# Evaluation Rubric :

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

# 

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

Check if a password strength is strong or weak using below criteria

Its length is at least 6

It contains at least one digit.

It contains at least one lowercase English character.

It contains at least one uppercase English character.

It contains at least one special character. The special characters are: !@#$%^&\*()-+.

**Test cases: (3 Marks)**

|  |  |
| --- | --- |
| **Expected Input** | **Expected Output** |
| Hel0$g | valid |
| abcdA | invalid |
| Kitty2\*g | valid |

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

**START**

**QUERY "Enter Passowrd: ", a**

**a=tocharacters(a)**

**l=length(a)**

**i=1**

**c=0**

**d=0**

**e=0**

**f=0**

**IF l>=6 THEN**

**WHILE i<=l DO**

**IF a[i]>='a' AND a[i]<='z' THEN**

**c=c+1**

**ENDIF**

**IF a[i]>='A' AND a[i]<='Z' THEN**

**d=d+1**

**ENDIF**

**IF a[i]>='0' AND a[i]<='9' THEN**

**e=e+1**

**ENDIF**

**IF a[i]>='!' AND a[i]<='/' OR a[i]>=':' AND a[i]<='@' THEN**

**f=f+1**

**ENDIF**

**i=i+1**

**ENDWHILE**

**IF c>0 && d>0 && e>0 && f>0 THEN**

**WRITE "1"**

**ELSE**

**WRITE "0"**

**ENDIF**

**ELSE**

**WRITE "0"**

**ENDIF**

**END**

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

**Test Case 1:Hel0$g**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| a | n | i | c a | d A | e 0 | f $ | Result |
| Hel0$g | 6 | 0 | 0 | 0 | 0 | 0 | 1 |
| H |  | 1 |  | 1 |  |  | 1 |
| e |  | 2 |  |  | 1 |  | 1 |
| l |  | 3 | 1 |  |  |  | 1 |
| 0 |  | 4 |  |  | 1 |  | 1 |
| $ |  | 5 |  |  |  | 1 | 1 |
| g |  | 6 | 1 | 1 | 1 | 1 | 1 |
|  |  |  |  |  |  |  | Valid |

**Test Case 2: abcdA**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| a | n | i | c a | d A | e 0 | f $ | Result |
| abcdA | 5 | 0 | 1 |  |  |  | 1 |
|  |  | 1 | 1 |  |  |  | 1 |
|  |  | 2 | 1 |  |  |  | 1 |
|  |  | 3 | 1 |  |  |  | 1 |
|  |  | 4 | 1 | 1 | 0 | 0 | 0 |
|  |  |  |  |  |  |  | Invalid |

Test Case 3: Kitty2\*g

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| a | n | i | c a | d A | e 0 | f $ | Result |
| Kitty2\*g | 8 | 0 | 0 | 0 | 0 | 0 | 1 |
| H |  | 1 |  | 1 |  |  | 1 |
| e |  | 2 | 1 |  |  |  | 1 |
| l |  | 3 | 1 |  |  |  | 1 |
| 0 |  | 4 | 1 |  |  |  | 1 |
| $ |  | 5 | 1 |  |  |  | 1 |
| g |  | 6 |  |  | 1 |  | 1 |
|  |  | 7 | 1 |  |  |  | 1 |
|  |  | 8 |  |  | 1 |  | 1 |
|  |  |  |  |  |  | 1 | 1 |
|  |  |  | 1 | 1 | 1 | 1 | 1 |
|  |  |  |  |  |  |  | Valid |

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

|  |  |  |  |
| --- | --- | --- | --- |
| **Expected input** | **Expected output** | **Actual output** | **Test result** |
| Hel0$g | valid | valid | 1 |
| abcdA | invalid | invalid | 0 |
| Kitty2\*g | valid | valid | 1 |