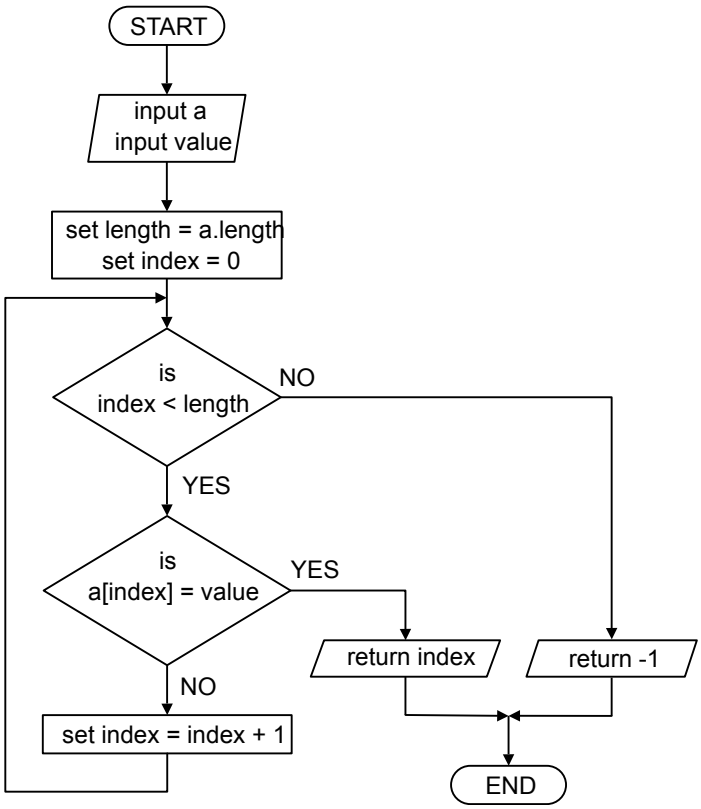
Array Assignment: `linearSearch`

Recall that when we studied flowcharts and pseudocode, we encountered linear search. We drew a flowchart that represented an algorithm that traversed a list looking for a specified value, returned the index of the value if it were found, and a value of `-1` if the value were not found. A solution for the flowchart is given to the right, and example pseudocode is given below.




```
FUNCTION
linearSearch( a, value )
BEGIN FUNCTION
   SET length TO LENGTH(a)
   SET index TO 0
   WHILE (index < length) DO
      IF (a[index] = value) THEN
         RETURN index
      END IF
      SET index TO index + 1
   END WHILE
   RETURN -1
END FUNCTION
```

The example pseudocode uses a `WHILE` loop; however, for this assignment you will be asked to use a `for` loop.

1. Create a Java class named `LinearSearch` organized into an appropriate package. Copy the code given at the end of this assignment, and run the code. The expected output is as follows:

```
printArray not implemented
printArrayReverse not implemented
printArray not implemented
printArrayReverse not implemented
The value 6 was not found in the array.
The value 8 was not found in the array.
The value 0 was not found in the array.
The value 2 was not found in the array.
The value 1 was not found in the array.
```

2. Write a method named `printArray` that takes an array of integers as a parameter, and uses a `for` loop to iterate through the array and print out each element of the array, all elements on the same line separated by a space character. After all the elements of the array have been printed, print a `newline` character (use `println`).

3. Write a method named `printArrayReverse` that takes an array of integers as a parameter, and uses a `for` loop to iterate through the array in reverse and print out each element of the array, all elements on the same line separated by a space character. After all the elements of the array have been printed, print a `newline` character.

4. Review our previous *linear search* algorithm and assignment (a solution is given above). Then write a method named `linearSearch` in Java that implements the linear search algorithm. The method should take two parameters – an array of integers, and an integer, named `value`, that it will search the array for. The method must use a `for` loop to traverse the array in search of `value`. The method

is to return the lowest index of the array that contains an element that is equal to `value`, or `-1` if the array does not contain an element equal to `value`.

5. Write another Java method, named `linearSearchReverse`, that implements the linear search algorithm. This method must conform to all the specifications in the previous question, except it must traverse the array in reverse order to find the highest index of the array that contains an element that is equal to `value`, or `-1` if the array does not contain an element equal to `value`.

Template Code for this assignment:

```
public class LinearSearch {
   public static void main(String[] args) {
      int[] a1 = { 2, 4, 6, 7, 10 };
      int[] a2 = new int[10];
      a2[8] = 999;

      printArray(a1);
      printArrayReverse(a1);
      printArray(a2);
      printArrayReverse(a2);

      findValue(a1, 6);
      findValue(a1, 8);
      findValueReverse(a2,0);
      findValueReverse(a1,2);
      findValueReverse(a1,1);
   }
   public static void findValue(int[] a, int value, boolean reverse) {
      int index;

      if(reverse) {
         index = linearSearchReverse(a, value);
      } else {
         index = linearSearch(a, value);
      }

      System.out.print("The value " + value + " was ");

      if(index == -1) {
         System.out.println("not found in the array.");
      } else {
         System.out.println("was found at index " + index);
      }
   }
   public static void findValue(int[] a, int value) {
      findValue(a, value, false);
   }
   public static void findValueReverse(int[] a, int value) {
      findValue(a, value, true);
   }


   // *** CODE CONTINUED ON NEXT PAGE
```

```
   public static void printArray(int[] a) {
      // *** REPLACE THIS LINE OF CODE WITH YOUR CODE ***
      System.out.println("printArray not implemented");
   }
   public static void printArrayReverse(int[] a) {
      // *** REPLACE THIS LINE OF CODE WITH YOUR CODE ***
      System.out.println("printArrayReverse not implemented");
   }
   public static int linearSearch(int[] a, int value) {
      // *** WRITE YOUR CODE FOR linearSearch HERE ***
      return -1;
   }
   public static int linearSearchReverse(int[] a, int value) {
      // *** WRITE YOUR CODE FOR linearSearchReverse HERE ***
      return -1;
   }
}
```