

test\valid identifier.y

```
1  LEX PART:
2  %{
3  #include "y.tab.h"
4  %}
5
6  %%
7  [0-9] {return DIGIT;}
8  [a-zA-Z] {return ALPHA;}
9  \n {return 0;}
10 . {return yytext[0];}
11 %%
12
13 int yywrap()
14 {
15     return 1;
16 }
17
18
19 YACC PART:
20 %{
21 #include<stdio.h>
22 #include<stdlib.h>
23 int yylex();
24 %}
25
26 %token DIGIT ALPHA
27
28 %%
29 var : ALPHA
30 |var ALPHA
31 |var DIGIT;
32 %%
33
34 int main(int argc, char * argv[])
35 {
36     printf("Enter an Identifier:");
37     yyparse();
38     printf("\nValid Identifier\n");
39     return 0;
40 }
41
42 int yyerror()
43 {
44     printf("\nInvalid Identifier\n");
45     exit(1);
46 }
47
48
49
50
51
```

```
52 ALGORITHM:
53 1. Start
54
55 2. Lex Part (Tokenization):
56     2.1 Define tokens based on lexical rules:
57         2.1.1 [0-9]: Matches digits; return the DIGIT token
58         2.1.2 [a-zA-Z]: Matches alphabetic characters; return the ALPHA token
59         2.1.3 \n: Matches newline and returns 0 to indicate the end of input
60         2.1.4 . : Matches any other character and returns it
61     2.2 Define yywrap() function:
62         2.2.1 Called when the input is exhausted
63         2.2.2 Return 1 to indicate EOF
64     2.3 Output tokens to YACC for parsing
65
66 3. YACC Part (Parsing):
67     3.1 Include necessary libraries for handling parsing logic
68     3.2 Declare tokens received from Lex (DIGIT, ALPHA)
69     3.3 Define grammar rules for parsing:
70         3.3.1 var: Starting rule for identifier:
71             - var : ALPHA: Matches a single alphabetic character
72             - var : var ALPHA: Matches an alphabetic sequence
73             - var : var DIGIT: Allows identifiers to end with digits after an initial
74 alphabetic character
75     3.4 Implement yyerror() function:
76         3.4.1 Print "Invalid Identifier" and exit when parsing fails
77
78 4. main() Function:
79     4.1 Prompt the user to enter an identifier
80     4.2 Invoke yyparse() to begin parsing the input
81     4.3 If parsing is successful, print "Valid Identifier"
82
83 5. Stop
```