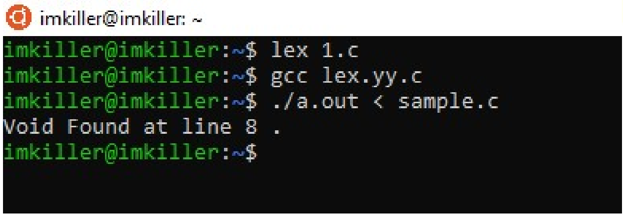

Sample.c

```
/*
 This is sample.c
*/
#include<stdio.h>

//This is a single line comment.

void fun()
{
       //Inside Fun
       printf("Hello Fun ");
}

//Driver Function
int main()
{
       printf("Hello world");
       fun();//Calling fun
       return 0;
}
```

Q9. WAP in lex to identify and display valid identifiers in a C program.

```
%{
       #include<stdio.h>
       int count=1;
%}

%%
\n {count++;}
void {printf("Void Found at line %d .\n",count);}
. {}
%%
int yywrap()
{
       return 1;
}
int main()
{
       yylex();
       return 0;
}
```

OUTPUT:

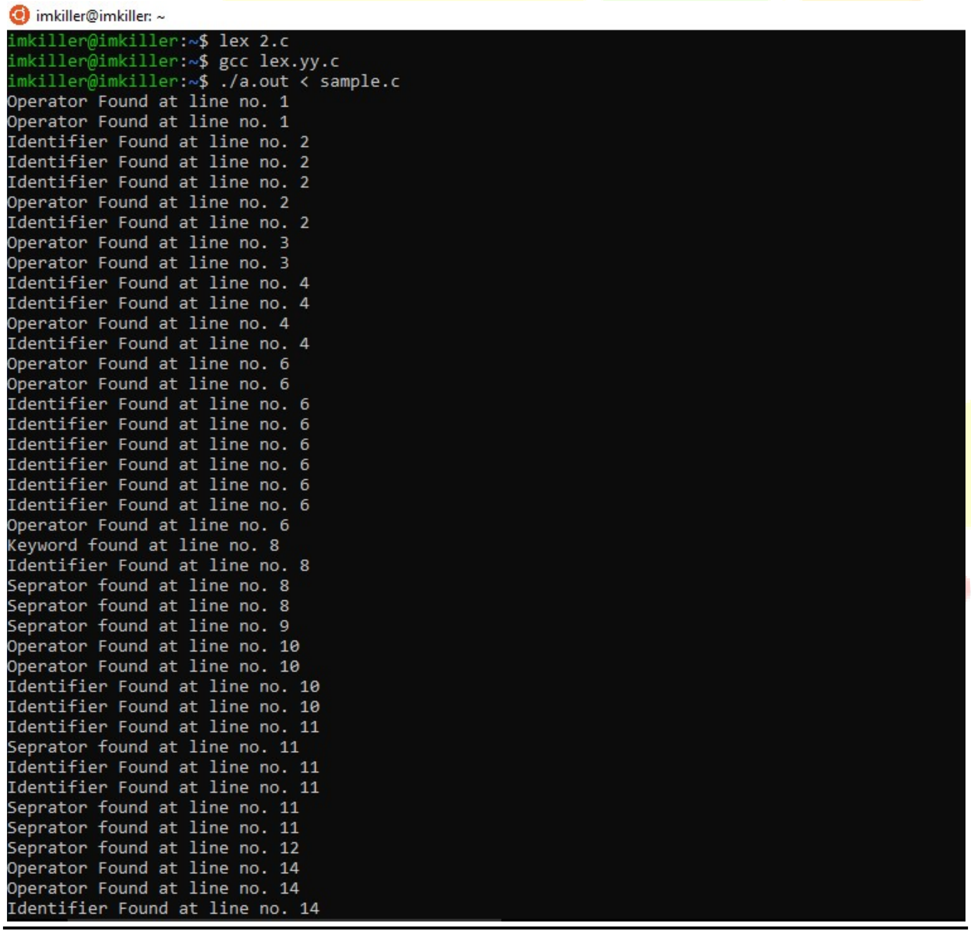


```
imkiller@imkiller:~$ lex 1.c
imkiller@imkiller:~$ gcc lex.yy.c
imkiller@imkiller:~$ ./a.out < sample.c
Void Found at line 8 .
imkiller@imkiller:~$
```

Q10. WAP in lex for identifying and priniting operators,keywords,identifiers and seprators.

```
%{
	#include<stdio.h>
	int count=1;
%}

%%
auto|double|int|struct|break|else|long|switch|case|enum|register|typedef|char|extern|return|union|continue|for|signed|void|do|if|static|w
hile|default|goto|sizeof|volatile|const|float|short|unsigned {printf("Keyword found at line no. %d\n",count);}
\n {count++;}
[{};,()] {printf("Seprator found at line no. %d\n",count);}
[+-/*=%] {printf("Operator Found at line no. %d\n",count);}
([a-zA-Z][0-9])+|[a-zA-Z]* {printf("Identifier Found at line no. %d\n",count);}
. {}
%%

int yywrap()
{
	return 1;
}

int main()
{
	yylex();
	return 0;
}
```

OUTPUT:

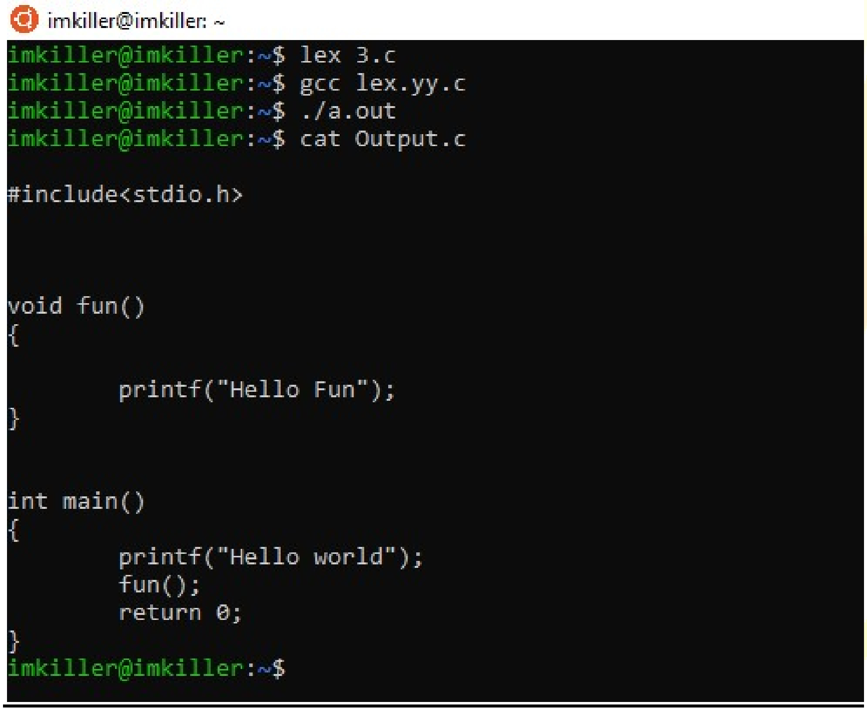


```
imkiller@imkiller:~$ lex 2.c
imkiller@imkiller:~$ gcc lex.yy.c
imkiller@imkiller:~$ ./a.out < sample.c
Operator Found at line no. 1
Operator Found at line no. 1
Identifier Found at line no. 2
Identifier Found at line no. 2
Identifier Found at line no. 2
Operator Found at line no. 2
Identifier Found at line no. 2
Operator Found at line no. 3
Operator Found at line no. 3
Identifier Found at line no. 4
Identifier Found at line no. 4
Operator Found at line no. 4
Identifier Found at line no. 4
Operator Found at line no. 6
Operator Found at line no. 6
Identifier Found at line no. 6
Identifier Found at line no. 6
Identifier Found at line no. 6
Identifier Found at line no. 6
Identifier Found at line no. 6
Identifier Found at line no. 6
Operator Found at line no. 6
Keyword found at line no. 8
Identifier Found at line no. 8
Seprator found at line no. 8
Seprator found at line no. 8
Seprator found at line no. 9
Operator Found at line no. 10
Operator Found at line no. 10
Identifier Found at line no. 10
Identifier Found at line no. 10
Identifier Found at line no. 11
Seprator found at line no. 11
Identifier Found at line no. 11
Identifier Found at line no. 11
Seprator found at line no. 11
Seprator found at line no. 11
Seprator found at line no. 12
Operator Found at line no. 14
Operator Found at line no. 14
Identifier Found at line no. 14
```

Q11. WAP in lex to remove the comments in a C program.

```
%{
	#include<stdio.h>
%}

%%
\/\/.* {}
\/\*(.*\n)*.*\*\/ {}
%%
int yywrap()
{
	return 1;
}
int main()
{
	yyin=fopen("sample.c","r");
	yyout=fopen("Output.c","w");
	yylex();
	return 0;
}
```

<u>OUTPUT:</u>



imkiller@imkiller: ~

```
imkiller@imkiller:~$ lex 3.c
imkiller@imkiller:~$ gcc lex.yy.c
imkiller@imkiller:~$ ./a.out
imkiller@imkiller:~$ cat Output.c

#include<stdio.h>



void fun()
{

        printf("Hello Fun");
}


int main()
{
        printf("Hello world");
        fun();
        return 0;
}
imkiller@imkiller:~$
```