1. Use flex to count number of 'a' and empty spaces in a string.

%{

int num\_of\_a = 0;

int num\_of\_spaces = 0;

%}

%%

[a] ++num\_of\_a;

[ ] ++num\_of\_spaces;

\n return 0;

%%

int yywrap(){}

int main()

{

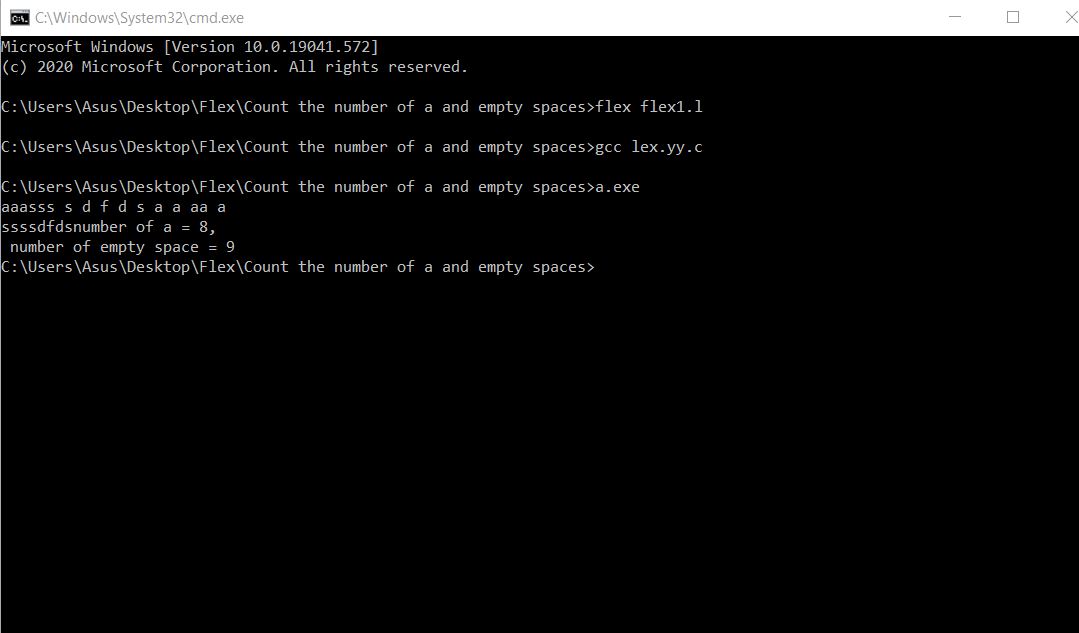
yylex();

printf("number of a = %d,\n number of empty space = %d",

num\_of\_a, num\_of\_spaces);

return 0;

}



2. Use flex to identify negative and positive integers.

%%

[+]?[0-9]+ {printf("positive integer\n");}

[-]?[0-9]+ {printf("negative integer\n");}

.

%%

int yywrap()

{

return 1;

}

int main()

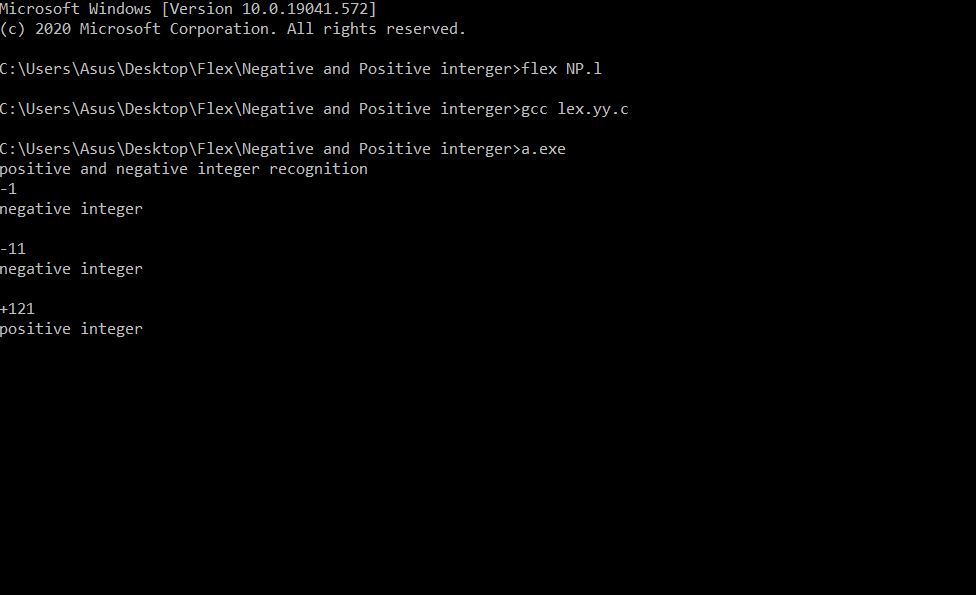
{

printf("positive and negative integer recognition\n");

yylex();

return 0;

}



3. Tokenize the following sentence using flex:

'a = (c+2)/(b-3)\*d

%{

int i = 1;

char v = '(';

char vi = ')';

%}

%%

[a-zA-z]+[0-9]\* {printf("%cid,%d",v,i++);}

[-=+\*/^><] {printf("%c%c%s%c",vi,v,yytext,vi);};

[0-9]\* {printf("%c%s",v,yytext);}

\n return 0;

%%

int yywrap(){}

int main(){

yylex();

printf("%c",vi);

return 0;

}

