

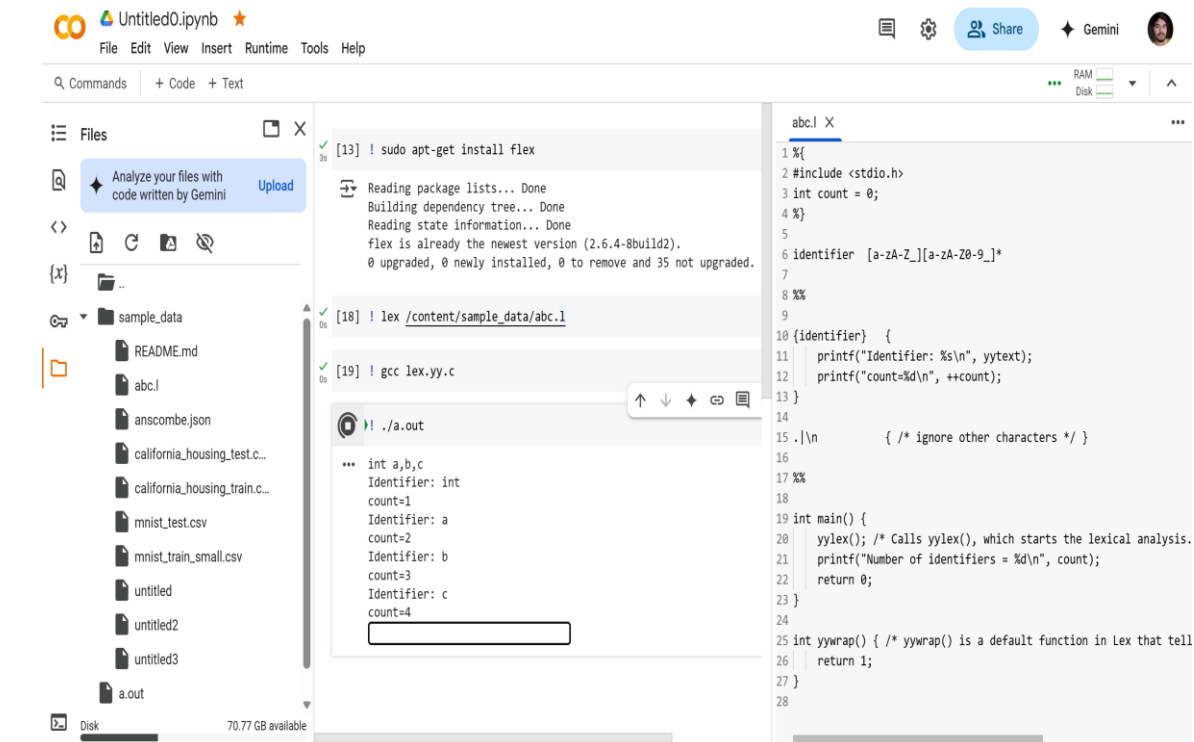
2200290120001

Aaditiya Tyagi

CS6A

Compiler Design lab 3 : Lex tool on Google collab and Linux terminal

To identify the identifiers:



The screenshot displays a Google Colab notebook titled "Untitled0.ipynb". The interface includes a file explorer on the left, a terminal window in the center, and a code editor on the right.

File Explorer: Shows a directory structure with files like README.md, abc.l, anscombe.json, california_housing_test.c..., california_housing_train.c..., mnist_test.csv, mnist_train_small.csv, untitled, untitled2, untitled3, and a.out. The disk usage is shown as 70.77 GB available.

Terminal Window: Contains the following commands and output:

```
[13] ! sudo apt-get install flex
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
flex is already the newest version (2.6.4-8build2).
0 upgraded, 0 newly installed, 0 to remove and 35 not upgraded.

[18] ! lex /content/sample_data/abc.l

[19] ! gcc lex.yy.c
./a.out
... int a,b,c
Identifier: int
count=1
Identifier: a
count=2
Identifier: b
count=3
Identifier: c
count=4
```

Code Editor: Shows the content of the file abc.l, which is a Lex program:

```
1 %{
2 #include <stdio.h>
3 int count = 0;
4 %}
5
6 identifier [a-zA-Z][a-zA-Z0-9]*
7
8 %%
9
10 {identifier} {
11     printf("Identifier: %s\n", ytext);
12     printf("count=%d\n", ++count);
13 }
14
15 .|\n          { /* ignore other characters */ }
16
17 %%
18
19 int main() {
20     yylex(); /* Calls yylex(), which starts the lexical analysis.
21     printf("Number of identifiers = %d\n", count);
22     return 0;
23 }
24
25 int yywrap() { /* yywrap() is a default function in Lex that tell
26     return 1;
27 }
28
```

```
Open [ ] +l
*2.lex x
1 %{
2 #include<stdio.h>
3 int count =0;
4 %}
5 identifier [a-zA-Z][a-zA-Z0-9]*
6 operator [+/*=]
7 number [0-9]+
8
9 %%
10 {identifier} {printf("identifier: %s\n", yytext);
11 printf("count=%d",++count);}
12 {operator} {printf("operator: %s\n", yytext);}
13 {number} {printf("number : %s\n", yytext);}
14
15 .|\n
16
17 %%
18 int main(){
19     yylex();
20     printf("numbers of identifier =%d\n ",count );
21     return 0;
22 }
23 int yywrap()
24 {
25     return 1;
26 }
```

```
Reading package lists... Done
student@student-HP-Pro-Tower-280-G9-PCI-Desktop-PC:~$ sudo apt-get install lex
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
E: Unable to locate package lex
student@student-HP-Pro-Tower-280-G9-PCI-Desktop-PC:~$ sudo apt-get install gedit
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
gedit is already the newest version (46.2-2).
0 upgraded, 0 newly installed, 0 to remove and 318 not upgraded.
student@student-HP-Pro-Tower-280-G9-PCI-Desktop-PC:~$
student@student-HP-Pro-Tower-280-G9-PCI-Desktop-PC:~$
student@student-HP-Pro-Tower-280-G9-PCI-Desktop-PC:~$ ! gcc lex.yy.c
student@student-HP-Pro-Tower-280-G9-PCI-Desktop-PC:~$ ! ./a.out
int a , b , c
identifier: int
count=1identifier: a
count=2operator: ,
identifier: b
count=3operator: ,
identifier: c
count=4
```

To identify the Keywords:

```

1%{
2#include <stdio.h>
3#include <string.h>
4
5char *keywords[] = {
6    "int", "float", "if", "else", "while", "return", "for", "void", NULL
7};
8
9int count = 0;
10
11int is_keyword(char *str) {
12    for (int i = 0; keywords[i] != NULL; i++) {
13        if (strcmp(str, keywords[i]) == 0) {
14            return 1;
15        }
16    }
17    return 0;
18}
19}%
20
21%%
22
23[0-9a-zA-Z_][0-9a-zA-Z_]* {
24    if (is_keyword(yytext)) {
25        printf("Keyword: %s\n", yytext);
26        count++;
27    }
28}

```

```

32%%
33
34int main() {
35    yylex();
36    printf("Number of keywords: %d\n", count);
37    return 0;
38}
39
40int yywrap() {
41    return 1;
42}
43

```

unction it appears in

```

student@student-HP-280-G3-SFF-Business-PC: $ lex akash.l
student@student-HP-280-G3-SFF-Business-PC: $ gcc lex.yy.c
student@student-HP-280-G3-SFF-Business-PC: $ ./a.out
int a, b, c;
Keywords : int
, , ;

```

