Quiz: Array Searching and Sorting

1. (15 points) Consider the following sorted array:

   ```java
double[] data = {1, 4, 6, 7, 9, 10, 14};
```

A. If using **binary search**, how many comparisons are needed (ie, how many elements in the array do we need to examine) to determine that 9 is in the array?

B. If using **sequential search**, how many comparisons (ie, how many elements in the array do we need to examine) are needed to determine that 9 is in the array?

C. If using **sequential search**, how many comparisons (ie, how many elements in the array do we need to examine) are needed to determine that 11 is not in the array?

2. (25 points) Fill in the blanks in the following code segment to implement sequential search of the given array. The method should return a boolean value indicating whether or not the parameter value was found in the array.

   ```java
   public static boolean seqSearch(int s){
       int[] a = {9, 12, 14, 3, 25};
       for( int i = __________; ________________; i++){
           if (______________ ){
               return ________________;
           }
       }
       return ________________ ;
   }
   ```
3. (10 points) The following code segment implements selection sort as described in the lecture video and textbook. Draw the array after each iteration of the loop (ie, at the point in the code indicated by the comment). Hint: You may not need all the spaces provided for a correct answer.

```java
int[] list = {5, 3, 8, -1};

for(int i = 0; i < list.length; i++) {
    //find minimum value
    int currentMin = list[0];
    int currentMinIndex = 0;

    for (int j = i+1; j < list.length; j++) {
        if (currentMin > list[j]) {
            currentMin = list[j];
            currentMinIndex = j;
        }
    }

    //swap min element into place
    if (currentMinIndex != i) {
        list[currentMinIndex] = list[i];
        list[i] = currentMin;
    }

    //**********draw array at this point **********
}
```