Arrays

Array Operation Idiom

Chapter 11, Examples 11-1, 11-2, 11-3, 11-4, 11-5, 11-6, 11-7, 11-8
Today...

- We will learn some common ways **arrays** are used
Today...

- We will learn some common ways arrays are used.

  - **Examples** of these operations include...
    - *initializing* elements in array
    - *drawing* something using elements in array
    - *adding* all elements in array
    - *calculating* the average of elements in array
    - *finding* the largest element in array
    - *finding* a particular element
      (e.g., finding element(s) greater than 10)
Since ...

these examples all use the *same pattern*, let’s first see what that **pattern** looks like and then look at each example.
Since ...

css: 105 | Arrays

these examples all use the *same pattern*, let’s first see what that pattern looks like and then look at each example.

Pattern in programming is referred to as *programming idiom*.
Since...

these examples all use the *same pattern*, let’s first see what that **pattern** looks like and then look at each example.

From now on, we will also refer to this as **array operation idiom**.
Since ... these examples all use the *same pattern*, let’s first see what that pattern looks like and then look at each example.

From now on, we will also refer to this as the array operation idiom. So what does it look like?
Array Operation Idiom

Part 1. Loop through an entire array

Part 2. During each iteration, apply a certain operation to elements

```javascript
for (let i = 0; i < array.length; i++) {
    // apply certain operation to elements
}
```
Array Operation Idiom

Part 1. Loop through an entire array

Part 2. During each iteration, apply a certain operation to elements

```javascript
for (let i = 0; i < array.length; i++) {
    // apply certain operation to elements
}
```
Array Operation Idiom

Part 1. Loop through an entire array

Part 2. During each iteration, apply a certain operation to elements

```javascript
for (let i = 0; i < array.length; i++) {
    // apply certain operation to elements
}
```
Now ...

- Let’s look at each example in detail
Array Operation Idiom

- Examples of these operations include...
  - initializing elements in array
  - drawing using elements in array
  - adding all elements in array
  - calculating the average of elements in array
  - finding the largest element in array
  - searching for a particular element
Array Operation Idiom

- Examples of these operations include...
  - initializing elements in array
    - drawing using elements in array
    - adding all elements in array
    - calculating the average of elements in array
    - finding the largest element in array
    - searching for a particular element
Remember what initialization was?

(HINT: variable initialization)
Remember what initialization was?

Initialization is **setting initial value**

(*HINT*: variable initialization)
let x = []; // array

function setup() {
    // initialize 6 racer starting positions
    for (let i = 0; i < 6; i++) {
        x[i] = 0;
    }
}

function draw() {
    ...  
    for (let i = 0; i < x.length; i++) {
        // calculate the y position
        let y = 40 * (i + 1);
        racerDraw(x[i], y, 30, i + 1);
        // update the racer's position
        x[i] += random(0, 3);
    }
}
Initializing Array - More examples

- Example 1:
  - “I want to create **multiple** circles of the **same diameter**”

```javascript
let circle_diam = [];

// initialize 15 circle diameter sizes
for (let i = 0; i < 15; i++) {
  circle_diam[i] = 25;
}

// draw circle
...
```
Initializing Array - More examples

- Example 2:
  - “I want to create **multiple** trees that have the **same height**”

```javascript
let tree_height = [];

// initialize 11 tree height sizes
for (let i = 0; i < 11; i++) {
  tree_height[i] = 60;
}

// draw trees
...
```
Notice that ...

```javascript
// initialize array elements to zero
for (let i = 0; i < arr.length; i++) {
  arr[i] = 0;
}
```

The value you assign may be 0, 25, 60, ...
Notice that ...

// initialize array elements to zero
for (let i = 0; i < arr.length; i++) {
    arr[i] = 0;
}

The value you assign may be 0, 25, 60, true, false, ...

// initialize array elements to true
for (let i = 0; i < arr.length; i++) {
    arr[i] = true;
}
Notice that ...

```javascript
// initialize array elements to zero
for (let i = 0; i < arr.length; i++) {
    arr[i] = 0;
}
```

The value you assign may be 0, 33, 11, true, false ...

```javascript
// initializing elements with same value
```

```javascript
// initialize array elements to true
for (let i = 0; i < arr.length; i++) {
    arr[i] = true;
}
```
Notice that ...

```javascript
// initialize array elements to zero
for (let i = 0; i < arr.length; i++) {
    arr[i] = 0;
}
```

You can also initialize with `random values`

```javascript
// initialize array elements to random value
for (let i = 0; i < arr.length; i++) {
    arr[i] = random(0, 100);
}
```
Array Operation Idiom

- Examples of these operations include...

  - **initializing elements in array**
  - drawing using elements in array
  - adding all elements in array
  - calculating the average of elements in array
  - finding the largest element in array
  - searching for a particular element
Array Operation Idiom

- Examples of these operations include...
  - initializing elements in array
  - **drawing using elements in array**
  - adding all elements in array
  - calculating the average of elements in array
  - finding the largest element in array
  - searching for a particular element
Drawing - Examples

- Draw circles using elements in an array

```javascript
// draw using array elements
for (let i = 0; i < circle_diam.length; i++) {
    ellipse(25 * (i + 1), 50, circle_diam[i], circle_diam[i]);
}
```
Drawing - More examples

- Draw a bar graph using elements in an array

```javascript
// visualize each element as a thin vertical bar
for (let i = 0; i < arr.length; i++) {
  line(i, height, i, height - arr[i]);
}
```
Array Operation Idiom

- Examples of these operations include...
  - initializing elements in array
  - **drawing using elements in array**
  - adding all elements in array
  - calculating the average of elements in array
  - finding the largest element in array
  - searching for a particular element
Array Operation Idiom

- Examples of these operations include...
  - initializing elements in array
  - drawing using elements in array
  - adding all elements in array
  - calculating the average of elements in array
  - finding the largest element in array
  - searching for a particular element
Sum all Elements

- Compute the sum (i.e. total) of all elements in an array by adding them one by one to an *accumulator* variable.

- Example array:

```javascript
arr = [5, 3, 10, 15, 7];

// compute the sum of all elements
let sum = 0;
for (let i = 0; i < arr.length; i++) {
    sum += arr[i];
}
print("sum:", sum);
```

https://editor.p5js.org/sanghosuh/sketches/RtjbAi3V-
Array Operation Idiom

- Examples of these operations include...
  - initializing elements in array
  - drawing using elements in array
  - **adding all elements in array**
  - calculating the average of elements in array
  - finding the largest element in array
  - searching for a particular element
Array Operation Idiom

- Examples of these operations include...
  - initializing elements in array
  - drawing using elements in array
  - adding all elements in array
  - **calculating the average of elements in array**
  - finding the largest element in array
  - searching for a particular element
Calculating Average

- Sum all elements in an array and divide total by the length of the array to compute the average element size.

```javascript
// calculate the average
let sum = 0;
for (let i = 0; i < arr.length; i++) {
  sum += arr[i];
}
let average = sum / arr.length;
print("average:", average);
```

https://editor.p5js.org/sanghosuh/sketches/XuWLtabdX
Array Operation Idiom

- Examples of these operations include...
  - initializing elements in array
  - drawing using elements in array
  - adding all elements in array
  - **calculating the average of elements in array**
  - finding the largest element in array
  - searching for a particular element
Array Operation Idiom

- Examples of these operations include...
  - initializing elements in array
  - drawing using elements in array
  - adding all elements in array
  - calculating the average of elements in array
  - finding the largest element in array
  - searching for a particular element
Finding Largest Element Value

- Search the array to find the largest element value
  - Start by guessing that the largest value is the first element
  - check all other elements one-by-one to see if any are larger
  - if larger value is found, it becomes new largest value found so far

```javascript
// find the largest element value
// (assuming arr has at least length 1)
let largest = arr[0];
for (let i = 1; i < arr.length; i++) {
  if (arr[i] > largest) {
    largest = arr[i];
  }
}
print("largest:", largest);
```

https://editor.p5js.org/sanghosuh/sketches/zDI3DbpcR
Finding Largest Element Index

- Search the array to find the index with largest element value
  - Start by guessing that index 0 has element with largest value
  - check all other elements one-by-one to see if any are larger
  - if larger one is found, use it’s index as the largest element index

```javascript
// find the index of the largest element
// (assuming arr has at least 1 element)
let indexOfLargest = 0;
for (let i = 1; i < arr.length; i++) {
  if (arr[i] > arr[indexOfLargest]) {
    indexOfLargest = i;
  }
}
print("index of largest:", indexOfLargest);
print("largest:", arr[indexOfLargest]);
```

https://editor.p5js.org/sanghosuh.sketches/IcwF5dlLl
Array Operation Idiom

- Examples of these operations include...
  - initializing elements in array
  - drawing using elements in array
  - adding all elements in array
  - calculating the average of elements in array
  - finding the largest element in array
  - searching for a particular element
Array Operation Idiom

- Examples of these operations include...
  - initializing elements in array
  - drawing using elements in array
  - adding all elements in array
  - calculating the average of elements in array
  - finding the largest element in array
  - searching for a particular element
Search for particular element

- You can search for elements that satisfy a particular condition
e.g., “How many elements in the array are greater than 10?”

```javascript
let arr = [8, 13, 9, 35, 3, 5, 20];
let count = 0;

for (let i = 1; i < arr.length; i++) {
    if (arr[i] > 10) {
        count++;
    }
}

print("# of elements > 10:", count);
```

https://editor.p5js.org/sanghosuh/sketches/NzKhO0JG-
Consider three arrays storing information about baseball players:

```javascript
let player = [4, 99, 66, 11]; // player number
let hits = [101, 110, 79, 140]; // hits
```

How to “search” the number of “hits” by a player with number 66?

```javascript
for (let i = 0; i < player.length; i++) {
  if (player[i] === 66) {
    print(hits[i]);
  }
}
```
Recap

- We looked at some common ways arrays are used
- We learned the pattern underlying these operations

```javascript
for (let i = 0; i < array.length; i++) {
  // apply certain operation to elements
}
```
Recap

- We looked at some **common ways** **arrays** are used
- We learned the pattern underlying these operations

```javascript
for (let i = 0; i < array.length; i++) {
    // apply certain operation to elements
}
```

- We saw that we can perform **meaningful actions** (e.g., finding the largest element) using the array operation idiom
Imagine

- You are writing a program that needs to **constantly** ...
  - calculate the total or average score
  - draw multiple shapes with the same or different sizes
  - find an object with the highest (point/score/speed/...)
  - find an object with certain conditions
Imagine

- You are writing a program that needs to **constantly** ...
  - calculate the total or average score
  - draw multiple shapes with the same or different sizes
  - find an object with the highest (point/score/speed/...)
  - find an object with certain conditions

It would make things more convenient if we can make them into **functions**, as it improves **readability & reusability**
Using Arrays with User-Defined Functions
Using Arrays with User-Defined Functions

Functions can have arrays as a parameter

Functions can return an array
Functions Can Use an Array as a Parameter

// returns largest element in array a
// (assuming a has at least 1 element)
function largestValue(a) {
  let largest = a[0];
  for (let i = 1; i < a.length; i++) {
    if (a[i] > largest) {
      largest = a[i];
    }
  }
  return largest;
}

https://editor.p5js.org/sanghosuh/sketches/yCtt6oy5J
Functions Can Return an Array

```javascript
// returns an array of size n with all
// elements set to value
function createArray(n, value) {
    let a = [];
    for (let i = 0; i < n; i++) {
        a[i] = value;
    }
    return a;
}
```

- **Usage**
  ```javascript
  // create length 100 array with all elements 0
  let arr = createArray(100, 0);
  ```

- **Variation:** createRandomArray with all element values initialized to random number in between low to high

This kind of function is called a "factory" since it returns a new array.

[Link to code](https://editor.p5js.org/sanghosuh.sketches/cbXNiXR6W)
let treeX = [166, 90, 77, 30, 110, 131, 55];
let treeLeafColour = [];

function setup() {
    ...
    // initialize to random tint of green
    for (let i = 0; i < treeX.length; i++) {
        treeLeafColour[i] = random(30, 130);
    }
}

function draw() {
    ...
    // draw trees for forest
    for (let i = 0; i < treeX.length; i++) {
        tree(treeX[i], 95, treeLeafColour[i]);
    }
}

https://editor.p5js.org/sanghosuh/sketches/-cSOWPc33