

Grading System

Football Scores Problem

This is the most basic solution. Anyone who knows how to iterate through an array will do this. This passes 8/13 cases to pass

```
function counts(teamA, teamB) {  
  const final = [];  
  teamB.forEach((goalsB) => {  
    let count = 0;  
    teamA.forEach((goalsA) => {  
      if (goalsB >= goalsA) count++;  
    });  
    final.push(count);  
  });  
  return final;  
}
```

The following is a little better and passes 12/13 cases, but this problem can be further optimized by implementing a binary search i think.

```
function counts(teamA, teamB) {  
  const final = [];  
  const sortedA = teamA.sort((a, b) => a - b);  
  
  teamB.forEach((goalsB) => {  
    const index = sortedA.findIndex((value) => goalsB < value);  
    final.push(index >= 0 ? index : teamA.length);  
  });  
  return final;  
}
```

Proposed Grading System for Football Scores Problem

1. Yes: pass 9 or more test cases
2. Strong Yes: pass 12 or more test cases

Counting Pairs Problem

basic solution. I am sure you can do worse than this, but this one passes 8/12

```
function countPairs(numbers, k) {  
  const numeros = [...new Set(numbers)]; //remove duplicates
```

```

if (k === 0) return numeros.length;
let count = 0;

numeros.forEach((outer) => {
  numeros.forEach((inner) => {
    if (outer + k === inner) count++;
  });
});
return count;
}

```

this is all i could think to pass all the cases

```

function countPairs(numbers, k) {
  let count = 0;
  let obj = {};
  let nums = [...new Set(numbers)];

  nums.forEach((num) => {
    if (!(num in obj)) {
      obj[num] = num + k;
    }
  });
  nums.forEach((num) => {
    if (obj[obj[num]]) count++;
  });
  return count;
}

```

Proposed Grading System for Counting Pairs Problem

1. Yes: pass 9 or more cases
2. Strong Yes: pass 12 or more cases

Anagram Diff

I think this is a little tougher than the first 2. Also, this might be more about logic than optimization.

I copied this from GeeksForGeeks and it passes 9/14 test cases

```

function countManipulations(s1, s2) {
  let count = 0;

  let char_count = new Array(26);
  for (let i = 0; i < char_count.length; i++) {
    char_count[i] = 0;
  }

  for (let i = 0; i < s1.length; i++) {

```

```

    char_count[s1[i].charCodeAt(0) - "a".charCodeAt(0)]++;
  }

  for (let i = 0; i < s2.length; i++) {
    char_count[s2[i].charCodeAt(0) - "a".charCodeAt(0)]--;
  }

  for (let i = 0; i < 26; ++i) {
    if (char_count[i] !== 0) {
      count += Math.abs(char_count[i]);
    }
  }
  return count / 2;
}

function getMinimumDifference(a, b) {
  const final = [];
  for (let i = 0; i < a.length; i++) {
    if (a[i].length !== b[i].length) {
      final.push(-1);
    } else {
      final.push(countManipulations(a[i], b[i]));
    }
  }
  return final;
}

```

This is what i could come up with for an optimal solutions. it passes all the tests.

```

function getMinimumDifference(a, b) {
  const final = [];

  for (let i = 0; i < a.length; i++) {
    let first = [...a[i]];
    let second = [...b[i]];
    if (first.length !== second.length) {
      final.push(-1);
    } else {
      let count = 0;
      for (let j = 0; j < first.length; j++) {
        const ind = second.indexOf(first[j]);
        if (ind < 0) {
          count++;
        } else {
          first[j] = "_";
          second[ind] = "_";
        }
      }
      final.push(count);
    }
  }
}

```

```
    return final;
}
```

Proposed Grading System for Anagram Difference

1. Yes: pass 10 or more cases
2. Strongly Yes: pass 13 or 14 test cases

Longest Even Length Word

This is an 'EASY' question, there is no optimization or logic needed for this. If candidates can write basic code, they will be able to pass 12/14 cases. Only failing cases are with sentences that dont have even words.

This passes 12/14 cases.

```
function longestEvenWord(sentence) {
    const arr = sentence.split(" ");
    let longest = "";
    for (let word of arr) {
        if (word.length % 2 === 0 && word.length > longest.length) longest =
word;
    }
    console.log(longest);
    return longest;
}
```

Proposed Grading System for Longest Even Length Word

1. Lets get rid of this question and find another