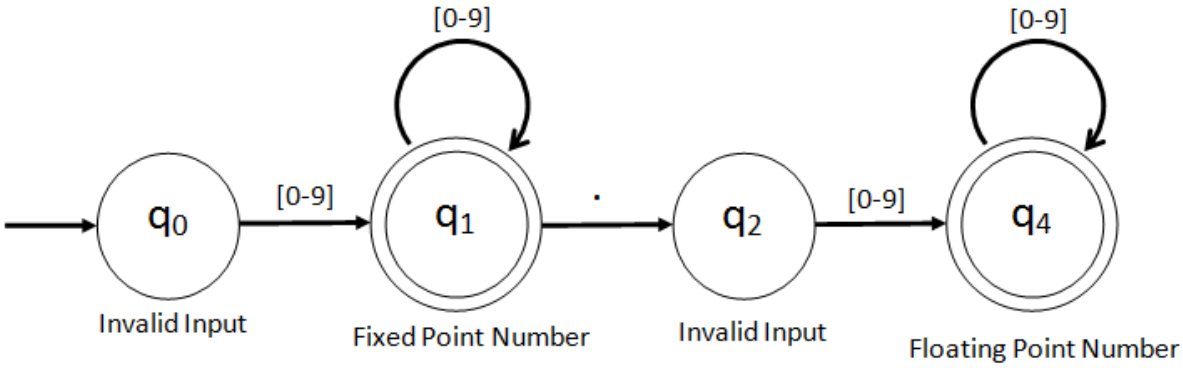
Lexical Analysis

Experiment 02: Finding out Numeric Constants.

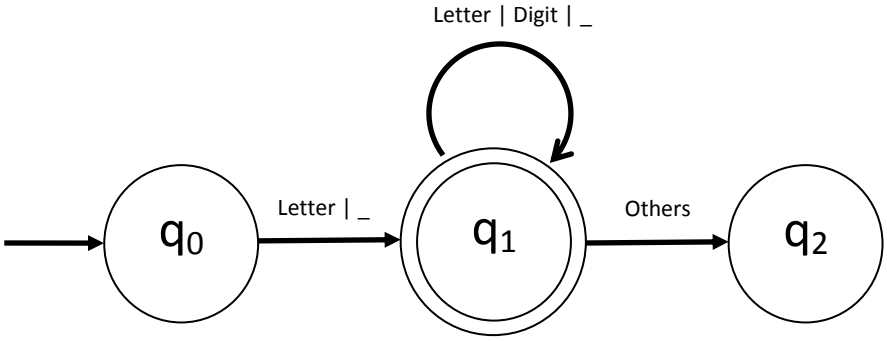
Write a program to detect whether the entered number is floating point number or fixed point number or others. In this above program we need to create a Finite Automata that accepts a language of digits consisting of [0-9] and identify it as fixed point or floating point number.

Corresponding NFA for the above program is:-



Lab Task:

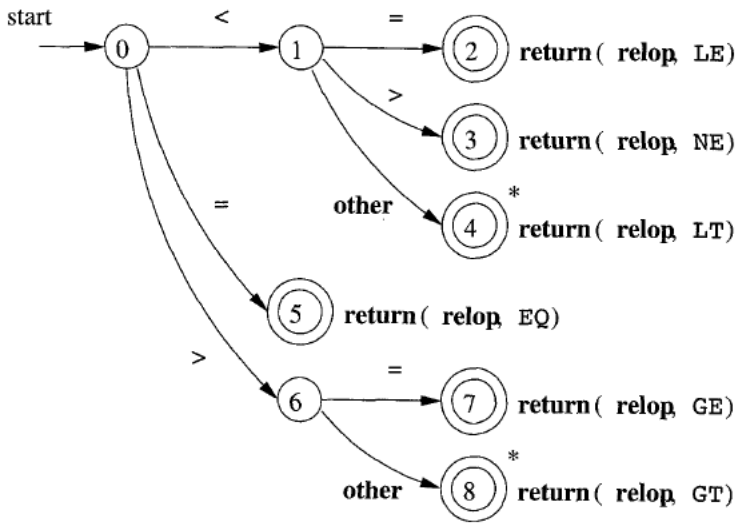
1. **Identifier:** Identifier is an entity which starts from a letter or underscore and then it can contain both letters and digit. Any other special character is not allowed in the identifier.



Letter → [A-Z a-z]
Digit → [0-9]

Write a program to detect whether the entered string is an identifier or not ; based on this finite automata.

2. The following diagram is a transition diagram of relational operators.



Write a program to detect all the relational operators present in a source program.

Assignment:

- Write a program in C that reads any simple C program as source and separates out the valid tokens from the source program like: keywords, identifiers, operators, separators , parenthesis and special symbols , also detects the preprocessor directives .

Useful Functions:

Introduction

The **ctype.h** header file of the C Standard Library provides declares several functions useful for testing and mapping characters. All the functions accepts **int** as a parameter, whose value must be EOF or representable as an unsigned char.

All the functions return non-zero (true) if the argument c satisfies the condition described, and zero if not.

Library Functions

Following are the functions defined in the header ctype.h:

S.N.	Function & Description
1	int isalnum(int c) This function check whether the passed character is alphanumeric.
2	int isalpha(int c) This function check whether the passed character is alphabetic.
3	int iscntrl(int c) This function check whether the passed character is control character.
4	int isdigit(int c) This function check whether the passed character is decimal digit.
5	int isgraph(int c) This function check whether the passed character has graphical representation using locale.
6	int islower(int c) This function check whether the passed character is lowercase letter.
7	int isprint(int c) This function check whether the passed character is printable .
8	int ispunct(int c) This function check whether the passed character is punctuation character.
9	int isspace(int c) This function check whether the passed character is white-space.
10	int isupper(int c) This function check whether the passed character is uppercase letter.
11	int isxdigit(int c) This function check whether the passed character is hexadecimal digit.

The library also contains two conversion functions that also accept and return an "int"

S.N.	Function & Description
1	int tolower(int c) This function convert uppercase letter to lowercase.
2	int toupper(int c) This function convert lowercase letter to uppercase.

Character Classes

S.N.	Character Class & Description
1	**Digits** A set of whole numbers { 0, 1, 2, 3, 4, 5, 6, 7, 8, 9 }
2	**Hexadecimal digits** This is the set of { 0 1 2 3 4 5 6 7 8 9 A B C D E F a b c d e f }
3	**Lowercase letters** This is a set of { a b c d e f g h i j k l m n o p q r s t u v w x y z }
4	**Uppercase letters** A set of whole numbers {A B C D E F G H I J K L M N O P Q R S T U V W X Y Z }
5	**Letters** This is a set of lowercase letters and uppercase letters
6	**Alphanumeric characters** This is a set of Digits, Lowercase letters and Uppercase letters
7	**Punctuation characters** This is a set of ! " # $ % & ' ( ) * + , - . / : ; < = > ? @ [\ ] ^ _ ` { | } ~
8	**Graphical characters** This is a set of Alphanumeric characters and Punctuation characters.
9	**Space characters** This is a set of tab, newline, vertical tab, form feed, carriage return, and space.
10	**Printable characters** This is a set of Alphanumeric characters, Punctuation characters and Space characters.
11	**Control characters** In ASCII, these characters have octal codes 000 through 037, and 177 (DEL).
12	**Blank characters** These are space and tab.
13	**Alphabetic characters** This is a set of Lowercase letters and Uppercase letters.

Description

The C library function **void isdigit(int c)** checks if the passed character is a decimal digit character.

Decimal digits are(numbers): 0 1 2 3 4 5 6 7 8 9

Declaration

Following is the declaration for isdigit() function.

```
int isdigit(int c);
```

Parameters

- **c** -- This is the character to be checked.

Return Value

This function returns nonzero value if c is a digit, else 0

Example

The following example shows the usage of isdigit() function.

```
#include <stdio.h>
#include <ctype.h>
#include <conio.h>


int main()
{
   int var1 = 'h';
   int var2 = '2';

   if( isdigit(var1) )
   {
      printf("var1 = |%c| is a digit\n", var1 );
   }
   else
   {
      printf("var1 = |%c| is not a digit\n", var1 );
   }
   if( isdigit(var2) )
   {
      printf("var2 = |%c| is a digit\n", var2 );
   }
   else
   {
      printf("var2 = |%c| is not a digit\n", var2 );
   }

   getch();
   return(0);
}
```

Let us compile and run the above program, this will produce the following result:

```
var1 = |h| is not a digit
var2 = |2| is a digit
```