**PRACTICAL # 03**

**OBJECT:**

Designing DFA for floating point number tokens recognition.

**THEORY:**

Integer tokens are simple digits. Floating point numbers on the other hand contain fractional parts. Even if the number to the right of the decimal point is 0, it is still a fractional part of the number. Floating point numbers can be positive or negative. A typical floating point number is represented as

|  |  |  |
| --- | --- | --- |
| Sign | Exponent | Mantissa |

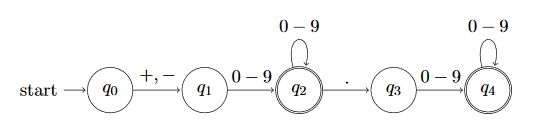
To design lexical analyzer to recognize floating point numbers, we need to create a DFA or regular expression to represent floating point number pattern.

**DFA:**

Regular Expression for floating point numbers:

***(+-)(0-9)(0-9)\* . (0-9)(0-9)\****

DFA for floating point numbers:



**ACTIVITIES**

**Activity 1**

Design a DFA and regular expression for floating point numbers that allows to skip the + sign if the number is positive.

**REVIEW QUESTIONS**

1. What are the parts of a floating point number?
2. What is precision in floating point numbers?