Announcements

- HW 4 grades posted on blackboard
  - Please contact your TA grader first for grading issues
    - Alex Sheppard: last name A – K
    - David Wilkie: last name L – R
    - Deepak Somashekhara: last name S – Z

- Reminder: HW 5 due on Wednesday

- 2-min presentations by:
  - Nishant Sachdev
  - Garrett Brown

Referencing example

```javascript
var list = new Array(5, 3, 8, 2);
var newList = list;
list[0]++;
alert(newList[0]);
```

- What will the alert box display?
**Another example**

```javascript
var a = new Array(10, 20, 30);
var b = new Array(50, 60, 70);
b = a;
```

- **What does this do?**
  - a and b now reference the same array
  - a[2] is the same as b[2]
  - No reference to the 50, 60, 70 array!
    - Garbage!
    - Some languages (Java) clean up

---

**Example: make change**

- **Problem:** given integer in 0, ..., 99, make change
  - Using quarters, dime, nickel, pennies
  - e.g.: 92 : 3 Q + 1 D + 1 N + 2 P
- **How?**
  - **Start with Q:** 92 – 25 = 67 (Q)
    - 67 – 25 = 42 (Q)
    - 42 – 25 = 17 (Q)
    - 17 – 25 = -8 X
    - 17 – 10 = 7 (D)
    - 7 – 10 = -3 X
  - **Then use D:**
    - 7 – 5 = 2 (N)
    - 2 – 5 = -3 X
  - **Nickels:**
    - 7 – 5 = 2 (N)
    - 2 – 5 = -3 X
  - **Pennies:**
    - 2 – 1 = 1 (P)
    - 1 – 1 = 0 (P)
    - done
Let’s write the function

- Function should return an array containing # of each coin type
  - ChangeMaker.html

- Version 1: change1
  - ⚫: specific to the coin set Q, D, N, P
  - ⚫: repeats code!
  - ⚫: magic numbers!

- Version 2: change2
  - ☀: No magic numbers
  - ☀: Reused code
  - ☀: Works for any sorted coin set
    - (75, 38, 12, 3, 2, 1)
    - Solution may not be unique

Recursion example: sum

```javascript
function sumList(list, start)
{
    // return sum from list[start] to list end

    if (start >= list.length)
        return 0;

    return (list[start] + sumList(list, start+1));
}
```
Recursion example: max

```
function maxL(list, start)
{
    // return largest in list[start] to list end
    if (start == list.length - 1)
        return list[start];
    return Math.max(list[start], maxL(list, start+1));
}
```

Recursion example: reverse string

```
function revStr(s)
{
    if (s.length <= 1)
        return s;
    return revStr(s.substring(1)) + s.charAt(0));
}
```

- Write a function that reverses a string
  - e.g., `revStr("harpo")` -> returns “oprah”

- Note that this function can also be written without recursion (using only loops)
Defining classes in Javascript

```javascript
function Employee (name) {
    this.name = name;
    this.employer = "UNC";
    this.getInfo = getEmployeeInfo;
}

function getEmployeeInfo() {
    return (this.name + " @ " + this.employer);
}

var e1 = new Employee("Jasleen");
e1.employer = "Microsoft";
alert(e1.getInfo());

FYI – will not test on this
```