# Evaluation Rubric :

|  |  |  |
| --- | --- | --- |
| **Evaluation parameter** | **Does not meet specifications** | **Meets specifications** |
| **Problem statement** |  | **2** |
| Problem Statement must be clearly defined |  | **√** |
| Expected input and output formats must be described |  | **√** |
| Explain the problem statement with an example(if applicable) |  | **√** |
| **Expected input & output** |  | **3** |
| Minimum of 5 test cases (if applicable) |  | **√** |
| Coverage |  | **√** |
| Border condition |  | **√** |
| Unexpected inputs |  | **√** |
| **Solution** |  | **5** |
| The correctness of the solution. |  | **√** |
| Check for all the elements (tokens) of the problem (Assignment, Arithmetic, conditional, relational, input, output etc) |  | **√** |
| **Trace Table :** |  | **5** |
| Columns are variables, conditions, print statements |  | **√** |
| Order |  | **√** |
| Trace table for each function(If applicable) |  | **√** |
| labeling the columns |  | **√** |
| Coverage (conditions, iterations... etc) |  | **√** |
| **Final Result** |  | **2** |
| Executable File Submission |  | **√** |
| **Executable File** |  | **3** |
| Check with all test cases |  | **√** |

## 

**Problem Statement**: **(2 Marks)**

Check if the Substring is present in the given string : Given two strings S1(string) and S2(substring), print "True" if substring(S2) is present in string(S1). If substring(S2) is not present in string(S1) then print "False".

**Test cases: (3 Marks)**

|  |  |
| --- | --- |
| **Expected Input** | **Expected Output** |
| Str = Daddy s=dad | True |
| Str= Aaba s= h | False |
| Str= aababa s=aab | True |
| Str = abcd s=cd | True |
| Str = a s=ab | False |

**Solution**: **(5 Marks)**

**Step 1:** START

**Step 2:** Input String and substring

Step 3: d= POSITION( String , substring)

Step 4: if d= 0 Print “False”

Else Print “True”

Step 5: End

**Trace Table** : **(5 Marks)**

|  |  |  |  |
| --- | --- | --- | --- |
| Test Case 1: |  |  |  |
| String | Substring | d=position(str,s) | Output |
| Daddy | Dad | 3 | TRUE |

|  |  |  |  |
| --- | --- | --- | --- |
| Test Case 2: |  |  |  |
| String | Substring | d=position(str,s) | Output |
| Hello | Moto | 0 | FALSE |

|  |  |  |  |
| --- | --- | --- | --- |
| Test Case 3: |  |  |  |
| String | Substring | d=position(str,s) | Output |
| Aaba | h | 0 | FALSE |

|  |  |  |  |
| --- | --- | --- | --- |
| Test Case 4: |  |  |  |
| String | Substring | d=position(str,s) | Output |
| Aababa | aba | 2 | TRUE |

|  |  |  |  |
| --- | --- | --- | --- |
| Test Case 5: |  |  |  |
| String | Substring | d=position(str,s) | Output |
| Key | Key Kettle | 1 | TRUE |

**Final Result :** **(2 Marks)**

|  |  |  |  |
| --- | --- | --- | --- |
| **Expected input** | **Expected output** | **Actual output** | **Test result** |
| Str = Daddy s=dad | True | Found | 1 |
| Str= Aaba s= h | False | Not found | 0 |
| Str= aababa s=aab | True | Found | 1 |