

Problem 3442: Maximum Difference Between Even and Odd Frequency I

Problem Information

Difficulty: Easy

Acceptance Rate: 0.00%

Paid Only: No

Problem Description

You are given a string

s

consisting of lowercase English letters.

Your task is to find the

maximum

difference

$\text{diff} = \text{freq}(a$

1

$) - \text{freq}(a$

2

$)$

between the frequency of characters

a

1

and

a

2

in the string such that:

a

1

has an

odd frequency

in the string.

a

2

has an

even frequency

in the string.

Return this

maximum

difference.

Example 1:

Input:

s = "aaaaabbc"

Output:

3

Explanation:

The character

'a'

has an

odd frequency

of

5

,

and

'b'

has an

even frequency

of

2

.

The maximum difference is

$$5 - 2 = 3$$

.

Example 2:

Input:

s = "abccabccab"

Output:

1

Explanation:

The character

'a'

has an

odd frequency

of

3

,

and

'c'

has an

even frequency

of

2

.

The maximum difference is

$3 - 2 = 1$

.

Constraints:

$3 \leq s.length \leq 100$

s

consists only of lowercase English letters.

s

contains at least one character with an odd frequency and one with an even frequency.

Code Snippets

C++:

```
class Solution {
public:
    int maxDifference(string s) {

    }
};
```

Java:

```
class Solution {
    public int maxDifference(String s) {
```

```
}  
}
```

Python3:

```
class Solution:  
    def maxDifference(self, s: str) -> int:
```

Python:

```
class Solution(object):  
    def maxDifference(self, s):  
        """  
        :type s: str  
        :rtype: int  
        """
```

JavaScript:

```
/**  
 * @param {string} s  
 * @return {number}  
 */  
var maxDifference = function(s) {  
  
};
```

TypeScript:

```
function maxDifference(s: string): number {  
  
};
```

C#:

```
public class Solution {  
    public int MaxDifference(string s) {  
  
    }  
}
```

C:

```
int maxDifference(char* s) {  
  
}
```

Go:

```
func maxDifference(s string) int {  
  
}
```

Kotlin:

```
class Solution {  
    fun maxDifference(s: String): Int {  
  
    }  
}
```

Swift:

```
class Solution {  
    func maxDifference(_ s: String) -> Int {  
  
    }  
}
```

Rust:

```
impl Solution {  
    pub fn max_difference(s: String) -> i32 {  
  
    }  
}
```

Ruby:

```
# @param {String} s  
# @return {Integer}  
def max_difference(s)  
  
end
```

PHP:

```
class Solution {  
  
    /**  
     * @param String $s  
     * @return Integer  
     */  
    function maxDifference($s) {  
  
    }  
}
```

Dart:

```
class Solution {  
    int maxDifference(String s) {  
  
    }  
}
```

Scala:

```
object Solution {  
    def maxDifference(s: String): Int = {  
  
    }  
}
```

Elixir:

```
defmodule Solution do  
    @spec max_difference(s :: String.t) :: integer  
    def max_difference(s) do  
  
    end  
end
```

Erlang:

```
-spec max_difference(S :: unicode:unicode_binary()) -> integer().  
max_difference(S) ->  
.
```


Racket:

```
(define/contract (max-difference s)
  (-> string? exact-integer?)
)
```

Solutions

C++ Solution:

```
/*
 * Problem: Maximum Difference Between Even and Odd Frequency I
 * Difficulty: Easy
 * Tags: string, hash
 *
 * Approach: String manipulation with hash map or two pointers
 * Time Complexity: O(n) or O(n log n)
 * Space Complexity: O(n) for hash map
 */

class Solution {
public:
    int maxDifference(string s) {

    }
};
```

Java Solution:

```
/**
 * Problem: Maximum Difference Between Even and Odd Frequency I
 * Difficulty: Easy
 * Tags: string, hash
 *
 * Approach: String manipulation with hash map or two pointers
 * Time Complexity: O(n) or O(n log n)
 * Space Complexity: O(n) for hash map
 */

class Solution {
    public int maxDifference(String s) {
```

```
}  
}
```

Python3 Solution:

```
"""  
Problem: Maximum Difference Between Even and Odd Frequency I  
Difficulty: Easy  
Tags: string, hash  
  
Approach: String manipulation with hash map or two pointers  
Time Complexity: O(n) or O(n log n)  
Space Complexity: O(n) for hash map  
"""  
  
class Solution:  
    def maxDifference(self, s: str) -> int:  
        # TODO: Implement optimized solution  
        pass
```

Python Solution:

```
class Solution(object):  
    def maxDifference(self, s):  
        """  
        :type s: str  
        :rtype: int  
        """
```

JavaScript Solution:

```
/**  
 * Problem: Maximum Difference Between Even and Odd Frequency I  
 * Difficulty: Easy  
 * Tags: string, hash  
 *  
 * Approach: String manipulation with hash map or two pointers  
 * Time Complexity: O(n) or O(n log n)  
 * Space Complexity: O(n) for hash map  
 */
```

```

/**
 * @param {string} s
 * @return {number}
 */
var maxDifference = function(s) {

};

```

TypeScript Solution:

```

/**
 * Problem: Maximum Difference Between Even and Odd Frequency I
 * Difficulty: Easy
 * Tags: string, hash
 *
 * Approach: String manipulation with hash map or two pointers
 * Time Complexity: O(n) or O(n log n)
 * Space Complexity: O(n) for hash map
 */

function maxDifference(s: string): number {

};

```

C# Solution:

```

/*
 * Problem: Maximum Difference Between Even and Odd Frequency I
 * Difficulty: Easy
 * Tags: string, hash
 *
 * Approach: String manipulation with hash map or two pointers
 * Time Complexity: O(n) or O(n log n)
 * Space Complexity: O(n) for hash map
 */

public class Solution {
    public int MaxDifference(string s) {

    }
}

```

```
}
```

C Solution:

```
/*
 * Problem: Maximum Difference Between Even and Odd Frequency I
 * Difficulty: Easy
 * Tags: string, hash
 *
 * Approach: String manipulation with hash map or two pointers
 * Time Complexity: O(n) or O(n log n)
 * Space Complexity: O(n) for hash map
 */

int maxDifference(char* s) {

}
```

Go Solution:

```
// Problem: Maximum Difference Between Even and Odd Frequency I
// Difficulty: Easy
// Tags: string, hash
//
// Approach: String manipulation with hash map or two pointers
// Time Complexity: O(n) or O(n log n)
// Space Complexity: O(n) for hash map

func maxDifference(s string) int {

}
```

Kotlin Solution:

```
class Solution {
    fun maxDifference(s: String): Int {

    }
}
```

Swift Solution:

```

class Solution {
    func maxDifference(_ s: String) -> Int {

    }
}

```

Rust Solution:

```

// Problem: Maximum Difference Between Even and Odd Frequency I
// Difficulty: Easy
// Tags: string, hash
//
// Approach: String manipulation with hash map or two pointers
// Time Complexity: O(n) or O(n log n)
// Space Complexity: O(n) for hash map

impl Solution {
    pub fn max_difference(s: String) -> i32 {

    }
}

```

Ruby Solution:

```

# @param {String} s
# @return {Integer}
def max_difference(s)

end

```

PHP Solution:

```

class Solution {

    /**
     * @param String $s
     * @return Integer
     */
    function maxDifference($s) {

    }

}

```

Dart Solution:

```
class Solution {  
  int maxDifference(String s) {  
  
  }  
}
```

Scala Solution:

```
object Solution {  
  def maxDifference(s: String): Int = {  
  
  }  
}
```

Elixir Solution:

```
defmodule Solution do  
  @spec max_difference(s :: String.t) :: integer  
  def max_difference(s) do  
  
  end  
end
```

Erlang Solution:

```
-spec max_difference(S :: unicode:unicode_binary()) -> integer().  
max_difference(S) ->  
.
```

Racket Solution:

```
(define/contract (max-difference s)  
  (-> string? exact-integer?)  
)
```