Formal Languages and Automata theory

CSE A 4th Semester Group A2

The National Institute of Engineering, Mysuru

**Problem statement:**

Design an NFA to check the strength of a given input password, whether strong or not and implement the same in code.

**BitWarden = {Q, δ, ∑, q0, F}**

**Q = {**q0, q1, q2, q3, q4, q5, q6, q7, q8, …., q31, q32, q33**}**

**∑ = {**0-9, a-z, A-Z, S**}**

**Here,**

* **0-9:** Represents all digits present in the keyboard.
* **a-z:** Represents all lowercase alphabets present in the keyboard.
* **A-Z:** Represents all uppercase alphabets present in the keyboard.
* **S:** Represents all symbols. i.e., non-alphanumeric characters. Here,

**S =** **{**~, `, !, @, #, $, %, ^, &, \*, (, ), \_, -, =, +, {, }, [, ], :, ;, ‘, “, <, >, ., ,, ?, /, |, \**}**

**q0 = {q0}**

**F = {q33}**

**Machine constraints:**

* The designed machine can accept a minimum of 5 characters, 4 of them should be different from each other and the 5th character is recurring.
* There should be no blank or white spaces in the provided password, if so, the machine rejects it.
* Maximum of 128 characters can be accepted by the machine.

**Machine features:**

* There is no restriction on the occurrence of a character for example an uppercase, lowercase, digit and a symbol can appear anywhere in the password any number of times.
* Password strength is displayed if the inputted string does not meet the requirements for a strong password

**Code specifications:**

**Input:** password of type string given as input

* C++ language has been used to develop the machine as classes with inheritance provide the required security.
* There are 4 classes namely Tier1, Tier2, Tier3, Tier4. Each class performs the following activities:
  + **Tier1:** consists of a member function checkTier1() which assesses the first character and choses path from q0 state to q1, q2, q3, q4 states according to design.
  + **Tier2:** consists of a member function checkTier2() which assesses subsequent characters from either of q1, q2, q3, q4 states to their corresponding states q5, q6, q7, q8, …., q16 in the design.
  + **Tier3:** consists of a member function checkTier3() which assesses subsequent characters from either of q5, q6, q7, q8, …, q15, q16 states to their corresponding states q17, q18, q19, …., q28 in the design.
  + **Tier4:** consists of member function checkTier4() which assesses subsequent characters from paths traversed till tier3 to either of q29, q30, q31, q32 states and then to final state q33 in the design. Another member function checkFinal() decides whether given password is strong or not.
* If the final state is reached then a password ‘strong’ message is displayed to the user. Otherwise, the strength of the password is displayed, if the machine dies in tier2 class then password is assumed to be ‘weak’, if machine dies in tier3 class password is assumed to be ‘ok’, if the machine dies in Tier4 class without reaching final state then password is assumed to be ‘good’.

**Output:** Strength of password is displayed.