**Practical 1:**

%{

# include <stdio.h>

int newLines = 0;

int tabs = 0;

int space = 0;

int restc = 0;

%}

%%

[\n] {newLines++;}

[\t] {tabs++;}

[ ] {space++;}

. {restc++;}

%%

int yywrap(void){}

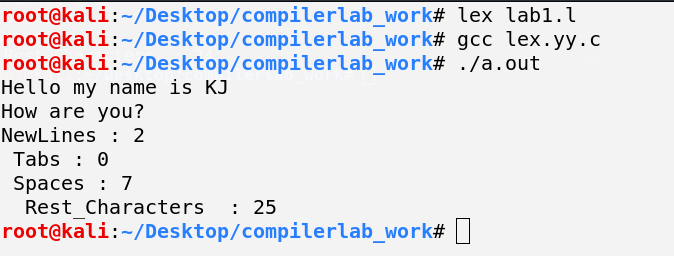
void main(){

yylex();

printf("NewLines : %d\n Tabs : %d\n Spaces : %d\n Rest\_Characters : %d\n",newLines,tabs,space,restc);

}

**OUTPUT**

****

**Practical 2:**

%{

#include <stdio.h>

%}

%%

[a-z|\_|A-Z][a-z|\_|A-Z|0-9]\* {printf("Valid Identifier");}

.\* {printf("Invalid Identifier");}

%%

int yywrap(void){}

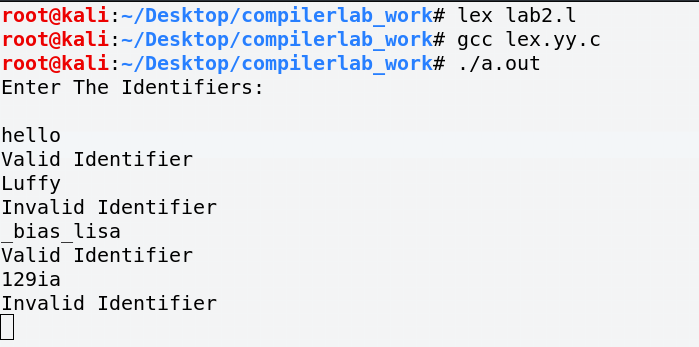
void main(){

printf("Enter The Identifiers: ");

yylex();

}

**OUTPUT**



**Practical 3**

%{

#include <stdio.h>

%}

%%

[0-9]+ {printf("Integer");}

[0-9]+[.][0-9]+ {printf("Float");}

%%

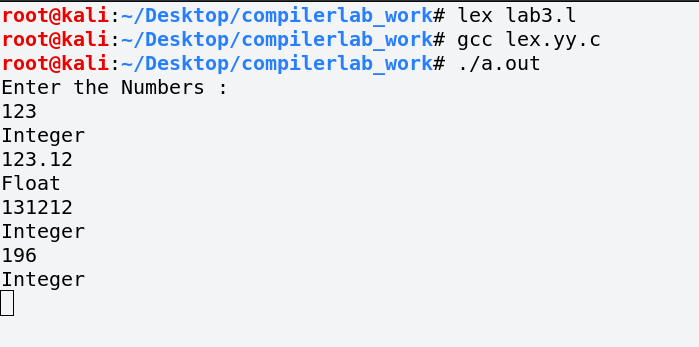
int yywrap(void){}

void main(){

printf("Enter the Numbers : \n");

yylex();}

**OUTPUT**



**Practical 4**

%{

#include <stdio.h>

%}

%%

[auto|double|int|struct|break|else|long|switch|case|enum|register|typedef|char|extern|return|union|continue|for|signed|void|do|if|static|while|default|goto|sizeof|volatile|const|float|short] {printf("\nkeyword");}

[{};,()] {printf("\t Seperator");}

[+-/=\*%] {printf("\t Operator");}

[a-zA-Z][0-9]+|[a-zA-Z]\* {printf("\t Identifier");}

.|\n

%%

int yywrap(void){}

void main(){

printf("Enter the C code : \n");

yylex();}

**OUTPUT**

****

**Practical 5**

%{

#include <stdio.h>

int characters = 0;

int tabs = 0;

int spaces = 0;

int lines = 0;

%}

%%

[ ] {spaces++;}

[\t] {tabs++;}

[\n] {lines++;}

. {characters++;}

%%

int yywrap(void){}

void main(){

extern FILE \*yyin;

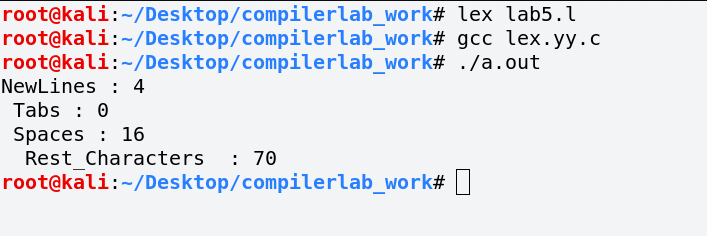
yyin = fopen("input.txt","r");

yylex();

printf("NewLines : %d\n Tabs : %d\n Spaces : %d\n Rest\_Characters : %d\n",lines,tabs,spaces,characters);

}

**OUTPUT**



**Practical 6**

%{

#include <stdio.h>

%}

%%

[\t " "] {fprintf(yyout," ");}

.|\n {fprintf(yyout,"%s",yytext);}

%%

int yywrap(void){}

void main(){

extern FILE \*yyin,\*yyout;

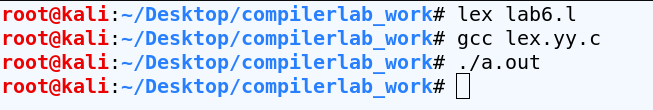
yyin = fopen("input.txt","r");

yyout = fopen("output.txt","w");

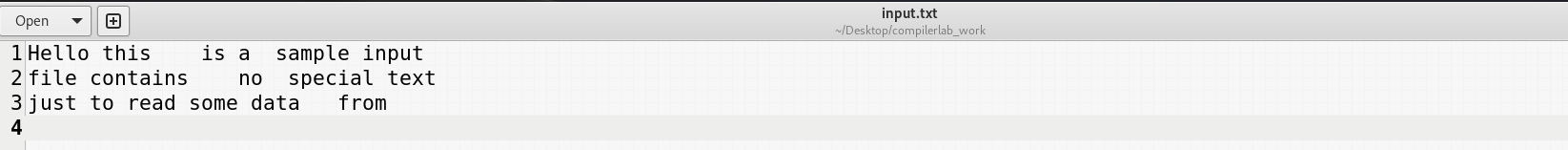
yylex();

}

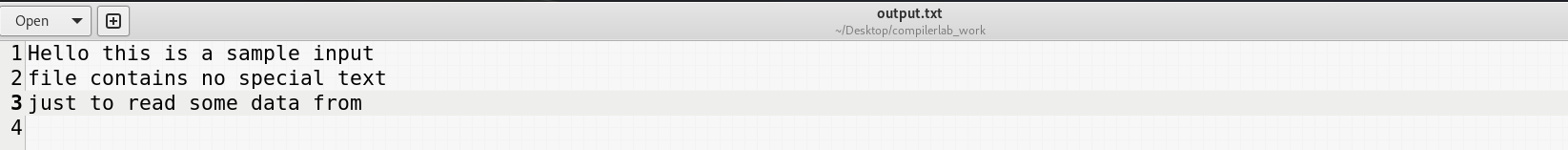
**OUTPUT**

****

**Input**

****

**Output**

****

**Practical 7**

# include <stdio.h>

%}

%%

\/\/(.\*) {};

\/\\*(.\*\n)\*.\*\\*\/ {};

%%

int yywrap(void){}

void main(int k, char \*\*args){

extern FILE \*yyin, \*yyout;

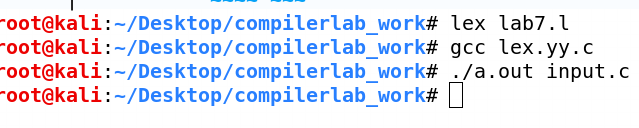
yyin = fopen(args[1],"r");

yyout = fopen("output.c","w");

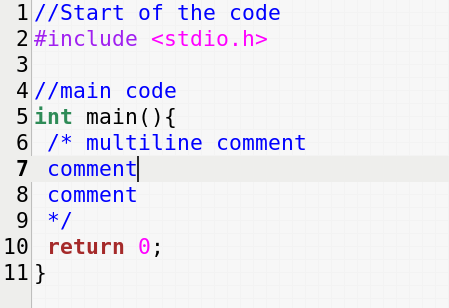
yylex();

}

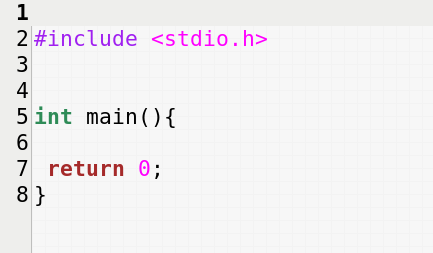
**OUTPUT**

****

**Input.c**

****

**Output.c**

****

**Practical 8**

%{

# include <stdio.h>

%}

%%

"<"[^>]\*> fprintf(yyout,"%s\n",yytext);

.|\n ;

%%

int yywrap(void){}

int main(int k, char \*\*args){

extern FILE \*yyin,\*yyout;

yyin = fopen(args[1],"r");

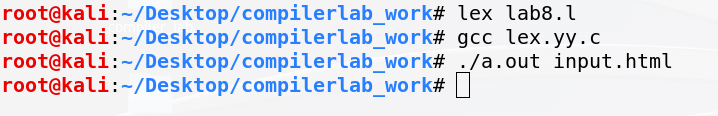
yyout = fopen("outputHTML.txt","w");

yylex();

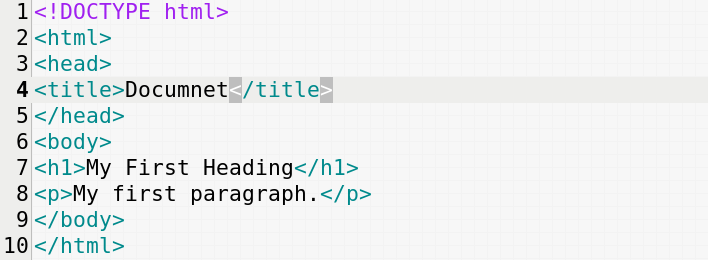
return 0;

}

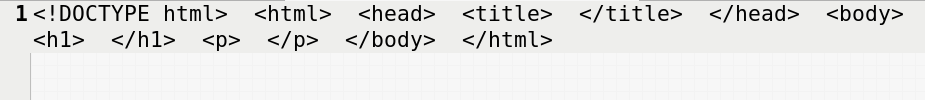
**OUTPUT**

****

**Input.html**

****

**Output.txt**

****

**Even no of a and even no of b:-**

**Practical 9**

%{

#include<stdio.h>

%}

%s A B C DEAD

%%

<INITIAL>b BEGIN A;

<INITIAL>a BEGIN A;

<INITIAL>[^ab\n] BEGIN DEAD;

<INITIAL>\n BEGIN INITIAL; {printf("Accepted\n");}

<A>b BEGIN INITIAL;

<A>a BEGIN C;

<A>[^ab\n] BEGIN DEAD;

<A>\n BEGIN INITIAL; {printf("NOT Accepted\n");}

<B>b BEGIN C;

<B>a BEGIN INITIAL;

<B>[^ab\n] BEGIN DEAD;

<B>\n BEGIN INITIAL; {printf("NOT Accepted\n");}

<C>b BEGIN B;

<C>a BEGIN A;

<C>[^ab\n] BEGIN DEAD;

<C>\n BEGIN INITIAL; {printf("Not Accepted\n");}

<DEAD>[^\n] BEGIN DEAD;

<DEAD>\n BEGIN INITIAL; {printf("Invalid\n");}

%%

int yywrap(){ return -1;}

void main()

{yylex();}

**OUTPUT**



**ThirdLast Element a:-**

**Practical 10:**

%{

#include<stdio.h>

%}

%s A B C D E F G

%%

<INITIAL>b BEGIN A;

<INITIAL>a BEGIN INITIAL;

<A>a BEGIN C;

<A>b BEGIN B;

<B>a BEGIN E;

<B>b BEGIN D;

<C>a BEGIN F;

<C>b BEGIN G;

<D>b BEGIN INITIAL;

<E>a BEGIN C;

<E>b BEGIN B;

<F>a BEGIN F;

<F>b BEGIN G;

<G>a BEGIN E;

<G>b BEGIN D;

<G>\n {printf("Accepted\n");}

<F>\n {printf("Accepted\n");}

<D>\n {printf("Accepted\n");}

<E>\n {printf("Accepted\n");}

<INITIAL>\n {printf("NOT Accepted\n");}

<A>\n {printf("Not Accepted\n");}

<B>\n {printf("Not Accepted\n");}

<C>\n {printf("Not Accepted\n");}

%%

int yywrap()

{

return -1;

}

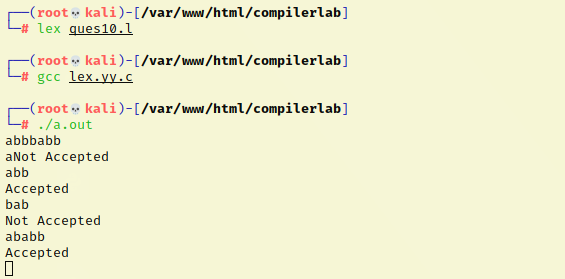
void main()

{

yylex();

}

**OUTPUT:**

****

**Last Element 1:-**

**Practical 11:**

%{

#include<stdio.h>

%}

%s A

%%

<INITIAL>0 BEGIN INITIAL;

<INITIAL>1 BEGIN A;

<A>0 BEGIN INITIAL;

<A>1 BEGIN A;

<INITIAL>\n {printf("NOT Accepted\n");}

<A>\n {printf("Accepted\n");}

%%

int yywrap()

{

return -1;

}

void main()

{

yylex();

}

**OUTPUT:**

