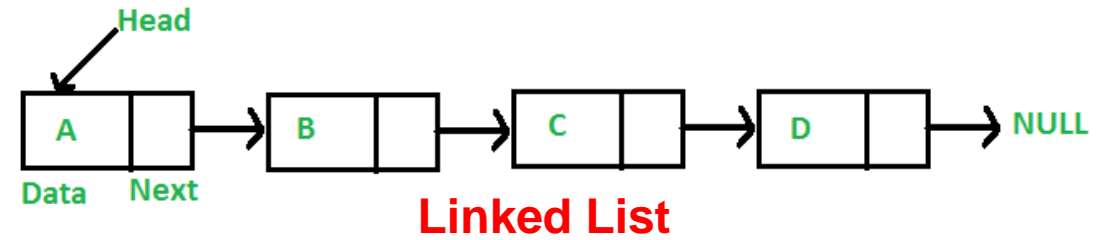
A **data structure** is a mechanism for storing and organizing data



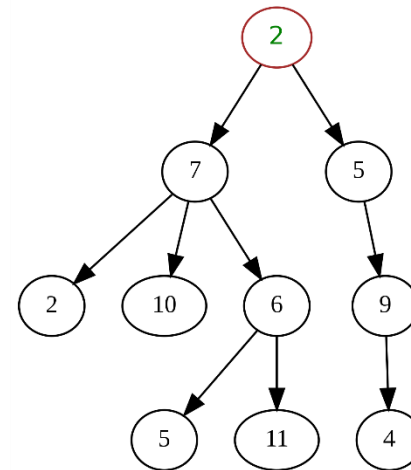
Arrays



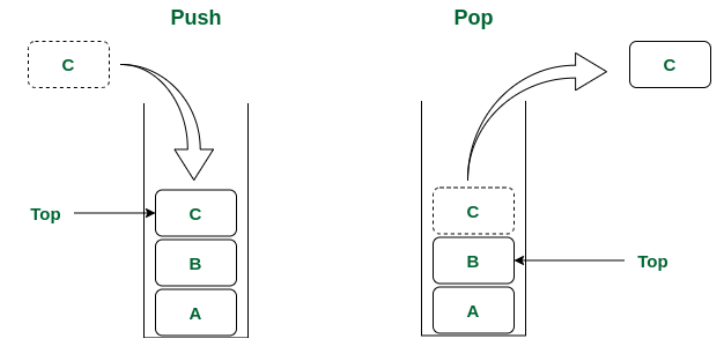
Queue



Linked List



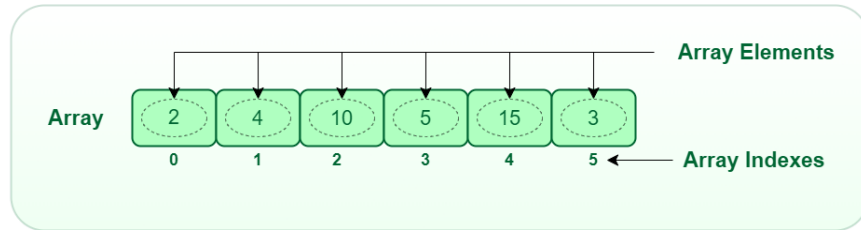
Trees



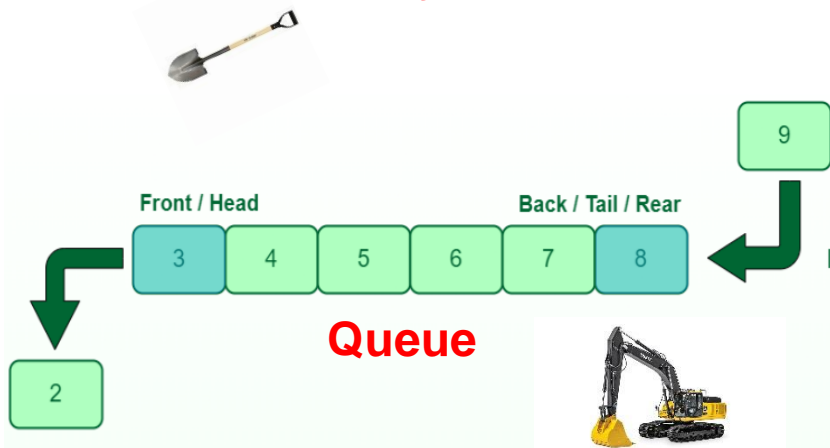
Stack

A **data structure** is a mechanism for storing and organizing data

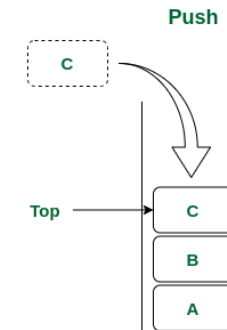
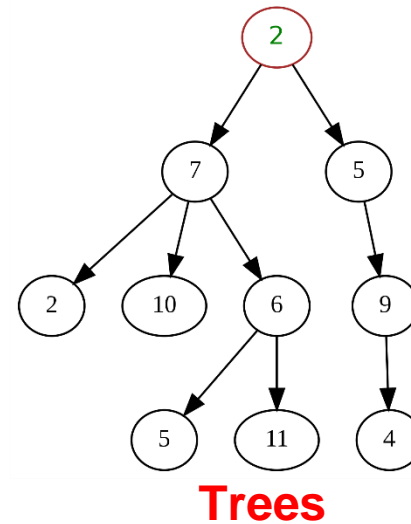
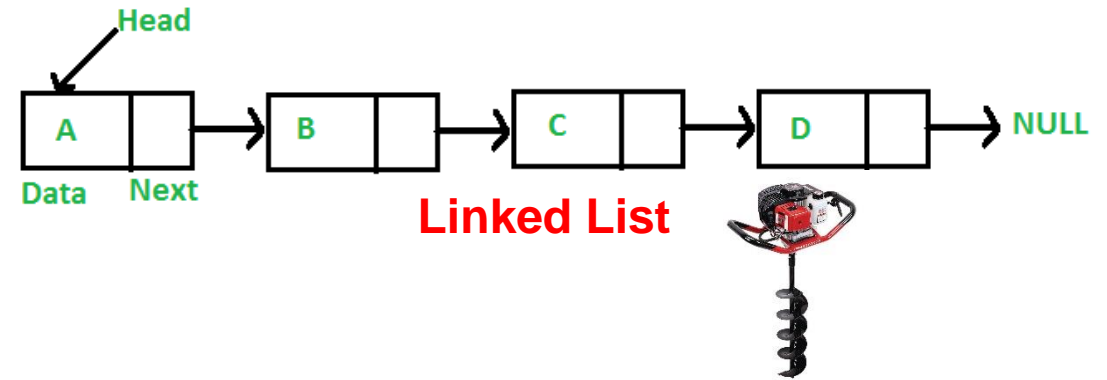
- We have structured ways of *accessing* and *managing* data



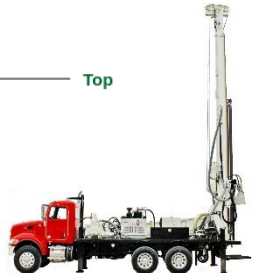
Arrays



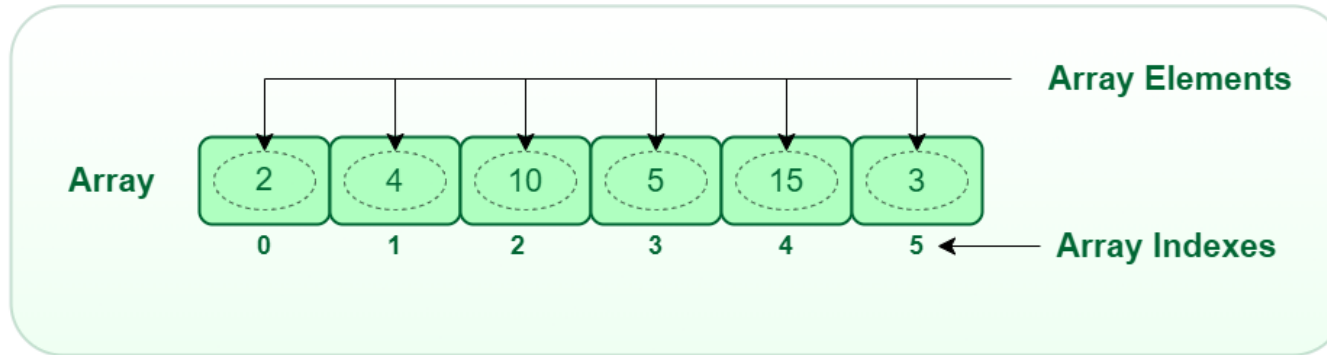
There are many types of data structure, and each data structure has its pros and cons



Stack



An **array** is a data structure that can hold multiple, similar values



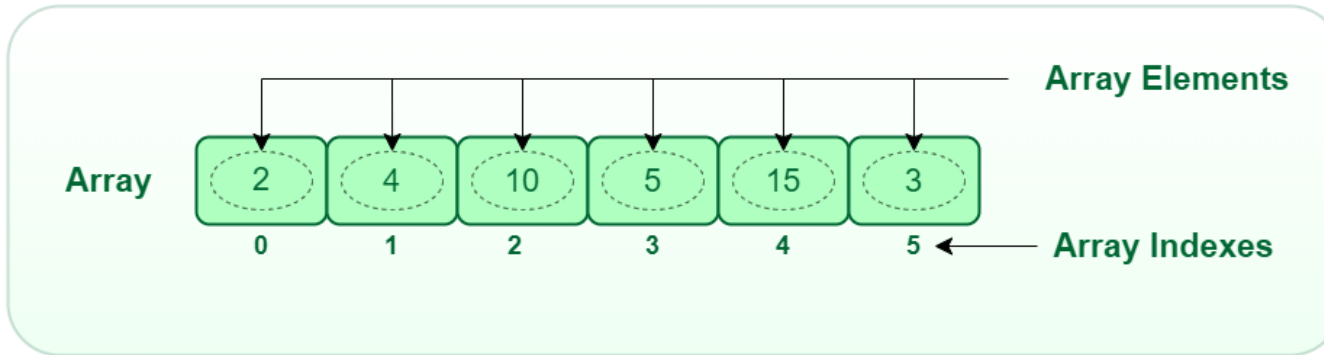
```
String[] cars = {"Volvo", "BMW", "Ford", "Mazda"};
```

```
int[] myNum = {10, 20, 30, 40};
```

Pros

- Holds multiple pieces of information
- Information is ordered (by index)
- Can easily change what is stored in each slot
- Can store duplicate data
- Easy to iterate through

An **array** is a data structure that can hold multiple, similar values



```
String[] cars = {"Volvo", "BMW", "Ford", "Mazda"};
```

```
int[] myNum = {10, 20, 30, 40};
```

Pros

- Holds multiple pieces of information
- Information is ordered (by index)
- Can easily change what is stored in each slot
- Can store duplicate data
- Easy to iterate through

Cons

- Can't change the length
- Can only store one data type

Array Limitations

Cons

- **Can't change the length**
- Can only store one data type

What can we do about this?

```
int[] myArray = {1, 2, 3};  
System.out.println(Arrays.toString(myArray));
```

What if we wanted to add 4 to the array?

Array Limitations

Cons

- **Can't change the length** *What can we do about this?*
- Can only store one data type

```
int[] myArray = {1, 2, 3};  
System.out.println(Arrays.toString(myArray));
```

```
int[] newArray = new int[myArray.length + 1];           // Create a new array that is one spot bigger  
for(int i = 0; i < myArray.length; i++) {  
    newArray[i] = myArray[i];                             // Fill new array with contents of old array  
}
```

Array Limitations

Cons

- **Can't change the length** *What can we do about this?*
- Can only store one data type

```
int[] myArray = {1, 2, 3};  
System.out.println(Arrays.toString(myArray));
```

```
int[] newArray = new int[myArray.length + 1];           // Create a new array that is one spot bigger  
for(int i = 0; i < myArray.length; i++) {               // Fill new array with contents of old array  
    newArray[i] = myArray[i];  
}
```

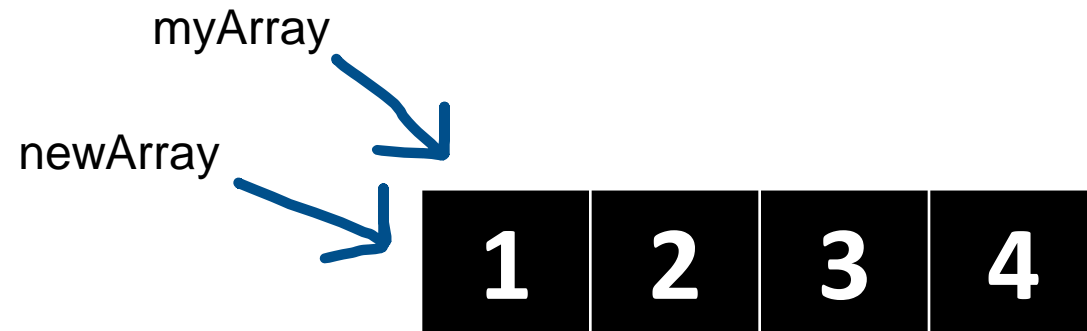
```
int new_value = 4;  
newArray[myArray.length] = new_value;                   // add new value to array  
myArray = newArray;                                       // Update reference variable
```


Array Limitations

```
int[] myArray = {1, 2, 3};  
System.out.println(Arrays.toString(myArray));  
  
int[] newArray = new int[myArray.length + 1];  
for(int i = 0; i < myArray.length; i++) {  
    newArray[i] = myArray[i];  
}  
  
int new_value = 4;  
newArray[myArray.length] = new_value;  
myArray = newArray;
```



We updated our reference variable (`myArray`) to point to our new array with the new element



Array Limitations

```
int[] myArray = {1, 2, 3};  
System.out.println(Arrays.toString(myArray));  
  
int[] newArray = new int[myArray.length + 1];  
for(int i = 0; i < myArray.length; i++) {  
    newArray[i] = myArray[i];  
}  
  
int new_value = 4;  
newArray[myArray.length] = new_value;  
myArray = newArray;
```



What happens to this array?
This is an unused object

We updated our reference variable
(myArray) to point to our new array with the
new element



Array Limitations

```
int[] myArray = {1, 2, 3};
System.out.println(Arrays.toString(myArray));

int[] newArray = new int[myArray.length + 1];
for(int i = 0; i < myArray.length; i++) {
    newArray[i] = myArray[i];
}

int new_value = 4;
newArray[myArray.length] = new_value;
myArray = newArray;
```

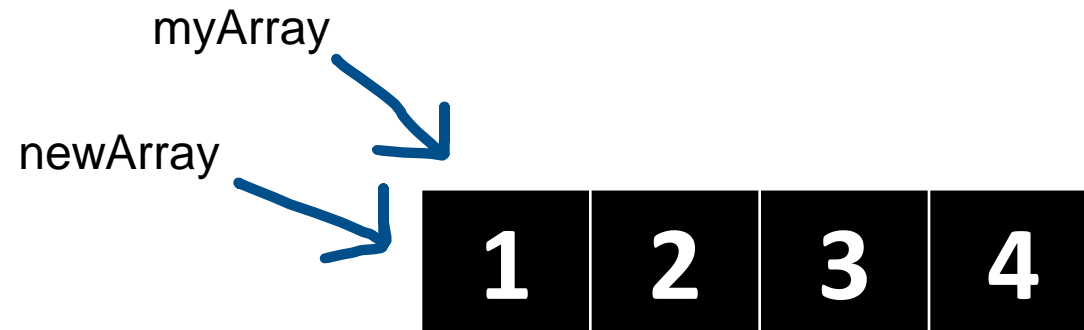


What happens to this array?
This is an unused object

We updated our reference variable
(myArray) to point to our new array with the
new element

Java has a mechanism called **Garbage
Collection**, with deletes unused object to
free up memory

(this runs automatically!)



Array Limitations

```
int[] myArray = {1, 2, 3};
System.out.println(Arrays.toString(myArray));

int[] newArray = new int[myArray.length + 1];
for(int i = 0; i < myArray.length; i++) {
    newArray[i] = myArray[i];
}

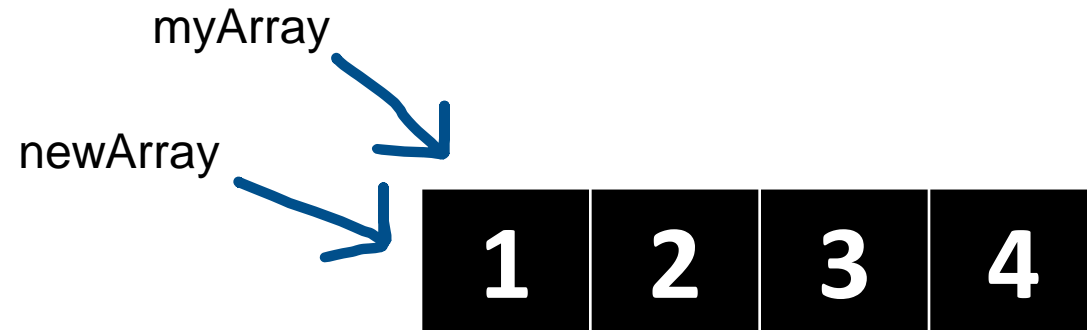
int new_value = 4;
newArray[myArray.length] = new_value;
myArray = newArray;
```



We updated our reference variable (`myArray`) to point to our new array with the new element

Java has a mechanism called **Garbage Collection**, with deletes unused object to free up memory

(this runs automatically!)



Java sees that we have an used/unreferenced object, so it will delete it!

Array Limitations

Cons

- **Can't change the length**

Solution

Create new array, copy everything over
(this can be expensive ☹)

```
int[] myArray = {1, 2, 3};  
System.out.println(Arrays.toString(myArray));
```

```
int[] newArray = new int[myArray.length + 1];  
for(int i = 0; i < myArray.length; i++) {  
    newArray[i] = myArray[i];  
}
```

```
int new_value = 4;  
newArray[myArray.length] = new_value;  
myArray = newArray;
```

- **Can only store one data type**

Solution

Store an object, use two separate arrays, use a different data structure

We are going to write our own dynamic array data structure

Users should be able to:

1. Print the array
2. Add a new element to the array
3. Get an element at a particular index
4. Find the index of a particular element