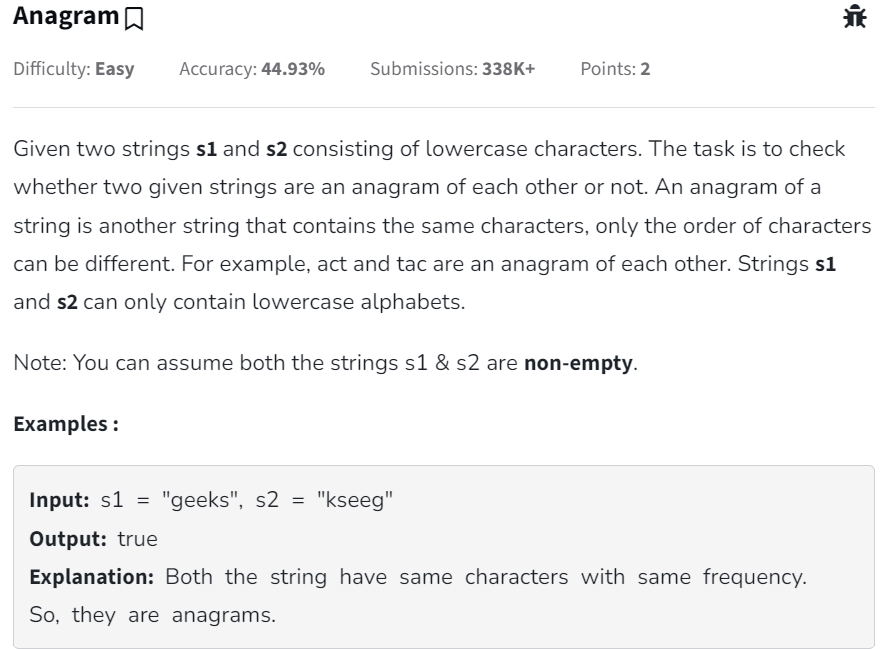
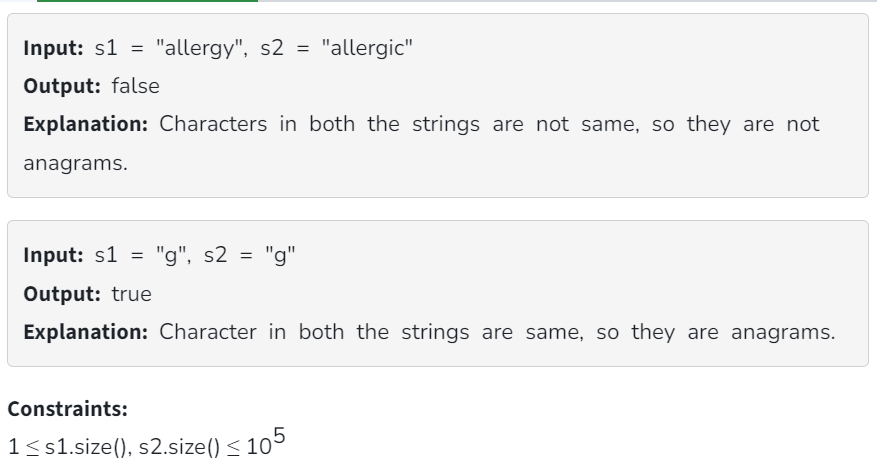
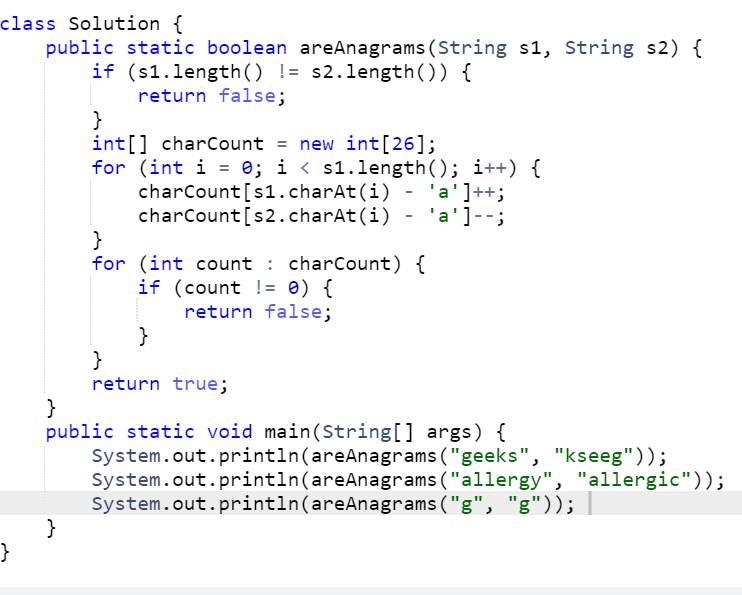
**12/11/2024** **GEEKS FOR GEEKS PROBLEMS** **FIZA THASHIN A (AI&DS)**

**1)**

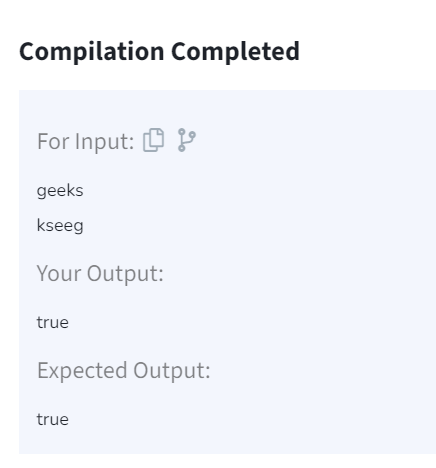


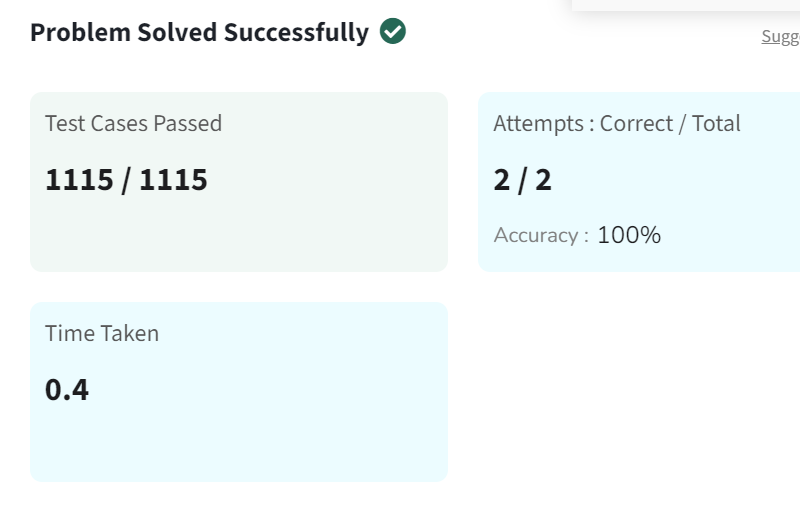
****

**SOLUTION**

****

**OUTPUT**

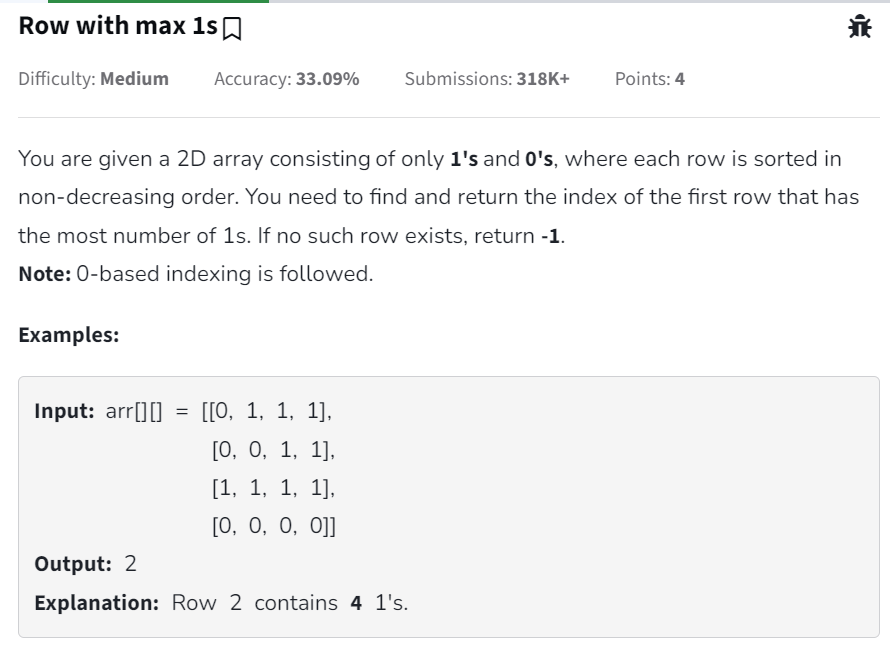
****

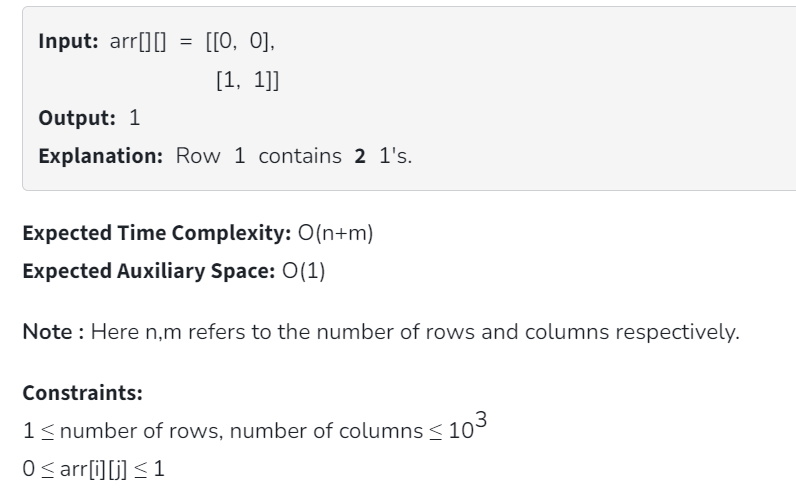
****

**TIME COMPLEXITY**

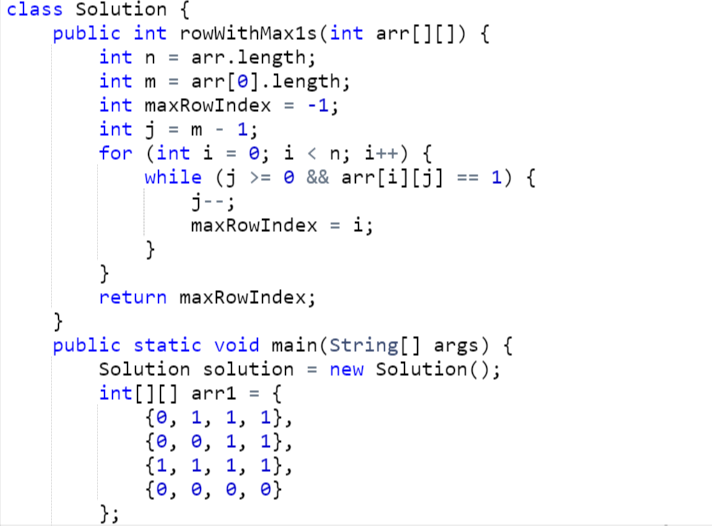
**O(n) -** where n is the length of the strings.

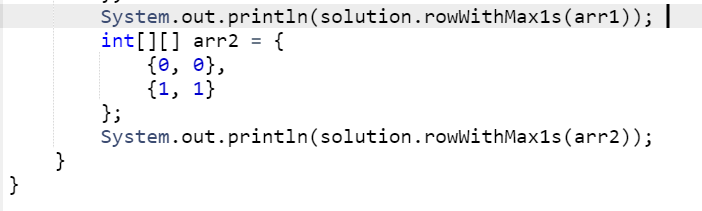
**2)**

****

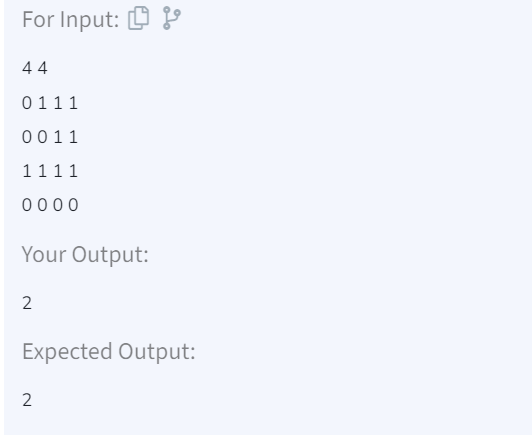
****

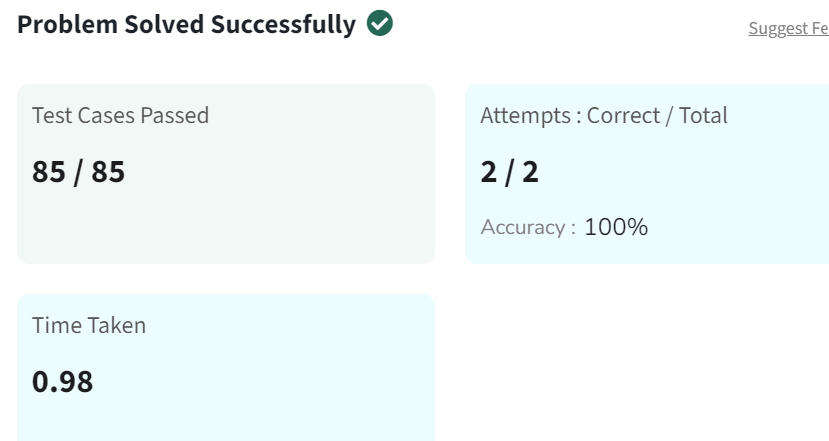
**SOLUTION**

****

****

**OUTPUT**

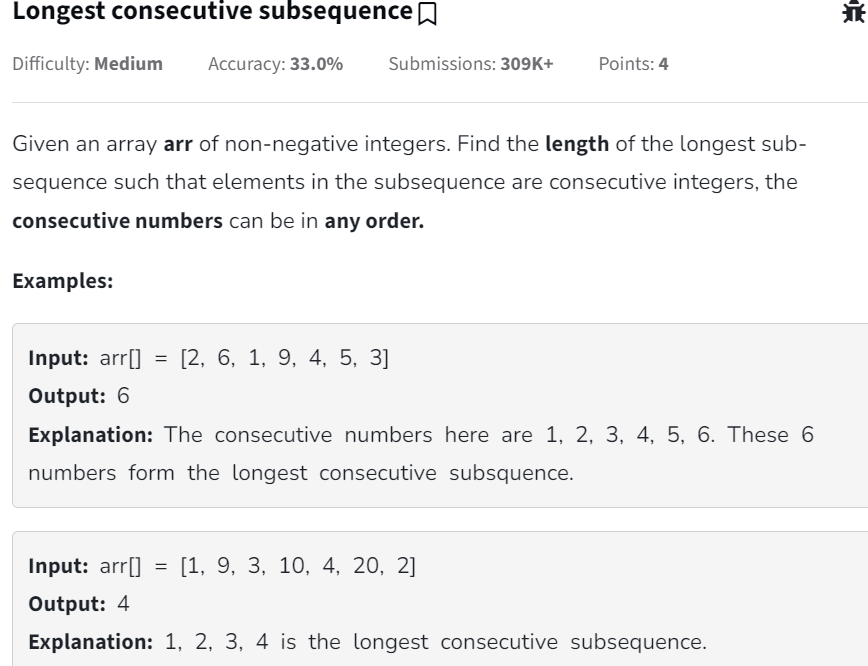
****

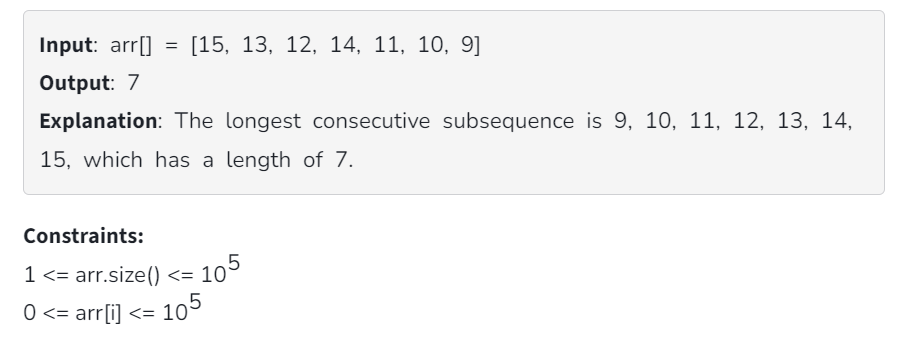
****

**TIME COMPLEXITY**

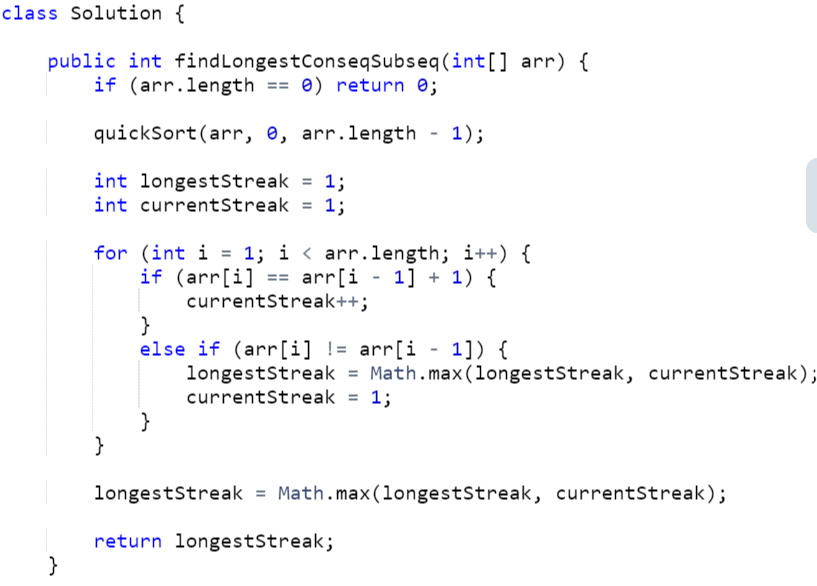
**O(n + m) -** The outer loop runs n times, and the inner loop in total runs at most m times for all rows combined.

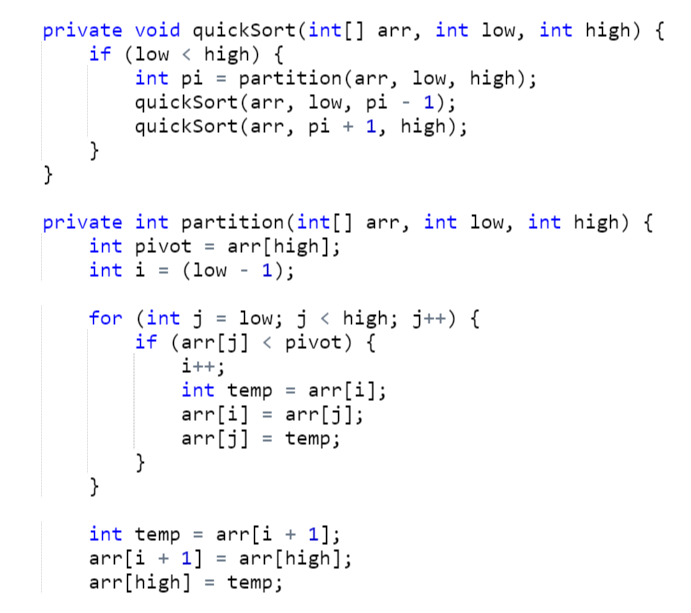
**3)**

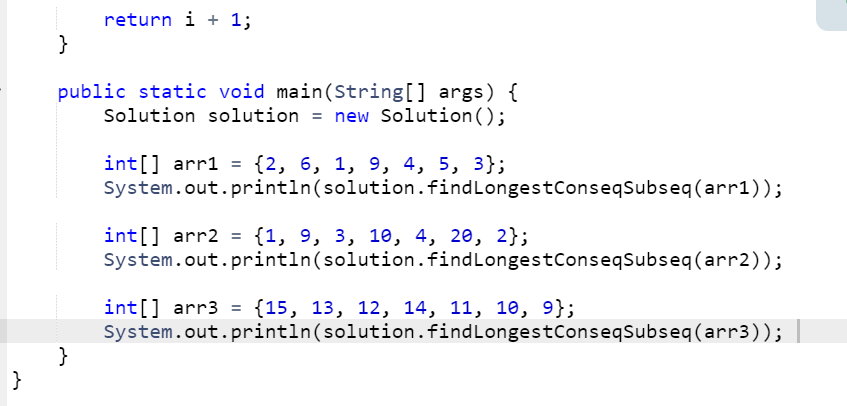
****

****

**SOLUTION**

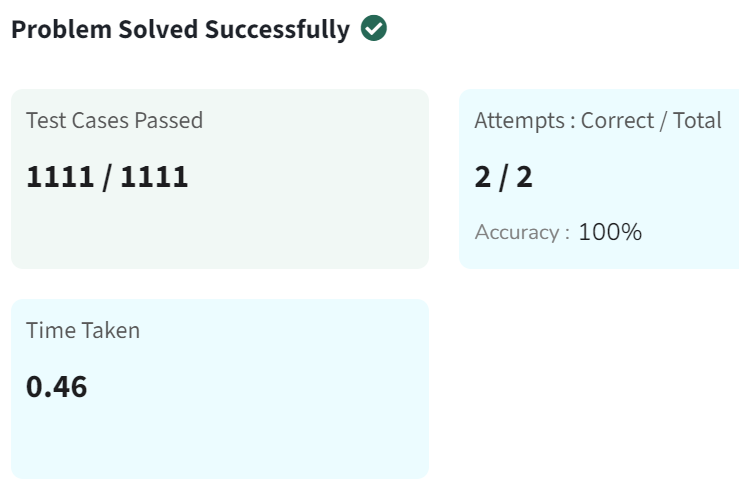
****

****

****

**OUTPUT**

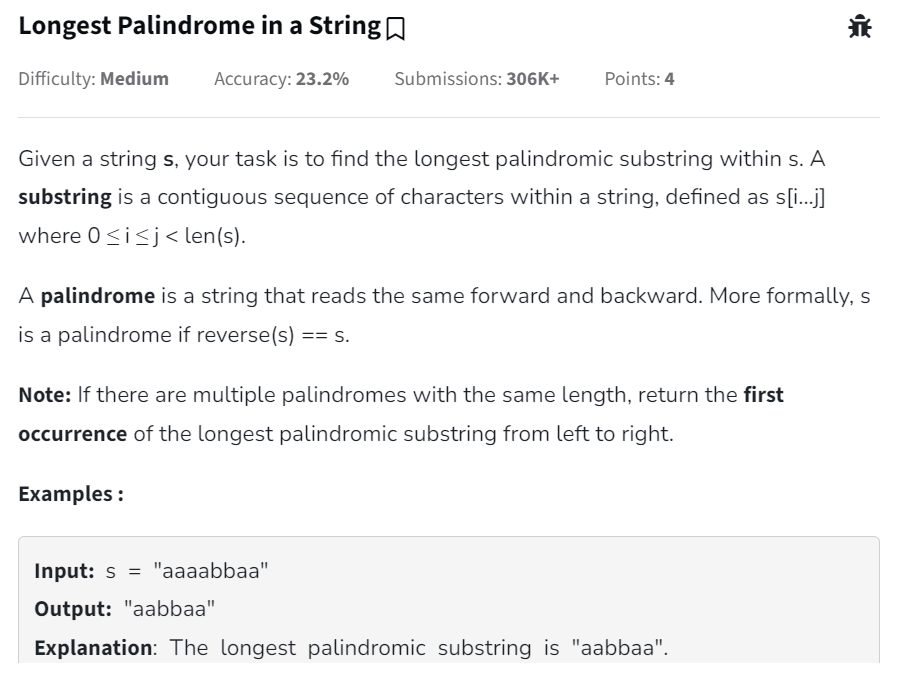
****

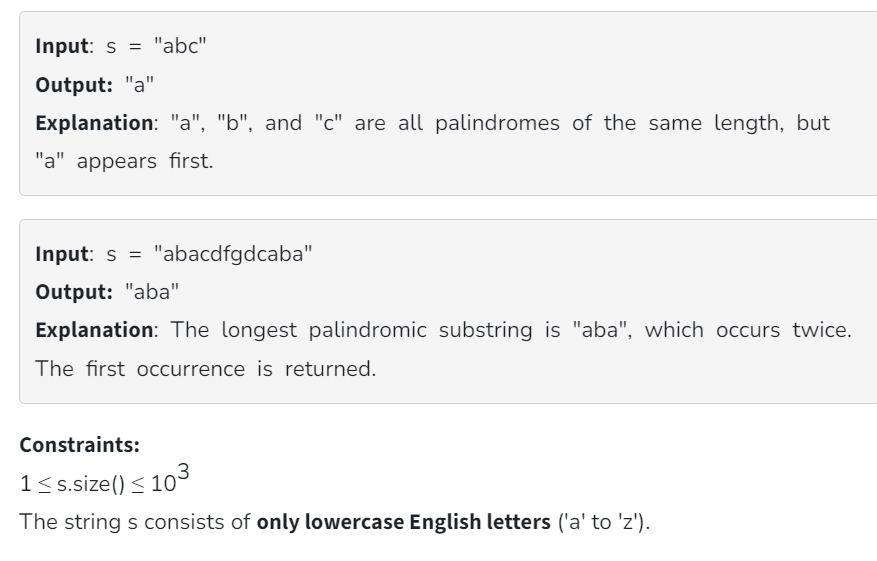
****

**TIME COMPLEXITY**

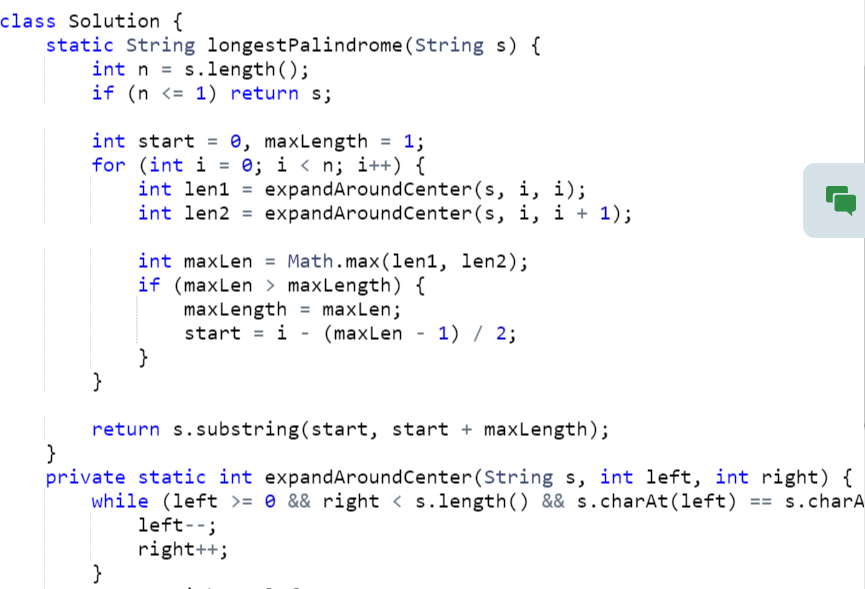
**O(n log n) -** The average time complexity of QuickSort is O(n log n), where n is the number of elements in the array**.**

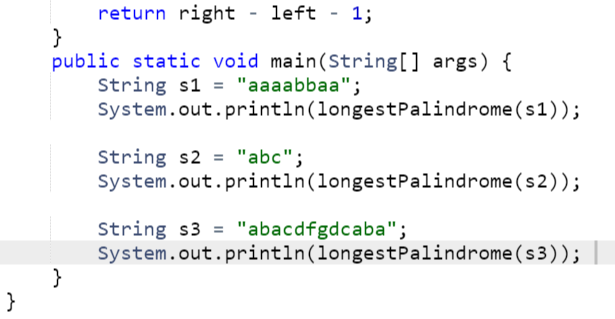
**4)**

****

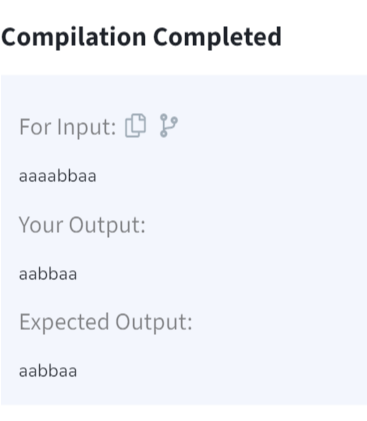
****

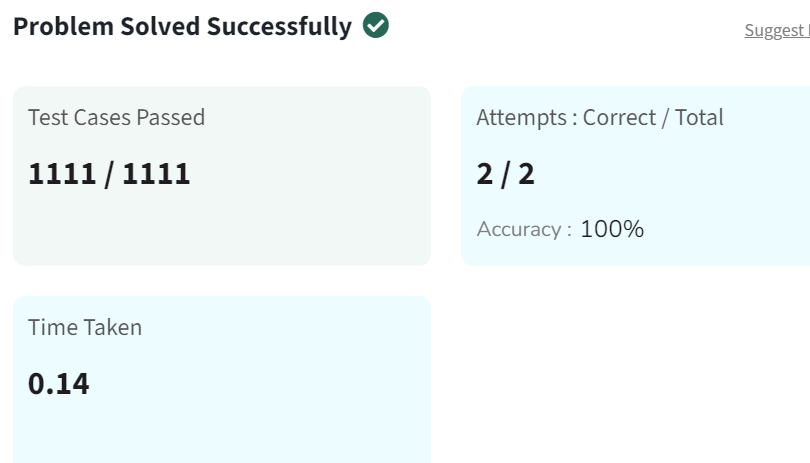
**SOLUTION**

****

****

**OUTPUT**

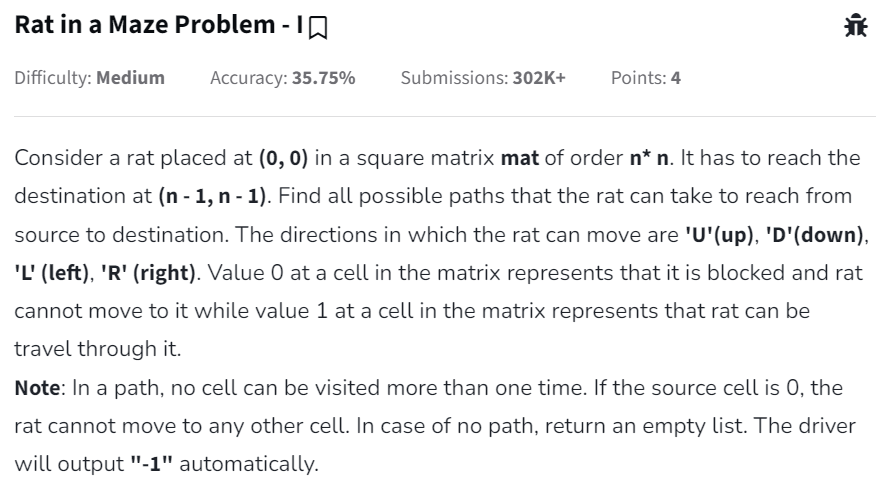
****

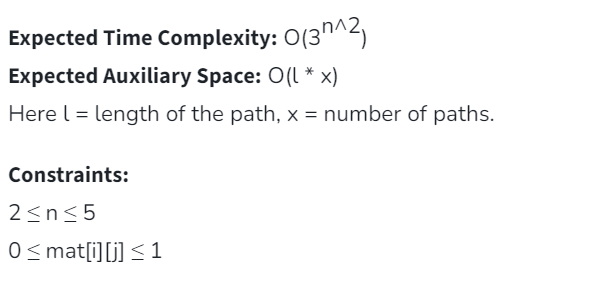
****

**TIME COMPLEXITY**

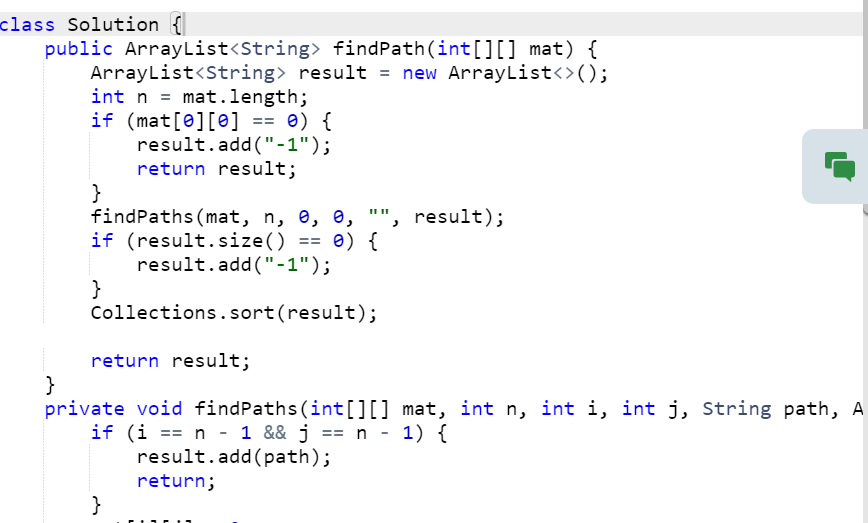
**O(n^2) -** because for each of the n characters, the algorithm performs an O(n) expansion to check the palindrome.

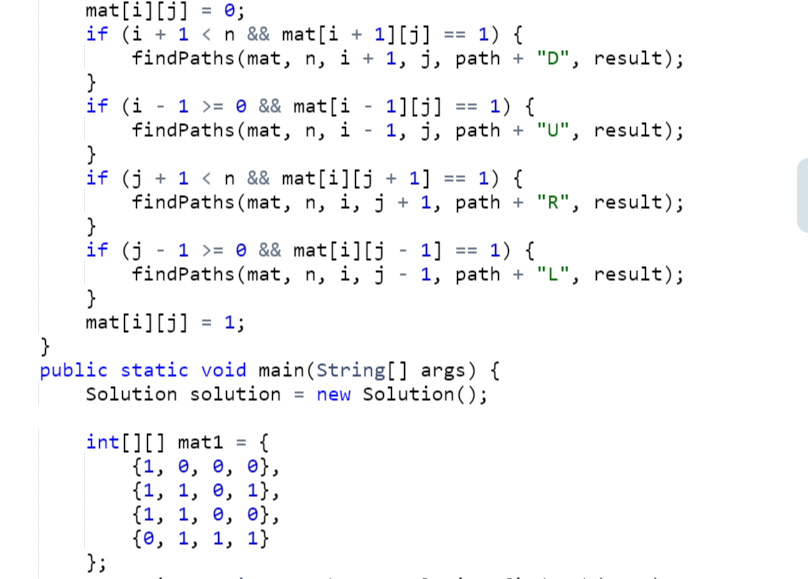
**5)**

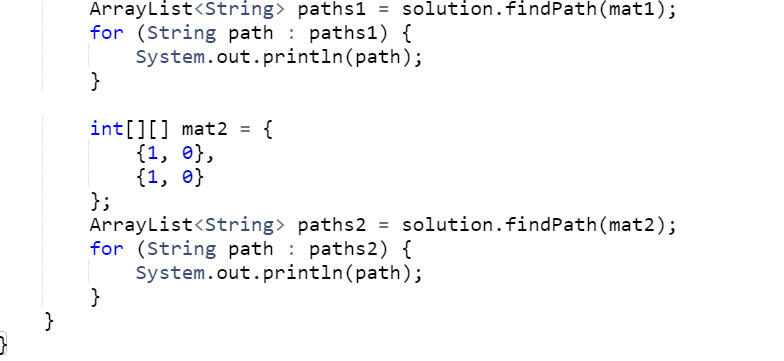
****

****

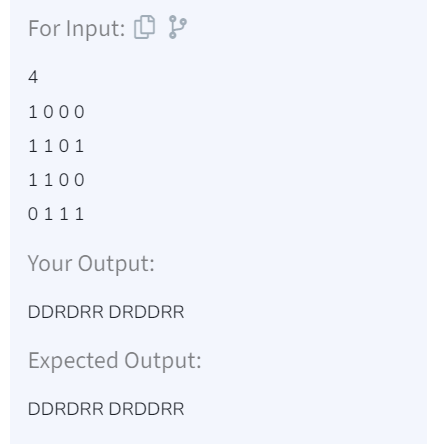
**SOLUTION**

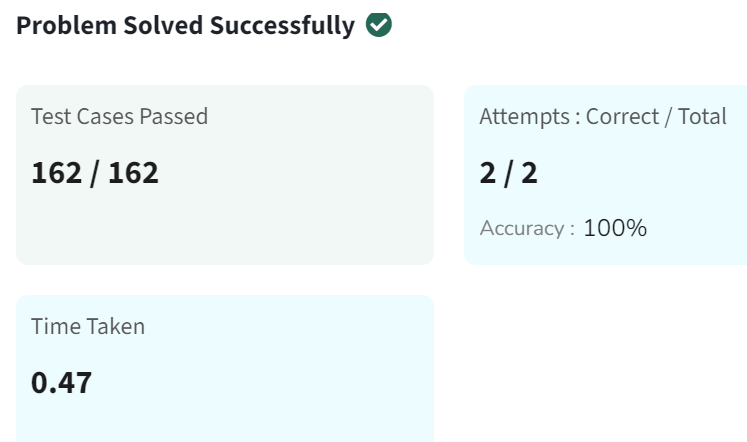
****

****

****

**OUTPUT**

****

****

**TIME COMPLEXITY: O(4^n \* n log n)**