**Whiteboarding Questions 4**

Everyone should pair up to take turns whiteboarding the following problems and **identifying the efficiency of each algorithm in terms of Big O Notation**. (Take turns on doing each one.)

1. Write a function that accepts an array as a parameter and returns true or false depending on whether there are any duplicate values. Use the \*nested loops approach\* as described in the slides.

2. Write a function that accepts an array as a parameter and returns true or false depending on whether there are any duplicate values. Use the \*hash approach\* as described in the slides.

3. Write a method that accepts a string as a parameter and returns an array containing each character as a separate value, but the character must also be capitalized.

4. Write a method that accepts an array of numbers and returns the sum of all odd numbers.

5. (If you didn’t get to it last week) Write a method that accepts two arrays, and returns a new array that does a “riffle shuffle” between them. That is, if the first array is [1, 3, 5, 7, 9], and the second array is [2, 4, 6, 8, 10], the returned array should be [1, 2, 3, 4, 5, 6, 7, 8, 9, 10].

BONUS: You can use your computer for this last section. Read about and implement the following sorting algorithms that sort an array of numbers. Be sure to identify the Big O efficiency of each algorithm!

\* Bubble sort

\* Insertion sort

\* Selection sort

\* Merge sort

\* Quick sort

\* Heap sort