Program 6

1BY18CS177

Answer :- (6a)

```
%{
#include<studio.h>
int c=0;
%}
%%
"/"[*][^"*/"]*"*/" {c++}
"//".*[^\n] {c++;}
[a-z A-Z 0-9]
%%
int main(int argc, char *argv)
{
FILE *f1, *f2;
if(argc>1)
{
f1 = fopen(argv[1], "r")
if(!f1)
{
printf("file error\n");
exit(1);
}
yyin = f1;
f2 = fopen(argv[2], "w"),
if(!f2)
{printf("error\n");
exit(1);
}
```

```
yyout = f2;
yylex();
printf("number of comment lines :%d\n", c)
}
return 0;
}
```

Output = ③

Program 66.

```
%{
#include <stdio.h>
#include <stdlib.h>
int id=0 dig=0 key=0 op=0
%}
%token DIGIT ID KEY OP
%%
input: DIGIT input {digit}.
| ID input
| KEY input
| OP input
| DIGIT
| ID
| KEY
| OP
%%
extern int yylex();
extern int yyparse();
extern FILE *yyin;
int main() {
```

```
FILE *myfile = fopen("66.c", "r");
if (!myfile)
{
printf("Unable to open file 66.c !\n");
return -1;
}
yyin = myfile;
do
{
yyparse();
} while (!feof(yyin))
printf("Numbers : %d \n keywords = %d \n] identifiers =
%d \n operators = %d \n"
dig, key, id, op);
}
int yyerror()
{
printf("Parsing error: Message : "); exit(0)
}
int yywrap()
{
return 1
}.
```

Output

```
Identifier include.
operator : <
Identifier stdio
Identifier n
operator >
..
```

```
.
.
.
Number : 1
keyword -- 2
operator -1)
Identifier = to
```

```
student@student-virtual-machine:~$ lex 6b.l
student@student-virtual-machine:~$ yacc -d 6b.y
student@student-virtual-machine:~$ cc y.tab.c lex.yy.c -ll
student@student-virtual-machine:~$ ./a.out
Identifier : include
Operator : <
Identifier : stdio
Identifier : h
Operator : >

Operator : /
Operator : *
Identifier : multi
Identifier : line
Identifier : comment
Identifier : can
Identifier : comment

Identifier : multiple
Identifier : lines
Identifier : from
Identifier : the
Identifier : compiler
Operator : *
Operator : /


Keyword: int
Identifier : main


Keyword: int
Identifier : a

Identifier : a
Operator : =
```

```
Digit : 50

Keyword: int
Operator : *
Identifier : ptr

Identifier : ptr
Operator : =
Identifier : a

Identifier : printf
Identifier : n
Identifier : address
Identifier : of
Identifier : a
Operator : =
Identifier : u
Identifier : t
Identifier : value
Identifier : of
Identifier : a
Operator : =
Identifier : d
Identifier : ptr
Operator : *
Identifier : ptr

Identifier : printf
Identifier : n
Identifier : address
Identifier : of
Identifier : ptr
Operator : =
Identifier : u
Identifier : t
```

```
Identifier : value
Identifier : of
Identifier : ptr
Operator : =
Identifier : u
Identifier : n
Identifier : ptr
Identifier : ptr

Operator : /
Operator : /
Identifier : single
Identifier : line
Identifier : comment
Identifier : can
Identifier : comment
Identifier : one
Identifier : lines
Identifier : from
Identifier : the
Identifier : computer

Keyword: return
Digit : 0


Numbers = 2
Keywords = 4
Identifiers = 56
Opeartors = 16
```

```
student@student-virtual-machine:~$ lex 6a.l
student@student-virtual-machine:~$ gcc lex.yy.c -ll
student@student-virtual-machine:~$ ./a.out pointer.c op.txt
The number of commented lines are 2
student@student-virtual-machine:~$ cat op.txt
#include <stdio.h>


int main()
{
        int a;
        a=50;
        int *ptr;
        ptr=&a;
        printf("\n address of a = %u \t value of a = %d", ptr, *ptr);
        printf("\n address of ptr = %u \t value of ptr = %u\n", &ptr, ptr);
                return 0;
}
student@student-virtual-machine:~$
```