Lex code:  
%option noyywrap

%{

#include <stdio.h>

%}

identifier [a-zA-Z][a-zA-Z0-9\_]\*

%%

int|float|double|char printf("'%s' is a dtype.\n", yytext);

while|for|do|if|else|break|continue|return|goto|switch|case printf("'%s' is a keyword.\n", yytext);

"#".\*/\n printf("'%s' is a preprocessing directive\n", yytext);

"//".\*/\n printf("'%s' is a comment.\n", yytext);

"{" printf("'%s' indicates start of block.\n", yytext);

"}" printf("'%s' indicates end of block.\n", yytext);

\".\*\" printf("'%s' is a string.\n", yytext);

[0-9]+ printf("'%s' is a integer.\n", yytext);

"=" printf("'%s' indicates assignment\n", yytext);

"+"|"-"|"\*"|"/"|"&"|"|"|"^"|"%" printf("'%s' is a operator\n", yytext);

{identifier} printf("'%s' is a identifier\n", yytext);

"&"{identifier} printf("'%s' is a pointer.\n", yytext);

";" {}

"\n" {}

. {}

%%

int main() {

yylex();

return 0;

}

Input code:  
#include<stdio.h>

#include<conio.h>

void main()

{

int a,b,c;

a=1;

b=2;

c=a+b;

printf("Sum:%d",c);

}

**lex q1.L(small)**

**gcc lex.yy.c**

**cat input.c**

q2)  
A)

#include <stdio.h>

int main(){

char b[50],r1[50],r2[50],p[50];int i=0,j=0,k=0;

scanf("%s",b);

while(b[i]!='|')r1[j++]=b[i++];r1[j]=0;i++;j=0;

while(b[i])r2[j++]=b[i++];r2[j]=0;

i=j=0;while(r1[i]&&r2[j]&&r1[i]==r2[j])p[k++]=r1[i++],j++;p[k]=0;

puts("---");

if(!r1[i]&&!r2[j])printf("A -> %s\n",p);

else{

printf("A -> %sA'\n",p);

if(!r1[i])printf("A' -> %s\n",r2+j);

else if(!r2[j])printf("A' -> %s\n",r1+i);

else printf("A' -> %s|%s\n",r1+i,r2+j);

}

}

gcc q.c

./a.out

**A->abcd|aBC**

B)  
  
#include <stdio.h>

int main(){

char buf[50],r1[50],r2[50];int i=0,j=0;

scanf("%s",buf);

while(buf[i]!='|')r1[j++]=buf[i++];r1[j]=0;i++;j=0;

while(buf[i])r2[j++]=buf[i++];r2[j]=0;

if(r1[0]!='A'&&r2[0]!='A'){puts("No left recursion");return 0;}

char \*a,\*b;

if(r1[0]=='A'){b=r1+1;a=r2;}else{a=r1+1;b=r2;}

printf("A -> %sA'\nA' -> %sA'|epsilon\n",a,b);

}

gcc q.c

./a.out

A -> Abc|ax

q3)

#include <stdio.h>

int g(char c){return c=='S'?0:c=='A'?1:c=='B'?2:c=='i'?0:c=='+'?1:c=='\*'?2:c=='$'?3:-1;}

char\* l(char t[3][4][5],char A,char a){int i=g(A),j=g(a);return i<0||j<0?0:t[i][j];}

int main(){

char t[3][4][5]={"A","","","", "Bi","","","", "","A+","A\*","e"};

char s[50]="$S",b[50];scanf("%s",b);

int i=1,j=0,k;char\* r;

while(b[j]){

printf("%s\t%s",s,b+j);

if(s[i]>='A'&&s[i]<='Z'){

r=l(t,s[i],b[j]);

if(!r||!\*r){puts("\nFAILED");break;}

if(\*r=='e')s[i--]=0;

else{for(k=0;r[k];k++)s[i++]=r[k];s[i--]=0;}

printf("\t(replace %s)\n",r);

}else{

if(s[i]==b[j]){

if(s[i]=='$'){puts("\nSUCCESS");break;}

s[i--]=0;j++;puts("\t(pop)");

}else{puts("\nFAILED");break;}

}

}

}

gcc q.c

./a.out

Enter: **i+i\*i$**