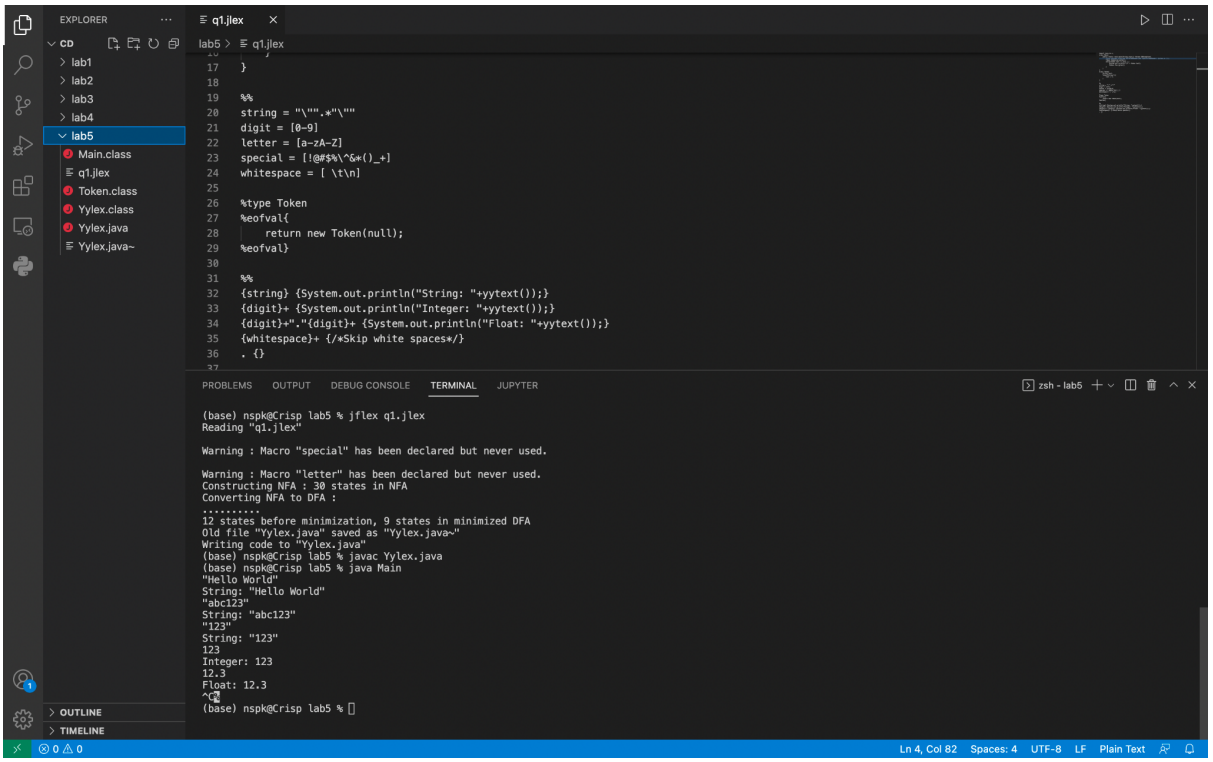
19CSE401 - Compiler Design

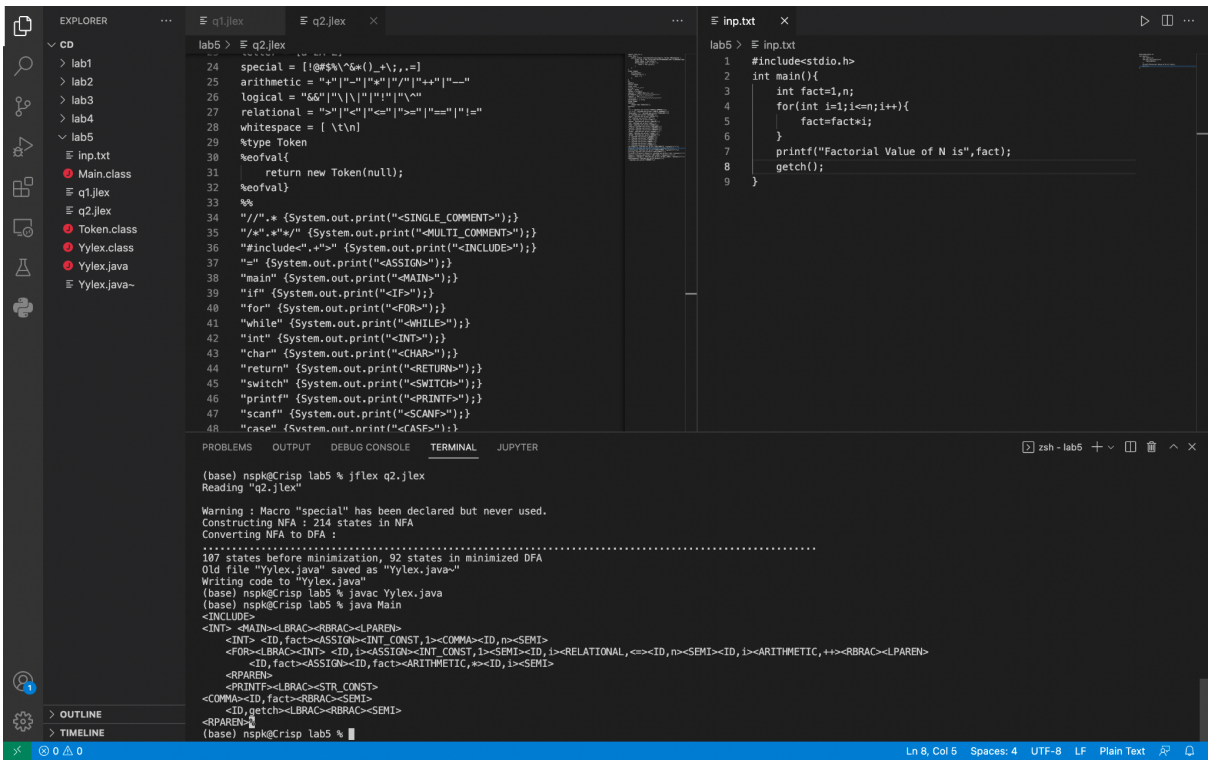
JLex Programming

Lab sheet-5

Done By:
N Sai Pavan Krishna
AM.EN.U4CSE19347

1. Create a lexical generator in JLex to identify the following tokens from the input given in terminal
    a. **String** : set of characters enclosed in “---”. Example: “amrita” , “amma123”, etc
    b. **Integer** : set of numbers
    c. **Float** : Ex. 0.34, 12.43, 12.0 etc





2. Consider the following token

Token	Lexemes		Token	Lexemes
MAIN	main		PRINTF	printf
LPAREN	{		SCANF	scanf
RPAREN	}		RETURN	return
LBRACE	(		INT	int
RBRACE	)		FLOAT	float
ID	For all identifiers		CHAR	char
NUM	Integer constants		/* - -- */	Multiline comment
STR	String Constant		//	Single line comment
REAL	Floating-point constants		SEMI	;
IF	If		COMMA	,
WHILE	while		ARITHMETIC	+, - ,*,/, ++, --
SWITCH	switch		LOGIC	&&, ||, !
CASE	case		RELATIONAL	<, <= , >, >= ==, !=
BREAK	break			

Write a Jlex program that generates the token of the form ***<Token, lexeme>*** except for keywords, for the given program. If none of the patterns matches for a lexeme, given an error statement specifying the line number in the program.

```
int main()
{
  int c, n, f = 1;

  printf("Enter a number to calculate its factorial\n");
  scanf("%d", &n);

  for (c = 1; c <= n; c++)
    f = f * c;

  printf("Factorial of %d = %d\n", n, f);

  return 0;
}
```