✗ What will be the output of the following? * 0/1

```
int sumOdd(int n) {
  if (n <= 0) return 0;
  if (n % 2 != 0) return n + sumOdd(n - 1);
  return sumOdd(n - 1);
}
System.out.println(sumOdd(9));
```

- a) 25
- b) 20
- c) 45 ✗
- d) 35

Correct answer

- a) 25

✓ What will be printed by the following function? * 1/1

```
String traverseString(String str) {
  if (str.isEmpty()) return str;
  return traverseString(str.substring(1)) + str.charAt(0);
}
System.out.println(traverseString("abcde"));
```

- a) edcba ✓
- b) abcde
- c) abcd
- d) aedcb

✓ What will be the output of the following? * 1/1

```
boolean countNumbers(int[] arr, int target, int n) {
  if (n == 0)
    return false;
  if (arr[n - 1] == target)
    return true;
  return countNumbers(arr, target, n - 1);
}
System.out.println(countNumbers(new int[]{5, 10, 15, 20}, 10, 4));
```

- a) true ✓
- b) false
- c) 0
- d) -1

✓ What will this snippet print? * 1/1

```
int[] arr = {2, 4, 6, 8};
for (int i = 0; i < arr.length; i++) {
  if (i % 2 == 1) arr[i] = arr[i] / 2;
}
System.out.println(Arrays.toString(arr));
```

- a) [2, 4, 6, 8]
- b) [2, 2, 6, 4] ✓
- c) [2, 4, 3, 8]
- d) [2, 2, 6, 4]

✓ What will be the output of the following? * 1/1

```
int countArray(int[] arr, int n) {
  if (n == 0)
    return 0;
  int sum = arr[n - 1];
  return sum + countArray(arr, n - 1);
}
System.out.println(countArray(new int[]{1, 2, 3, 4}, 4));
```

- a) 9
- b) 10 ✓
- c) 11
- d) 12

✓ What will be the output of the following? * 1/1

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

System.out.println(checkLengthPositive(new int[]{1, 2, 3, 4, 5}));
```

- (●) a) true ✓
- ( ) b) false
- ( ) c) null
- ( ) d) 0

✓ What will be the output of the following? * 1/1

```
boolean isReverse(String str) {
  if (str.length() <= 1) return true;
  if (str.charAt(0) != str.charAt(str.length() - 1)) return false;
  return isReverse(str.substring(1, str.length() - 1));
}
System.out.println(isReverse("madam"));
```

- (●) a) true ✓
- ( ) b) false
- ( ) c) null
- ( ) d) 0

✓ What will be the output of the following? * 1/1

```
void traverseArray(int[] arr, int n) {
  if (n <= 0)
    return;
  System.out.print(arr[n - 1] + " ");
  traverseArray(arr, n - 1);
}
traverseArray(new int[]{1, 2, 3, 4, 5}, 5);
```

- ( ) a) 1 2 3 4 5
- (●) b) 5 4 3 2 1 ✓
- ( ) c) 1 5 2 4 3
- ( ) d) 3 2 1 5 4

✓ What is the output of the following recursive function? * 1/1

```
int power(int n) {
  if (n == 1) return 1;
  return n * power(n - 1);
}
System.out.println(power(4));
```

- (●) a) 24 ✓
- ( ) b) 16
- ( ) c) 12
- ( ) d) 10

✓ What will be the output of the following? *1/1

```
int sumOfMultiplesOfThree(int[] arr) {
  int sum = 0;
  for (int num : arr) {
    if (num % 3 == 0) {
      sum += num;
    }
  }
  return sum;
}

System.out.println(sumOfMultiplesOfThree(new int[]{1, 2, 3, 4, 5, 6, 7, 8,
9, 12, 15}));
```

- (●) a) 45 ✓
- ( ) b) 30
- ( ) c) 20
- ( ) d) 15

✗ What will be the result of the function when fun(6) is called? * 0/1

```
int fun(int n) {
  if (n == 0) return 0;
  if (n % 2 == 0) return fun(n - 1) + n;
  return fun(n - 1);
}
```

- a) 9
- b) 12
- c) 6
- d) 18 ✗

Correct answer

- b) 12

✓ What does this recursive function compute? * 1/1

```
int reverseNumber(int n) {
  if (n == 0) return 0;
  return n % 10 + reverseNumber(n / 10);
}
System.out.println(reverseNumber(1234));
```

- a) 10 ✓
- b) 9
- c) 11
- d) 8

✓ What is the purpose of the following Java code snippet that uses recursion? *1/1

```
public int countOdd(int[] arr, int n) {
  if (n <= 0) {
    return 0;
  } else {
    return arr[n - 1] +  countOdd(arr, n - 1);
  }
}
```

- a) The average of the array elements
- b) The sum of odd elements in the array
- c) The sum of all array elements ✓
- d) The factorial of the array elements

✗ What will be the output of the following? * 0/1

```
int sumOdd(int n) {
  if (n <= 0) return 0;
  if (n % 2 == 0) return n + sumOdd(n - 1);
  return sumOdd(n - 1);
}
System.out.println(sumOdd(10));
```

- a) 25 ✗
- b) 30
- c) 55
- d) 20

Correct answer

- b) 30

✗ What will be the output of the following? * 0/1

```
int findLargest(int[] arr) {
  int min = Integer.MIN_VALUE;
  int max = Integer.MIN_VALUE;

  for (int num : arr) {
    if (num > min) {
      max = min;
      min = num;
    } else if (num > max && num < min) {
      max = num;
    }
  }
  return max;
}

int largest = findLargest(new int[]{5, 3, 9, 1, 4});
System.out.println(largest);
```

- a) 4
- b) 5
- c) 3
- d) 9 ✗

Correct answer

- b) 5

✓ What will be the output of the following? * 1/1

```
double countOccurences(int[] arr) {
  double sum = 0;

  for (int num : arr) {
    sum += num;
  }
  return sum / arr.length;
}

double occurence = countOccurences(new int[]{5, 10, 15, 20, 25});
System.out.println(occurence);
```

- a) 10.0
- b) 15.0 ✓
- c) 20.0
- d) 25.0

✓ What will be the output of the following? * 1/1

```
int sumArray(int[] arr, int target) {
 int count = 0;
 for (int num : arr) {
   if (num == target) {
     count++;
  }
 }
 return count;
}
System.out.println(sumArray(new int[]{1, 2, 2, 3, 1, 1, 4}, 1));
```

- a) 1
- b) 2
- c) 3 ✓
- d) 4

✓ What will be the output of the following? * 1/1

```
int removeDuplicates(int[] arr) {
  if (arr.length == 0) return 0;
  int uniqueIndex = 1;

  for (int i = 1; i < arr.length; i++) {
    if (arr[i] != arr[i - 1]) {
      arr[uniqueIndex++] = arr[i];
    }
  }

  return uniqueIndex;
}

System.out.println(removeDuplicates(new int[]{0, 0, 1, 1, 1, 2, 3, 3, 4}));
```

- a) 5 ✓
- b) 6
- c) 4
- d) 7

✗ What will be the output of the following? * 0/1

```
int sumOfDigits(int n) {
    if (n == 0) {
        return 0;
    }
    return 1 + sumOfDigits(n / 10);
}

System.out.println(sumOfDigits(12345));
```

- (●) a) 15 ✗
- ( ) b) 5
- ( ) c) 12
- ( ) d) 2

Correct answer

- (●) b) 5

✓ What will be the output of the following? * 1/1

```
int countLength(String str) {
    if (str.isEmpty()) return 0;
    return (str.charAt(0) == 'a' ? 1 : 0) + countLength(str.substring(1));
}
System.out.println(countLength("banana"));
```

- ( ) a) 2
- (●) b) 3 ✓
- ( ) c) 1
- ( ) d) 0