SEARCHING

```
// return true if the list contains a single digit anywhere.
// Note that the only way to return false is by examining every element.
// Therefore you can only return false AFTER the loop has ended.

public static boolean findSingleDigits_Glad(int[] nums) {
  for (int n : nums) {
    if (-9 <= n && n <= 9) {
      return true;
    }
  }
  return false;
}


// return false if the list contains a multiple digit number anywhere.
// Note that the only way to return true is by examining every element.
// Therefore you can only return true AFTER the loop has ended.

public static boolean findMultipleDigits_Sad(int[] nums) {
  for (int n : nums) {
    if (n < -9 || 9 < n) {
      return false;
    }
  }
  return true;
}
```

AGGREGATE VALUES

```
public static int countMultiplesOf3(int[] nums) {
  int count = 0
  for (int n : nums) {
    if (n % 3 == 0) {
      count++;
    }
  }
  return count;
}
```

```
public static int sumNonMultiplesOf5(int[] nums) {
  int sum = 0;
  for (int n : nums) {
    if (n % 5 != 0) {
      sum += n;
    }
  }
  return sum;
}
```

```
public static int sumOddIndexedValues(int[] nums) {
  int sum = 0;
  for (int i = 0; i < nums.length; i++) {
    int n = nums[i];
    if (i % 2 != 0) {
      sum += n;
    }
  }
  return sum;
}
```

EXAMINING TWO OR MORE VALUES AT ONCE

```
public static int count13(int[] nums) {
  int count = 0;
  for (int i = 0; i < nums.length-1; i++) {
    int n1 = nums[i];
    int n2 = nums[i+1];
    if (n1 == 1 && n2 == 3) {
      count++;
    }
  }
  return count;
}
```

```
public static int count13(int[] nums) {
  int count = 0;
  for (int i = 0; i < nums.length; i++) {
    if (i + 1 < nums.length) {
      int n1 = nums[i];
      int n2 = nums[i+1];
      if (n1 == 1 && n2 == 3) {
        count++;
      }
    }
  }
  return count;
}
```

```
public static int count13(int[] nums) {
  int count = 0;
  for (int i = 1; i < nums.length; i++) {
    int n1 = nums[i-1];
    int n2 = nums[i];
    if (n1 == 1 && n2 == 3) {
      count++;
    }
  }
  return count;
}
```

```
public static int count13(int[] nums) {
  int count = 0;
  for (int i = 0; i < nums.length; i++) {
    if (i - 1 >= 0) {
      int n1 = nums[i-1];
      int n2 = nums[i];
      if (n1 == 1 && n2 == 3) {
        count++;
      }
    }
  }
  return count;
}
```