Q1 Instructions

0 Points

To receive full credit on this quiz, you must score at least 50%.

The Github repo for Lecture 9 is at:
https://github.com/ucsd-cse12-w21/ucsd-cse12-w21.github.io/tree/master/lectures/lecture-09

Q2 Run-Time

1 Point

```
public static boolean isSorted1(int[] arr) {
  for(int i = 0; i < arr.length - 1; i += 1) {
   if(arr[i] > arr[i + 1]) {
    return false;
   }
  }
  return true;
 }

 public static boolean isSorted2(int[] arr) {
  for(int i = 0; i < arr.length; i += 1) {
   for(int j = i + 1; j < arr.length; j += 1) {
    if(arr[i] > arr[j]) {
     return false;
    }
   }
  }
  return true;
 }
```

Which of the following are true for the above methods assuming arr.length is very large and the array is already sorted (i.e. worst case)? Select all that apply:

- ☐ isSorted1() and isSorted2() both have linear run-times
- ☐ isSorted1() and isSorted2() both have parabolic run-times
- ☑ isSorted1() has a linear run-time and isSorted2() has a parabolic run-time
- ☐ isSorted1() has a parabolic run-time and isSorted2() has a linear run-time
- ☑ isSorted1() will always run faster than isSorted2()
- ☐ isSorted2() will always run faster than isSorted1()
- ☐ isSorted1() and isSorted2() run at the same speed

Q3 Counting Steps-Find 1

1 Point

```
boolean find( String[] theList, String toFind ) {
  for ( int i = 0;
      i < theList.length;
      i += 1 ) {
    if ( theList[i].equals( toFind )) {
      return true;
    }
  }
  return false;
}
```

Which of the following are true for the above find() method? Select all that apply:

☐ The worst case, best case, and average case have nearly the same steps (within a few steps)

☑ The best case is much better than the worst case

☑ The average case is better than the worst case, but not as good as the best case

☑ The worst cast has linear run-time based on the # of times evaluated

☐ The worst cast has parabolic run-time based on the # of times evaluated

Q4 Counting Steps-Find 2

1 Point

```
boolean find( String[] theList, String toFind ) {
   boolean found = false;
   for ( int i = 0;
        i < theList.length;
        i += 1 ) {
      if ( theList[i].equals( toFind )) {
         found = true;
      }
   }
   return found;
}
```

Which of the following are true for the above find() method? Select all that apply:

☑ The worst case, best case, and average case have nearly the same steps (within a few steps)

□ The best case is much better than the worst case

□ The average case is much better than the worst case, but not as good as the best case

☑ The worst cast has linear run-time based on the # of times evaluated

□ The worst cast has parabolic run-time based on the # of times evaluated