LAB REPORT

CSE332: Compiler Design Lab

01

[Report Number]

Topic: Write a Flex program to recognize the following types of strings.

- Starts with a letter or underscore
- Followed by any number of letters, digits, or underscores
- Must end with a digit

Submitted To

Mushfiqur Chowdhury

Lecturer

Department of CSE, Daffodil International University

Submitted By

Student ID: 0242220005101711

Section: 63N1

Student Name: Md Tausif Bin Mozid

Date of Assignment Distribution: 10 November 2025

Date of Assignment Submission: 13 November 2025

Experiment No: 01		Mapping: CO1 and CO2
Experiment Name	Write a Flex program to recognize the following types of strings. ❖ Starts with a letter or underscore ❖ Followed by any number of letters, digits, or underscores ❖ Must end with a digit	

Experiment Details:

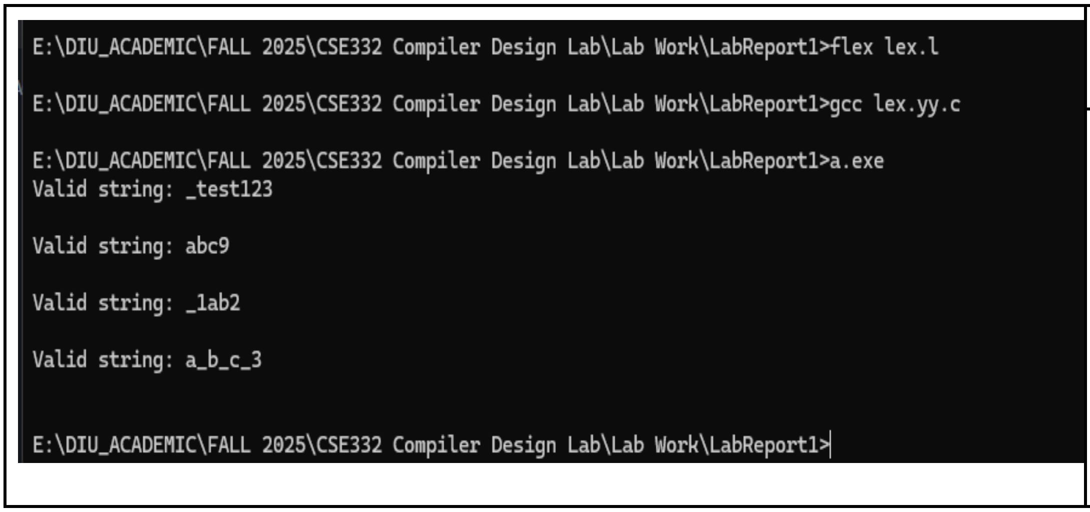
Lex Code:

```
/*Definaiton Section*/
%{
#include <stdio.h>

%}

/* Rules Section */
%%
[A-Za-z_][A-Za-z0-9_]*[0-9]   {
    printf("Valid string: %s\n", yytext);
}

. {}
%%

/* User code section */
int main() {
    yyin = fopen("input.c", "r");
    yylex();
    return 0;
}

int yywrap() {
    return 1;
}
```

Input.c File:

```
_test123
abc9
_1ab2
a_b_c_3
test_ hello
```

Obtained Output:

	Desired Output?
	YES

```
E:\DIU_ACADEMIC\FALL 2025\CSE332 Compiler Design Lab\Lab Work\LabReport1>flex lex.l

E:\DIU_ACADEMIC\FALL 2025\CSE332 Compiler Design Lab\Lab Work\LabReport1>gcc lex.yy.c

E:\DIU_ACADEMIC\FALL 2025\CSE332 Compiler Design Lab\Lab Work\LabReport1>a.exe
Valid string: _test123

Valid string: abc9

Valid string: _1ab2

Valid string: a_b_c_3


E:\DIU_ACADEMIC\FALL 2025\CSE332 Compiler Design Lab\Lab Work\LabReport1>
```

Observation/ Comments:

a) The program correctly identifies strings that start with a letter or underscore, contain letters/digits/underscores, and end with a digit.

b) Inputs like _test123, abc9, and a_b_c_3 are recognized as valid.

c) Invalid inputs such as 9abc or test_ are ignored.

d) The program uses regular expressions and Flex rules effectively for lexical analysis.

e) It reads from input.c and terminates properly using yywrap().

Appendix A: Course Outcomes, Complex Engineering Problems (EP) and Complex Engineering Activities (EA) Addressing.

Table: CSE312 Course Outcomes (COs) with Mappings

COs	CO Statements	POs	Learning Domains	Knowledge Profile	Complex Engineering Problem	Complex Engineering Activities
CO1	Demonstrate a comprehensive understanding of fundamental database management concepts, including the relational data model, normalization techniques, and SQL basics.	PO1	C2 A2 P2	K2 K3 K4 K8	EP1 EP4	
CO2	Design, implement and optimize relational databases, incorporating advanced SQL queries, indexing techniques and query optimization strategies.	PO3	C3 A3 P3	K2 K3 K4 K6 K8	EP1 EP2 EP7	EA3
CO3	Understand and Analyze security measures, distributed database architectures and emerging trends in database management, demonstrating an understanding of the broader context and challenges in the field.	PO5	C4 A4 P3	K6	EP4	

Table: Addressing CO (1 to 3), Knowledge Profile (K), Attainment of Complex Engineering Problems (EP):

SN	Engineering Problem (EP) Definition	Attain Ment	CO	Justification (with Knowledge Profile)
01	EP1: Depth of Knowledge required	Yes/No	CO1, CO2	
02	EP2: Range of Conflicting Requirements	Yes/No	CO2	

03	EP4: Familiarity of Issues	Yes/No	CO1, CO3	
04	EP7: Interdependence	Yes/No	CO2	

Table: Addressing CO’s

SN	COs	Attainment	Justification
01	CO1	Yes/No	These Lab activities attain CO1 by ..
02	CO2	Yes/No	N/A
03	CO3	Yes/No	These Lab activities attain CO3 by ..