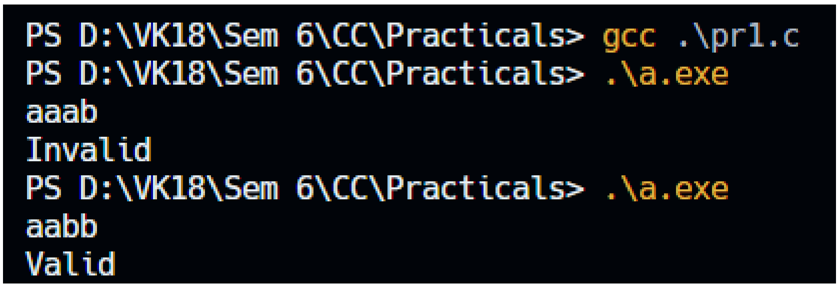
Practical : 1

AIM :

Output :



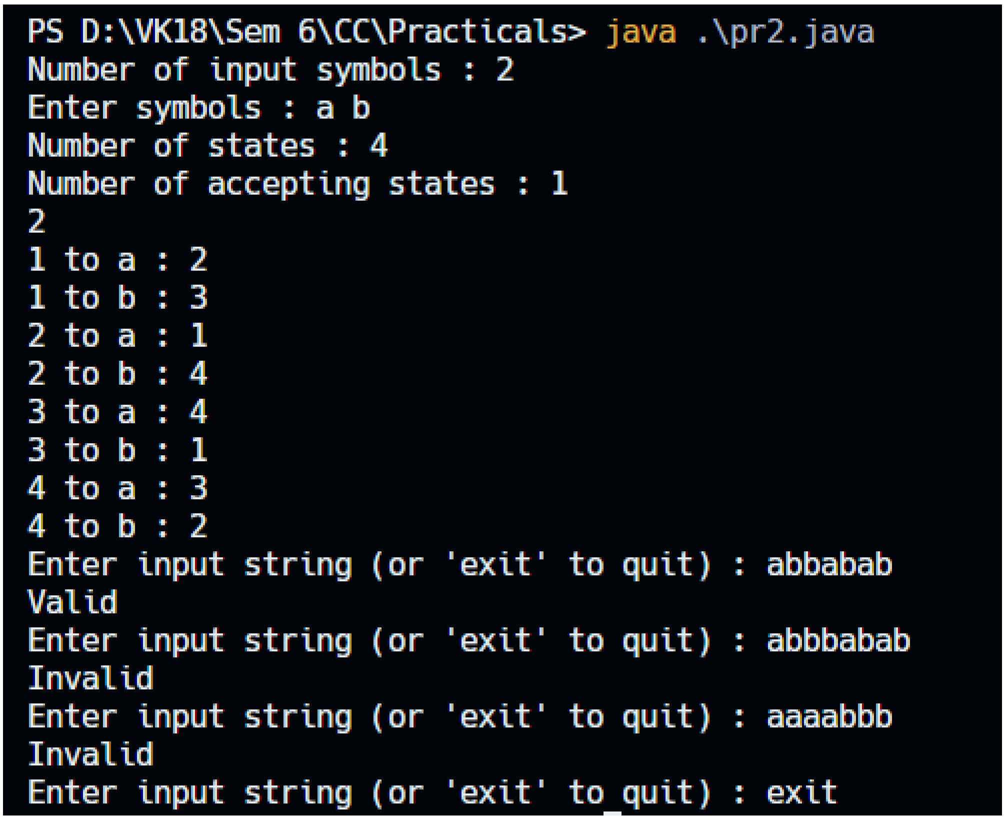
```
PS D:\VK18\Sem 6\CC\Practicals> gcc .\pr1.c
PS D:\VK18\Sem 6\CC\Practicals> .\a.exe
aaab
Invalid
PS D:\VK18\Sem 6\CC\Practicals> .\a.exe
aabb
Valid
```

Conclusion :

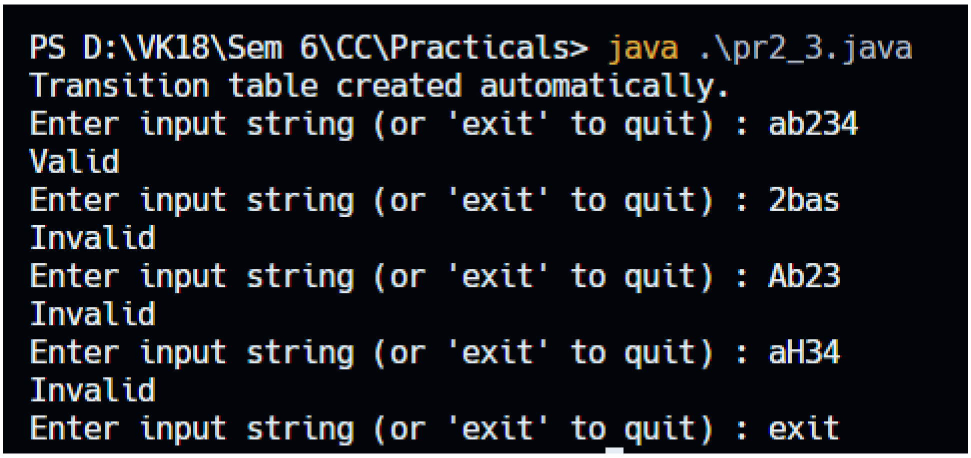
Practical : 2

AIM :

Output :



```
PS D:\VK18\Sem 6\CC\Practicals> java .\pr2.java
Number of input symbols : 2
Enter symbols : a b
Number of states : 4
Number of accepting states : 1
2
1 to a : 2
1 to b : 3
2 to a : 1
2 to b : 4
3 to a : 4
3 to b : 1
4 to a : 3
4 to b : 2
Enter input string (or 'exit' to quit) : abbabab
Valid
Enter input string (or 'exit' to quit) : abbbabab
Invalid
Enter input string (or 'exit' to quit) : aaaabbb
Invalid
Enter input string (or 'exit' to quit) : exit
```



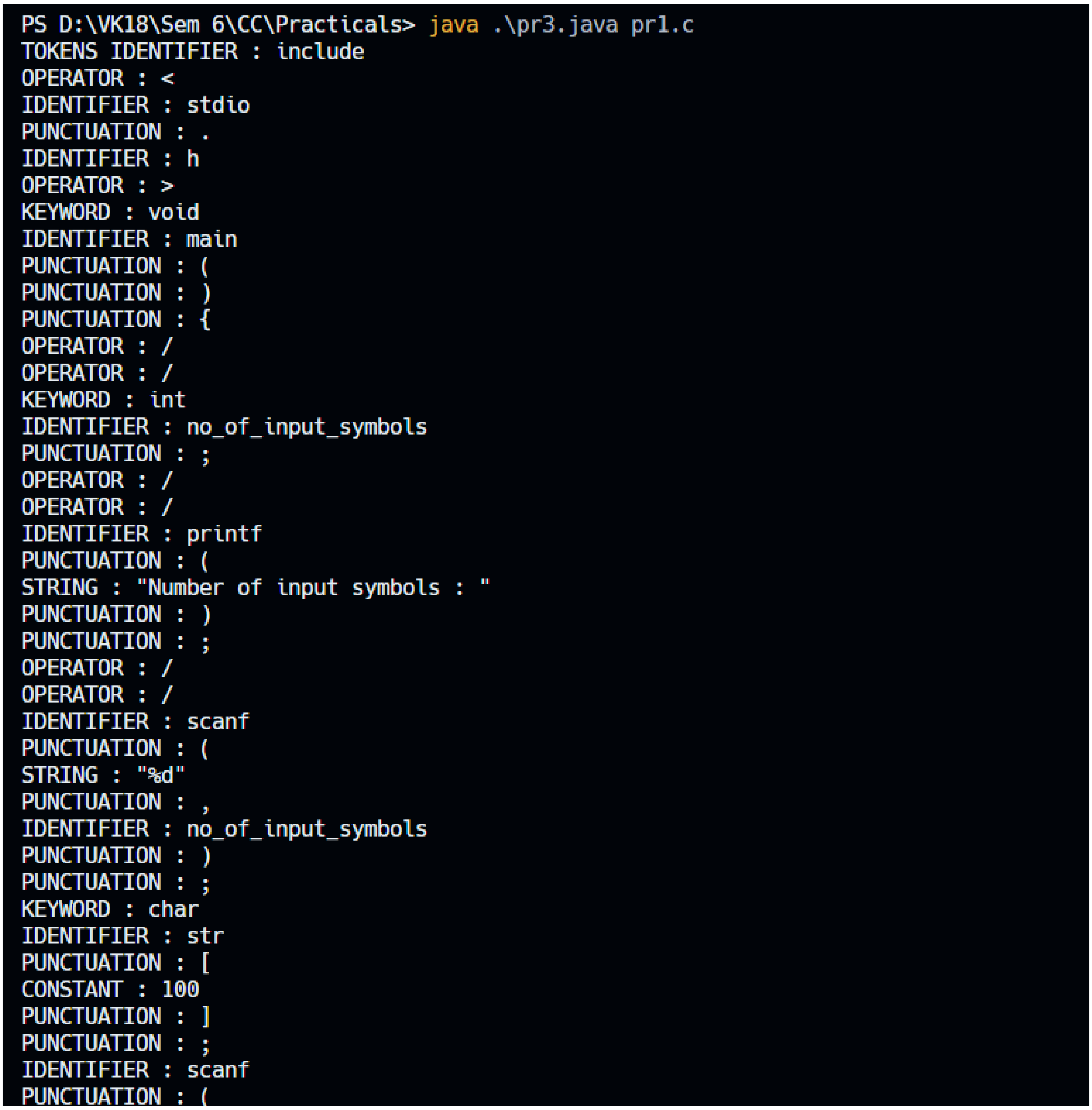
```
PS D:\VK18\Sem 6\CC\Practicals> java .\pr2_3.java
Transition table created automatically.
Enter input string (or 'exit' to quit) : ab234
Valid
Enter input string (or 'exit' to quit) : 2bas
Invalid
Enter input string (or 'exit' to quit) : Ab23
Invalid
Enter input string (or 'exit' to quit) : aH34
Invalid
Enter input string (or 'exit' to quit) : exit
```

Conclusion :

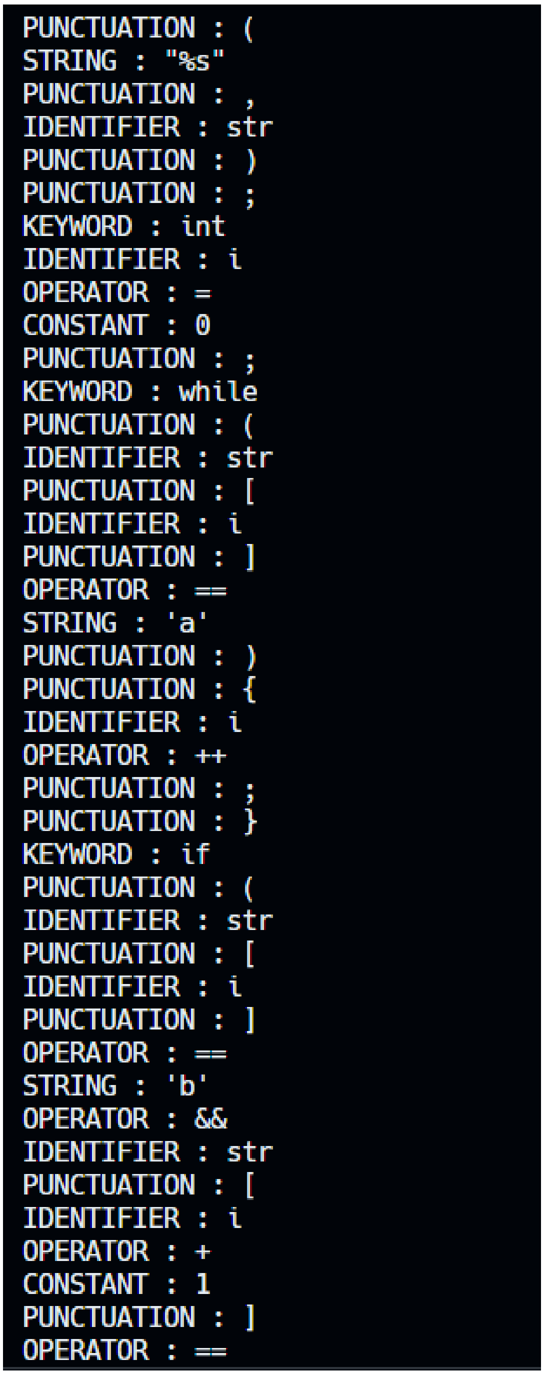
Practical : 3

AIM :

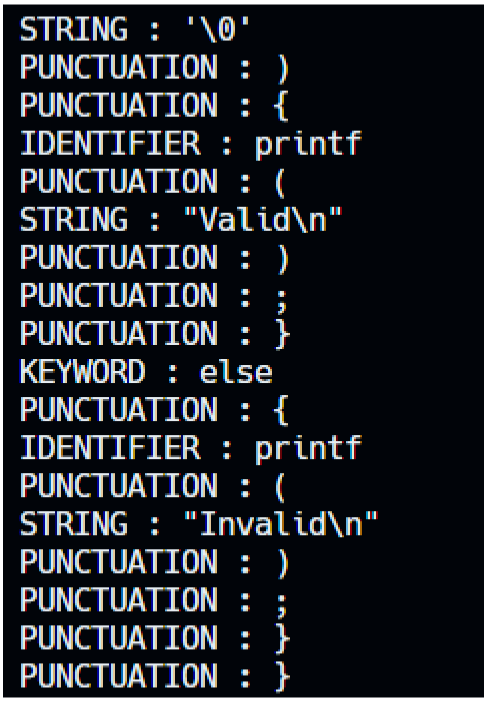
Output :



```
PS D:\VK18\Sem 6\CC\Practicals> java .\pr3.java pr1.c
TOKENS IDENTIFIER : include
OPERATOR : <
IDENTIFIER : stdio
PUNCTUATION : .
IDENTIFIER : h
OPERATOR : >
KEYWORD : void
IDENTIFIER : main
PUNCTUATION : (
PUNCTUATION : )
PUNCTUATION : {
OPERATOR : /
OPERATOR : /
KEYWORD : int
IDENTIFIER : no_of_input_symbols
PUNCTUATION : ;
OPERATOR : /
OPERATOR : /
IDENTIFIER : printf
PUNCTUATION : (
STRING : "Number of input symbols : "
PUNCTUATION : )
PUNCTUATION : ;
OPERATOR : /
OPERATOR : /
IDENTIFIER : scanf
PUNCTUATION : (
STRING : "%d"
PUNCTUATION : ,
IDENTIFIER : no_of_input_symbols
PUNCTUATION : )
PUNCTUATION : ;
KEYWORD : char
IDENTIFIER : str
PUNCTUATION : [
CONSTANT : 100
PUNCTUATION : ]
PUNCTUATION : ;
IDENTIFIER : scanf
PUNCTUATION : (
```



```
PUNCTUATION : (
STRING : "%s"
PUNCTUATION : ,
IDENTIFIER : str
PUNCTUATION : )
PUNCTUATION : ;
KEYWORD : int
IDENTIFIER : i
OPERATOR : =
CONSTANT : 0
PUNCTUATION : ;
KEYWORD : while
PUNCTUATION : (
IDENTIFIER : str
PUNCTUATION : [
IDENTIFIER : i
PUNCTUATION : ]
OPERATOR : ==
STRING : 'a'
PUNCTUATION : )
PUNCTUATION : {
IDENTIFIER : i
OPERATOR : ++
PUNCTUATION : ;
PUNCTUATION : }
KEYWORD : if
PUNCTUATION : (
IDENTIFIER : str
PUNCTUATION : [
IDENTIFIER : i
PUNCTUATION : ]
OPERATOR : ==
STRING : 'b'
OPERATOR : &&
IDENTIFIER : str
PUNCTUATION : [
IDENTIFIER : i
OPERATOR : +
CONSTANT : 1
PUNCTUATION : ]
OPERATOR : ==
```



```
STRING : '\0'
PUNCTUATION : )
PUNCTUATION : {
IDENTIFIER : printf
PUNCTUATION : (
STRING : "Valid\n"
PUNCTUATION : )
PUNCTUATION : ;
PUNCTUATION : }
KEYWORD : else
PUNCTUATION : {
IDENTIFIER : printf
PUNCTUATION : (
STRING : "Invalid\n"
PUNCTUATION : )
PUNCTUATION : ;
PUNCTUATION : }
PUNCTUATION : }
```

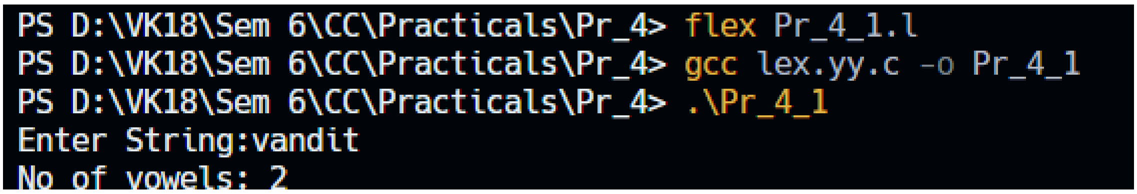
Conclusion :

Practical : 4

AIM :

Objective-1:

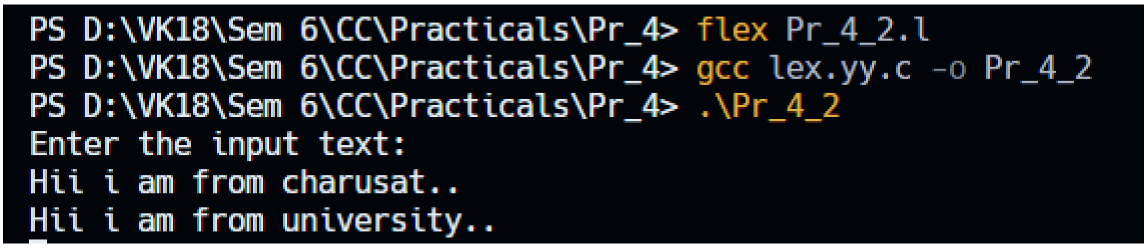
Output :



```
PS D:\VK18\Sem 6\CC\Practicals\Pr_4> flex Pr_4_1.l
PS D:\VK18\Sem 6\CC\Practicals\Pr_4> gcc lex.yy.c -o Pr_4_1
PS D:\VK18\Sem 6\CC\Practicals\Pr_4> .\Pr_4_1
Enter String:vandit
No of vowels: 2
```

Objective-2 :

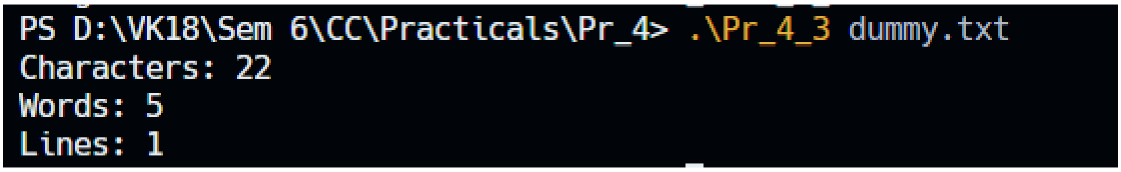
Output :



```
PS D:\VK18\Sem 6\CC\Practicals\Pr_4> flex Pr_4_2.l
PS D:\VK18\Sem 6\CC\Practicals\Pr_4> gcc lex.yy.c -o Pr_4_2
PS D:\VK18\Sem 6\CC\Practicals\Pr_4> .\Pr_4_2
Enter the input text:
Hii i am from charusat..
Hii i am from university..
```

Objective-3 :

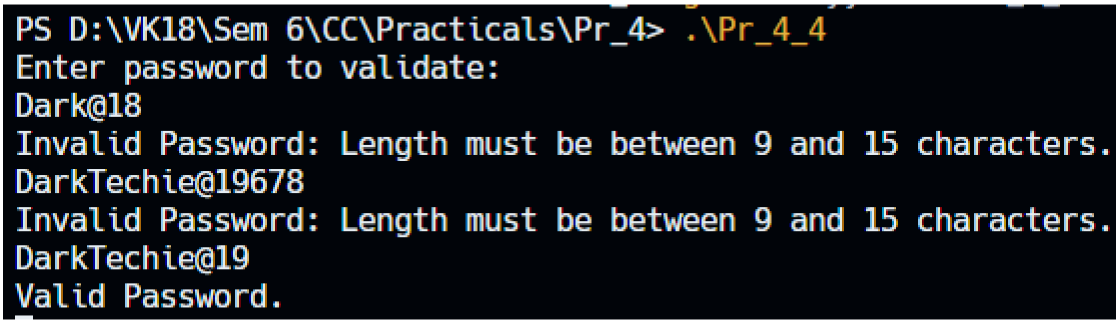
Output :



```
PS D:\VK18\Sem 6\CC\Practicals\Pr_4> .\Pr_4_3 dummy.txt
Characters: 22
Words: 5
Lines: 1
```

Objective-4 :

Output :



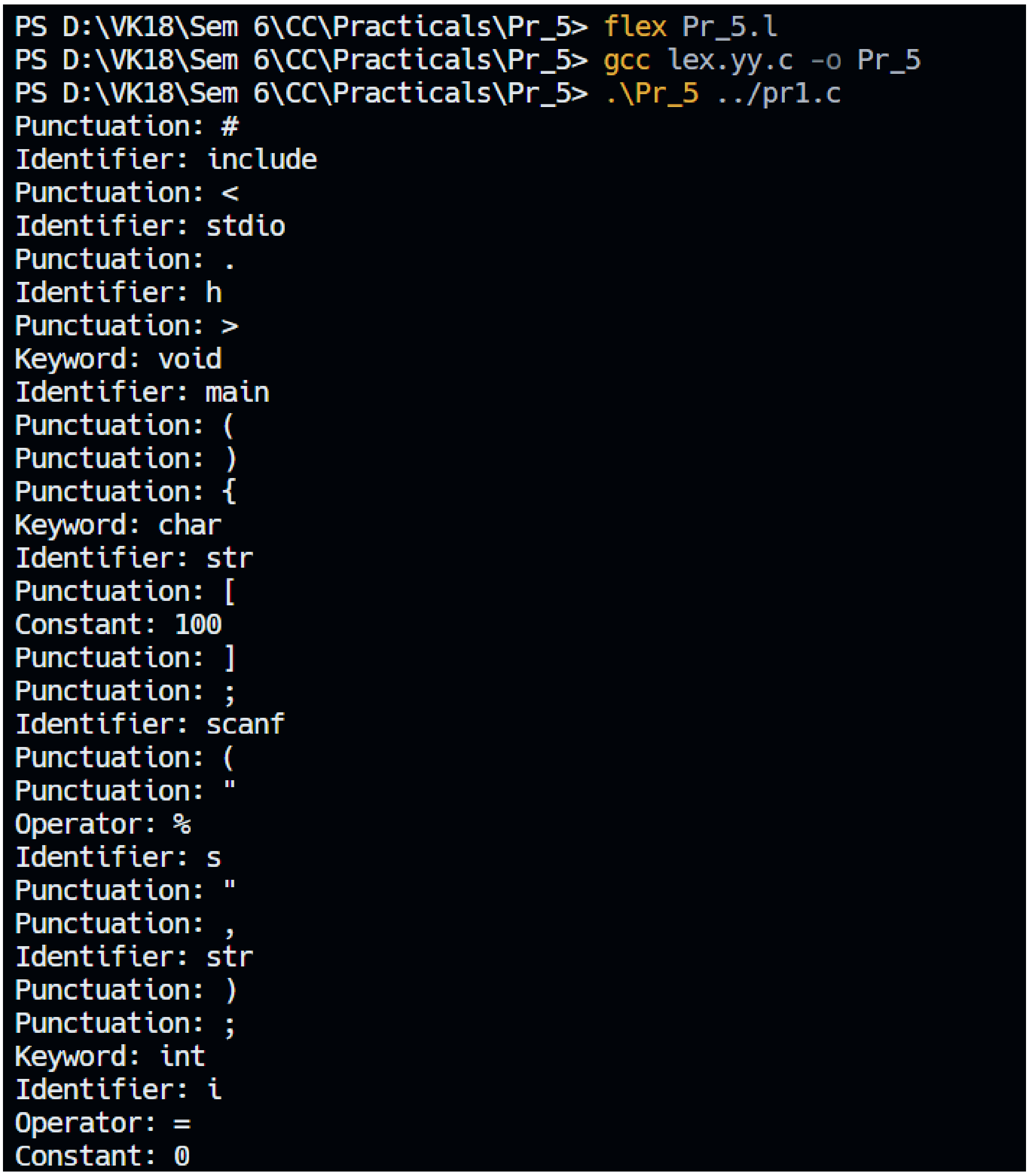
```
PS D:\VK18\Sem 6\CC\Practicals\Pr_4> .\Pr_4_4
Enter password to validate:
Dark@18
Invalid Password: Length must be between 9 and 15 characters.
DarkTechie@19678
Invalid Password: Length must be between 9 and 15 characters.
DarkTechie@19
Valid Password.
```

Conclusion :

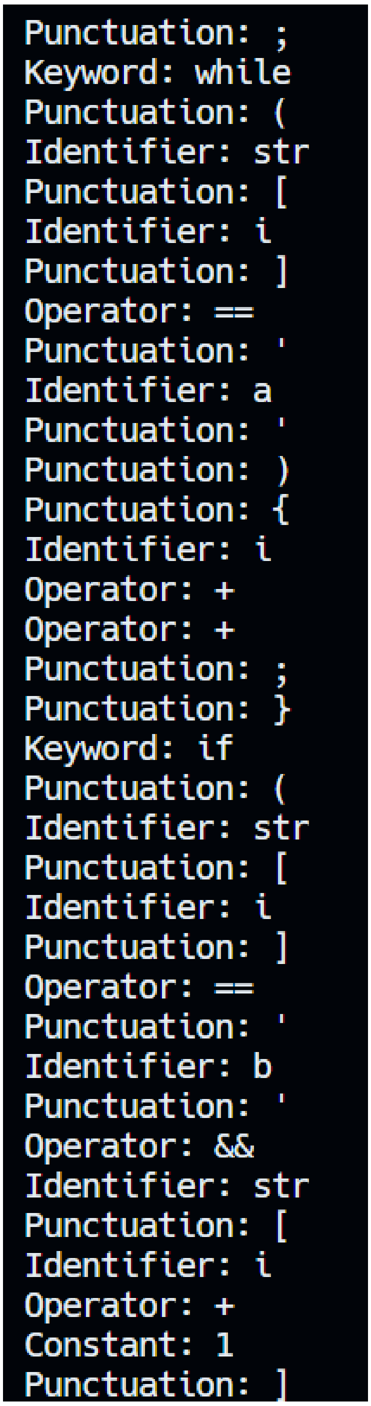
Practical : 5

AIM :

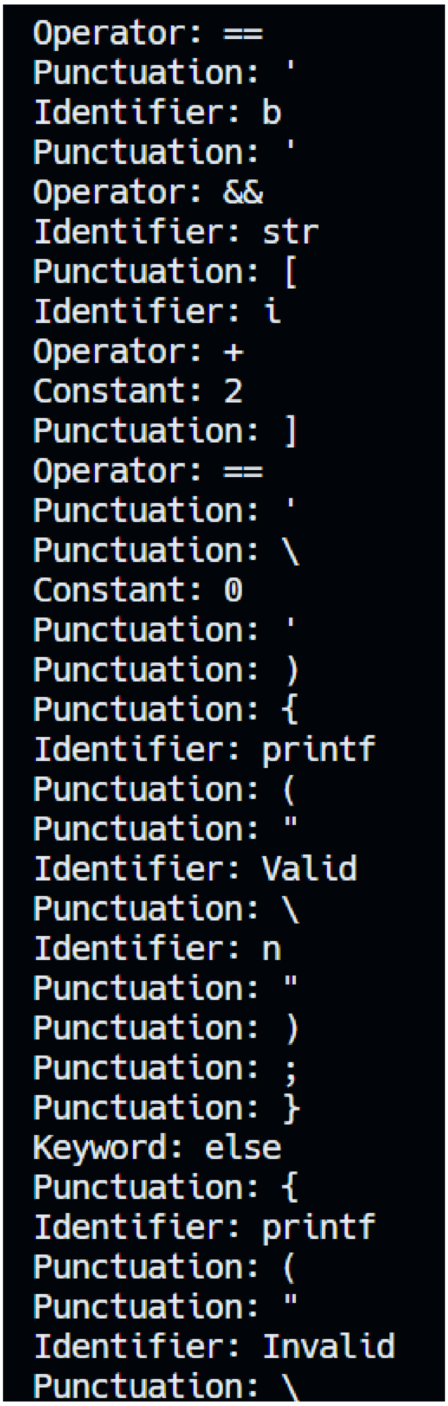
Output :



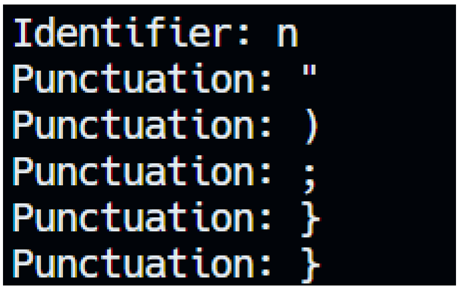
```
PS D:\VK18\Sem 6\CC\Practicals\Pr_5> flex Pr_5.l
PS D:\VK18\Sem 6\CC\Practicals\Pr_5> gcc lex.yy.c -o Pr_5
PS D:\VK18\Sem 6\CC\Practicals\Pr_5> .\Pr_5 ../pr1.c
Punctuation: #
Identifier: include
Punctuation: <
Identifier: stdio
Punctuation: .
Identifier: h
Punctuation: >
Keyword: void
Identifier: main
Punctuation: (
Punctuation: )
Punctuation: {
Keyword: char
Identifier: str
Punctuation: [
Constant: 100
Punctuation: ]
Punctuation: ;
Identifier: scanf
Punctuation: (
Punctuation: "
Operator: %
Identifier: s
Punctuation: "
Punctuation: ,
Identifier: str
Punctuation: )
Punctuation: ;
Keyword: int
Identifier: i
Operator: =
Constant: 0
```



```
Punctuation: ;
Keyword: while
Punctuation: (
Identifier: str
Punctuation: [
Identifier: i
Punctuation: ]
Operator: ==
Punctuation: '
Identifier: a
Punctuation: '
Punctuation: )
Punctuation: {
Identifier: i
Operator: +
Operator: +
Punctuation: ;
Punctuation: }
Keyword: if
Punctuation: (
Identifier: str
Punctuation: [
Identifier: i
Punctuation: ]
Operator: ==
Punctuation: '
Identifier: b
Punctuation: '
Operator: &&
Identifier: str
Punctuation: [
Identifier: i
Operator: +
Constant: 1
Punctuation: ]
```



```
Operator: ==
Punctuation: '
Identifier: b
Punctuation: '
Operator: &&
Identifier: str
Punctuation: [
Identifier: i
Operator: +
Constant: 2
Punctuation: ]
Operator: ==
Punctuation: '
Punctuation: \
Constant: 0
Punctuation: '
Punctuation: )
Punctuation: {
Identifier: printf
Punctuation: (
Punctuation: "
Identifier: Valid
Punctuation: \
Identifier: n
Punctuation: "
Punctuation: )
Punctuation: ;
Punctuation: }
Keyword: else
Punctuation: {
Identifier: printf
Punctuation: (
Punctuation: "
Identifier: Invalid
Punctuation: \
```



```
Identifier: n
Punctuation: "
Punctuation: )
Punctuation: ;
Punctuation: }
Punctuation: }
```

Conclusion :