

**UNIVERSITY INSTITUTE OF ENGINEERING**

**Department of Computer Science & Engineering**

# Subject Name: Competitive Coding 2

**Subject Code:** 20CSP-351

**Submitted to: Submitted by:**

Faculty name: Mr. Ankesh Gupta Name: Sahil Kaundal

UID: 21BCS8197

Section: 616

Group: A

**INDEX**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Ex. No** | **List of Experiments** | **Conduct (MM: 12)** | **Viva**  **(MM: 10)** | **Record (MM: 8)** | **Total**  **(MM: 30)** | **Remarks/Signature** |
| 1. | Arrays, Stacks, Queues linked list |  |  |  |  |  |
| 2. | String Matching |  |  |  |  |  |
| 3. |  |  |  |  |  |  |
| 4. |  |  |  |  |  |  |
| 5. |  |  |  |  |  |  |
| 6. |  |  |  |  |  |  |
| 7. |  |  |  |  |  |  |
| 8. |  |  |  |  |  |  |
| 9. |  |  |  |  |  |  |
| 10. |  |  |  |  |  |  |

**Experiment 2.1**

**Student Name:** Sahil Kaundal **UID:** 21BCS8197

**Branch:** BE CSE (Lateral Entry) **Section/Group:** 616/A

**Semester:** 6th **Date of Performance:** 22/02/2023

**Subject Name:** CC-2 Lab **Subject Code:** 20CSP-351

1. **Aim/Overview of the practical:**

Repeated String Match

Given two strings a and b, return the minimum number of times you should repeat string a so that string b is a substring of it. If it is impossible for b​​​​​​ to be a substring of a after repeating it, return -1

https://leetcode.com/problems/repeated-string-match/

1. **Apparatus / Simulator Used:**

* Windows 7 or above
* Google Chrome

**3. Code:**

class Solution {

public:

    int repeatedStringMatch(string A, string B) {

        int repeatedCount = 1;

        string repeatedString = A;

        while(A.find(B) == std::string::npos && A.length() <= 10000)

        {

            repeatedCount++;

            A += repeatedString;

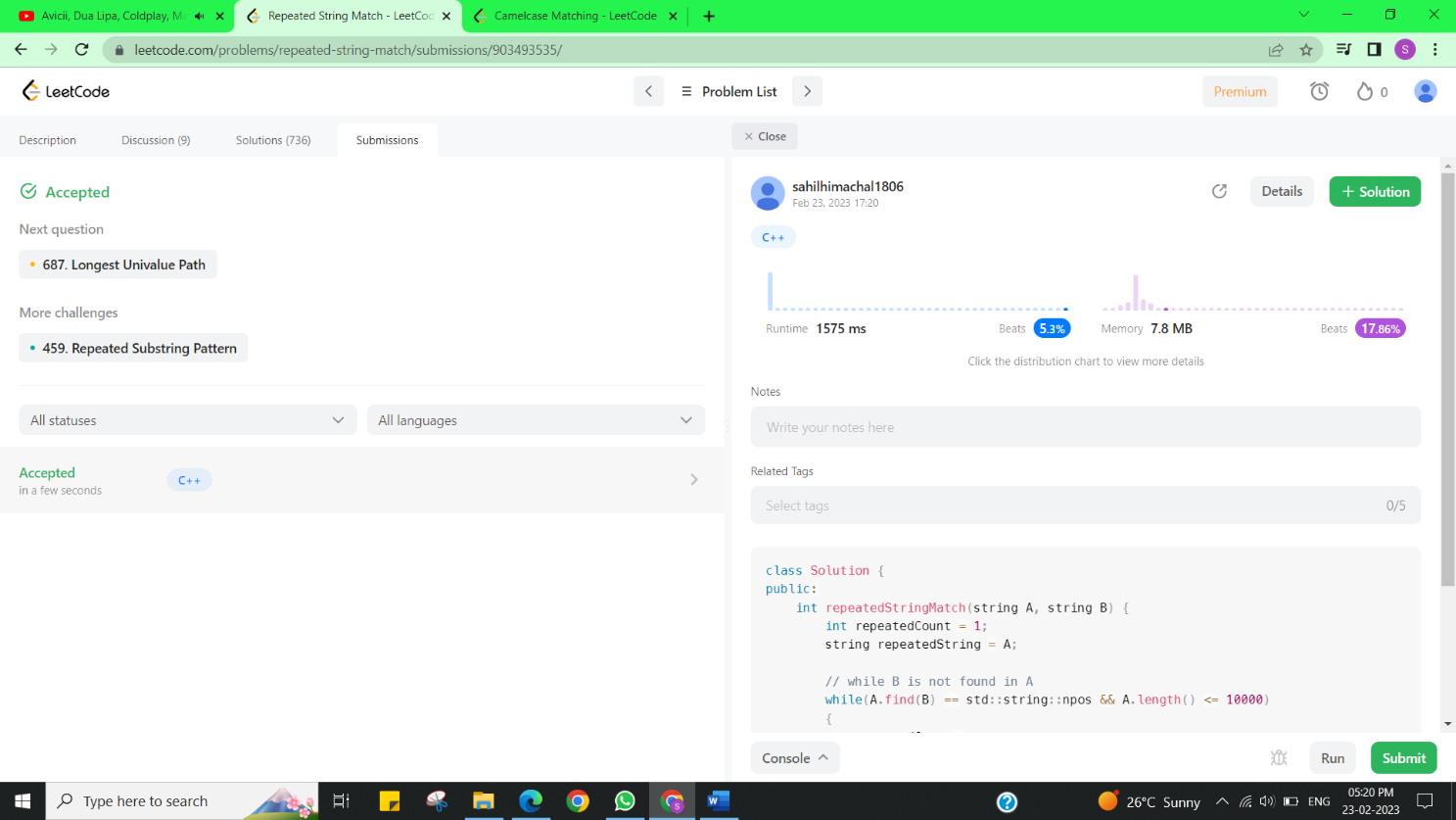
        }

        return (A.length() > 10000) ? -1 : (repeatedCount);

    }

};

**4. Result/Output/Writing Summary:**



**Experiment 2.2**

1. **Aim/Overview of the practical:**

Camelcase Matching

Given an array of strings queries and a string pattern, return a boolean array answer where answer[i] is true if queries[i] matches pattern, and false otherwise.

<https://leetcode.com/problems/camelcase-matching/>

1. **Apparatus / Simulator Used:**

* Windows 7 or above
* Google Chrome

1. **Code:**

class Solution {

 public:

  vector<bool> camelMatch(vector<string>& queries, string pattern) {

    vector<bool> ans;

    for (const string& query : queries)

      ans.push\_back(isMatch(query, pattern));

    return ans;

  }

 private:

  bool isMatch(const string& query, const string& pattern) {

    int j = 0;

    for (const char c : query)

      if (j < pattern.length() && c == pattern[j])

        ++j;

      else if (isupper(c))

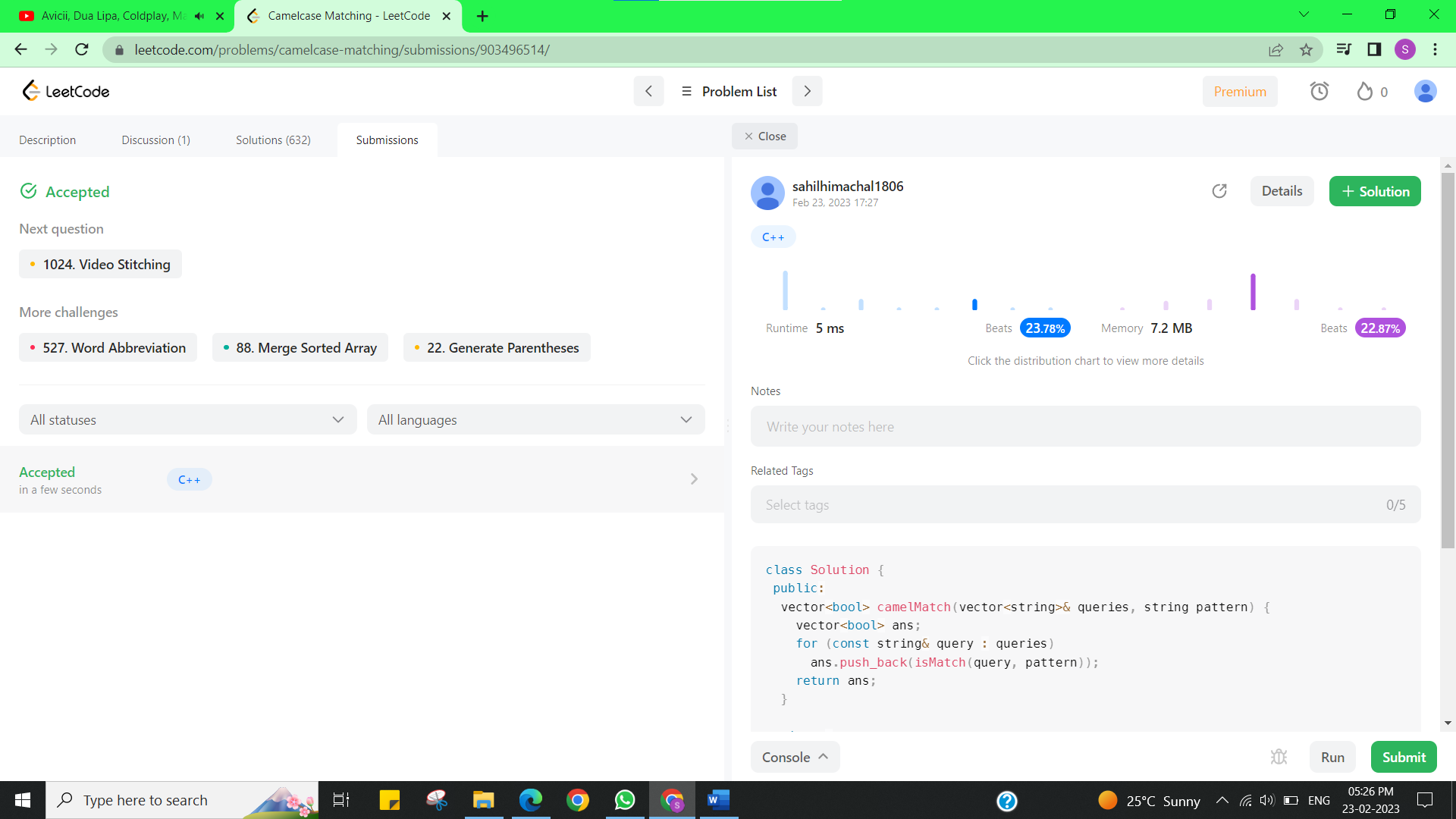
        return false;

    return j == pattern.length();

  }

};

1. **Result/Output/Writing Summary:**

****

**Learning outcomes (What I have learnt):**

* + Learned the concept of Repeated string match.
  + Learnt about Camelcase Matching.

**Evaluation Grid (To be created as per the SOP and Assessment guidelines by the faculty):**

|  |  |  |  |
| --- | --- | --- | --- |
| Sr. No. | Parameters | Marks Obtained | Maximum Marks |
| 1. |  |  |  |