SET1:	printf("Op: %d\n",d	if(a==0)	yylex();	in_comment = 0;	gets(com);	3.%{	{	temp++;	}	void FIRST(char );	void FIRST(char c)
1. #include <stdio.h></stdio.h>	);		printf("Number of	}	if(com[0]=='/")	int nmacro, nheader;	FILE *f;	printf("t%d = t%d %c	3. %{	int count,n=0;	{
#include <ctype.h></ctype.h>	}	printf("\n It is not a comment"):	Constants : %d\n", cons);	}	{	%}	char file[10];	%c\n", temp, temp - 1, op2, a2):	int yylineno;	char prodn[10][10],	int j;
#include <string.h></string.h>	2. #include <stdio.h></stdio.h>	comment );	fclose(yyin);	else		%%	printf("Enter File Name : "):	42);	%}	first[10];	if(!(isupper(c)))first[n++]=c;
int main(){	int main()	}	}	1	if(com[1]=='/")	^#define { nmacro++; }	scanf("%s",file);	}	9696	int main()	for(j=0;j <count;j++)< td=""></count;j++)<>
int b=0,c=0,d=0,e;	{	Else	4. #include <stdio.h></stdio.h>	if (c == '/' && (c =		^#include { nheader++; }	f = fopen(file,"r");	return 0;	^(.*)\n printf("%4d\t%s".	{	(
	char com[30];	printf("\n It is not a	int main()	getchar()) == '*')	printf("\n It is a comment"):			}	++yylineno, yytext);	int i,choice;	1
char a[100];	int i,a=0;	comment");	-	{		.	yyin = f;	2. #include <stdio.h></stdio.h>	%%	char c,ch;	if(prodn[j][0]==c)
scanf("%s",a);	printf("\n Enter	3	int number1, number2.	in comment = 1:	else if(com[1]=='*')	%%	yylex();	#include <ctype.h></ctype.h>	int yywrap(void) {	printf("How many	{
e=strlen(a);	comment:");	else	sum;	11_comment = 2,	(-2(-)	int yywrap(void) {	printf("\n Number of html tags: %d",tags);	int main()		productions ? :");	if(prodn[j][2]=='\$') first[n++]='\$';
for(int i=0;i <e;i++){< td=""><td>gets(com);</td><td></td><td>printf("Enter two integers:</td><td>,</td><td>for(i=2;i&lt;=30;i++)</td><td>return 1;</td><td></td><td>{</td><td>return 1;</td><td>scanf("%d",&amp;count);</td><td></td></e;i++){<>	gets(com);		printf("Enter two integers:	,	for(i=2;i<=30;i++)	return 1;		{	return 1;	scanf("%d",&count);	
	if(com[0]=='/')	printf("\n It is not a comment");	");	else if (c == ' '     c == '\t')		}	fclose(yyin);	char a[10];	}	printf("Enter %d	else if(islower(prodn[j][2]))first[
if(isspace(a[i]))(		1	scanf("%d %d", &number1,			int main(int argc, char	}	int flag, i=1;	int main(int argc, char *argv[]) {	productions epsilon=\$	n++]=prodn[j][2];
	if(com[1]=='/')	3. digit [0-9]	&number2);	while ((c =	if(com[i]=='*'&&co	*argv[]) {		printf("\n Enter an		:\n\n",count);	else FIRST(prodn[j][2]);
continue;	printf("\n It		sum = number1 +	while ((c = getchar()) == ' '    c == '\t');	m[i+1]=='/')	yyin = fopen(argv[1], "r");	SET3:	printt("\n Enter an identifier:");	yyin = fopen(argv[1], "r");	for(i=0;i <count;i++)< td=""><td>}</td></count;i++)<>	}
}	is a comment");	%{	number2;	putchar(' ');	printf("\n It is a	yylex();	1. #include <stdio.h></stdio.h>	gets(a);	yylex();	scanf("%s%c",prodn[i],&ch);	3
else if(isalpha(a[i])){	else	int cons=0;	printf("%d + %d = %d", number1, number2, sum);	1	comment");	printf("Number of macros	int main() {	if(isalpha(a[0]))	fclose(yyin);	do	1
ii(isaipiia(a[ij))(	if(com[1]=='+')	%}	return 0:	,	a=1;	defined = %d\n", nmacro);	int temp = 1;		}	{	, , , , , , , , , ,
b++;	{	%%	retuin 0,	else	break;	printf("Number of header	char op1, op2, a1, a2,	flag=1;	4. #include <stdio.h></stdio.h>	n=0;	2. #include <stdio.h></stdio.h>
1	for(i=2;i<=30;i++)	{digit}+ { cons++; printf("%s	}	{	}	files included = %d\n", nheader):	res1, res2;	else	int main()	printf("Element :");	#include <ctype.h></ctype.h>
else	1	is a constant\n", yytext); }		putchar(c);	else continue;	fclose(yyin);	printf("Enter an	printf("\n Not a valid identifier");	{	scanf("%c",&c);	#include <string.h></string.h>
if(isdigit(a[i])){		.	SET2:	}	conunue;	iciose(yyiii),	expression: ");	Not a valid identifier ),	int a,b,c;		int n,m=0,p,i=0,j=0;
C++;	if(com[i]=='*'&&co m[i+1]=='/')	%%	1. #include <stdio.h></stdio.h>	}	}	}	scanf(" %c %c %c %c %c",	while(a[i]!='\0')	printf("enter two	FIRST(c);	char a[10][10],f[10];
1	m(i+1)== / )	int yywrap(void) {	int main() {	}	if(a==0) printf("\n It	4. %{	&res1, &op1, &a1, &op2, &a2):	1	numberså€ );	printf("\n FIRST(%c)= { ",c);	void follow(char c);
else	printf("\n It	return 1; }	int c;	return 0;	is not a comment");	int tags;	if (op2 == '*') {	if(!isdigit(a[i])&&!is	scanf("%d	for(i=0;i <n;i++)< td=""><td>void first(char c);</td></n;i++)<>	void first(char c);
if((a[i]=='+')  (a[i]=='-	is a comment");	int main(void)	int in_comment = 0;	}	}	%}		alpha(a[i]))	%d†,&a,&b);	printf("%c ",first[i]);	int main()
')  (a[i]=='*')  (a[i=='/'])){	a=1;	1	while ((c = getchar()) !=	2. #include <stdio.h></stdio.h>	else	%%	printf("t%d = %c %c %c\n", temp, a1, op2, a2);	{	c=a+b;	printf("}\n");	1
	d-1,	FILE *f:	EOF)	int main()	printf("\n It	"<"[^>]*> { tags++;	temp++:	flag=0;	return 0;	printf("press 1 to continue:	int i,z;
d++;	break;	char file[10]:	{	, ,	is not a comment");	printf("%s \n", yytext); }	printf("t%d = %c %c	break;	}	");	
}	1		if (in_comment)	1	}	.[\n {}	t%d\n", temp, res1, op1,	} i++;		scanf("%d%c",&choice,&ch)	char c,ch;
}	else	printf("Enter File Name : ");		char com[30];	else	%%	temp-1);	}	SET4:	;	printf("Enter the no.of productions:");
printf("Id: %d\n",b);	continue;	scanf("%s",file);	if (c == '*' && (c =	int i,a=0;	printf("\n It	int yywrap(void) {	} else {	if(flag==1)	1. #include <stdio.h></stdio.h>	}	scanf("%d",&n);
printf("Char:	conunue;	f = fopen(file,"r");	getchar()) == '/')	printf("\n Enter	is not a comment");	return 1; }	printf("t%d = %c %c	printf("\n		while(choice==1);	Jeanny 1964 , carly,
%d\n",c);	}	yyin = f;	{	comment:");	1	int main(void)	%c\n", temp, res1, op1, a1);	Valid identifier");	#include <ctype.h></ctype.h>	}	
					•						

printf("Enter the		-3	SET5:	printf("Grammar without	for(i=0;i<=len;i++)	%%	SET6:	printf("\n");	gets(input);	{	(include void main printf i
productions(epsilon=\$):\n") :	if(a[i][j+1]!='\0')first(a[i][j+1 ]):	%%	1. #include <stdio.h></stdio.h>	left recursion:\n");	{	[a-z] {printf("%c",yytext[0]-	1. #include <stdio.h></stdio.h>	for (i = 0; i < len; i++)	E();	i++;	nt) { printf("%s is a keyword\n",yytext);
for(i=0;i <n;i++)< td=""><td></td><td>int main()</td><td>#include<string.h></string.h></td><td>printf("%c- &gt;%c%c\",non_terminal,bet</td><td>printf("\n");</td><td>32);}</td><td>#include<string.h></string.h></td><td>{</td><td></td><td>F();</td><td>count_key++; }</td></n;i++)<>		int main()	#include <string.h></string.h>	printf("%c- >%c%c\",non_terminal,bet	printf("\n");	32);}	#include <string.h></string.h>	{		F();	count_key++; }
scanf("%s%c",a[i],&ch);	if(a[i][j+1]=='\0'&&c!=a[i][0]	{	#define SIZE 10	a,non_terminal);	for (i = 0; i < len; i++)	· {} %%	#include <ctype.h></ctype.h>	for (j = 0; j < 5; j++)	if(strlen(input)==i&&error= =0)	Tprime();	{ letter}({letter} {digit})* {   printf("%s is a identifier\n",
do	follow(a[i][0]);	printf("\n Enter the input	int main () {	printf("\n%c\'- >%c%c\' E\n",non_terminal	{		#include <stdlib.h></stdlib.h>	{		}	yytext); count_id++; }
{	lollow(a[i][o]),	string:"); yylex();	char non_terminal;	,alpha,non_terminal);	if ((expr[i] >= 'a' && expr[i] <= 'z')    (expr[i] >=	int yywrap(void){} int main()	int main()	if (expr[i] == operators[j])	printf("\nAccepted\n");	}	{digit}+ { printf("%s is a
m=0;	)	yyieA(),	char beta,alpha;	} else	'A' && expr[i] <= 'Z'))	I I I I I I I I I I I I I I I I I I I	{	{	else printf("\nRejected\n");	void F()	number\n", yytext); }
printf("Enter the element	}	int yywrap()	int num;	printf(" can't be reduced\n"):	{	printf("\nenter the string :	char expr[100];	<pre>printf("%c\t %p\t\t operator ", expr[i],&amp;expr[i]);</pre>	}	{	\"(\\. [^"\\])*\" { printf("%s is a string literal\n", yytext);
whose FOLLOW is to be found:");	}	{	char production[10][SIZE];		printf("%c\t %p\t\t	");	char operators[5] = {'+', '- ', '*', '/','='};	printf("\n"):	void E()	if(isalnum(input[i]))i++;	}
scanf("%c",&c);	void first(char c)	return 1;	int index=3; /* starting	} else printf(" is not left	identifier", expr[i], &expr[i]);	yylex();	int len, i, j;	}		else if(input(i]=='(')	.
follow(c);	{	}	of the string following "->"	recursive.\n");	printf("\n");	}	printf("Enter an	)	т();	{	%%
printf("FOLLOW(%c) = {	int k;	4. %[	printf("Enter Number of	index=3;	}	4. %{	expression: ");	}	Eprime();	i++;	int yywrap(void) {
",c);	if(!(isupper(c)))f[m++]=c;	#include <stdio.h></stdio.h>	Production : ");	}	}	#include <stdio.h></stdio.h>	scanf("%s", expr);	printf("\n");	}	E();	return 1;
for(i=0;i <m;i++)< td=""><td>for(k=0;k<n;k++)< td=""><td>%}</td><td>scanf("%d",#);</td><td>}</td><td>printf("\n");</td><td>%}</td><td>len = strlen(expr);</td><td>}</td><td>void Eprime()</td><td>if(input[i]==')')</td><td>}  </td></n;k++)<></td></m;i++)<>	for(k=0;k <n;k++)< td=""><td>%}</td><td>scanf("%d",#);</td><td>}</td><td>printf("\n");</td><td>%}</td><td>len = strlen(expr);</td><td>}</td><td>void Eprime()</td><td>if(input[i]==')')</td><td>}  </td></n;k++)<>	%}	scanf("%d",#);	}	printf("\n");	%}	len = strlen(expr);	}	void Eprime()	if(input[i]==')')	} 
printf("%c ",f[i]);	{	%%	printf("Enter the	2. #include <stdio.h></stdio.h>	for (i = 0; i < len; i++)		printf("Symbol\t Address\t\t Type");	}	{	i++;	int main(int argc, char *argv[]) {
printf(" }\n"); printf("Do you want to	if(a[k][0]==c)	[a-zA-Z0-9_]+(@[a-z]+)(.[a-	grammar as E->E-A :\n"); for(int i=0:i <num:i++){< td=""><td>#include<string.h></string.h></td><td>{</td><td>%%</td><td>printf("\n");</td><td>2. #include<stdio.h></stdio.h></td><td>if(input[i]=='+')</td><td></td><td>yyin = fopen(argv[1], "r");</td></num:i++){<>	#include <string.h></string.h>	{	%%	printf("\n");	2. #include <stdio.h></stdio.h>	if(input[i]=='+')		yyin = fopen(argv[1], "r");
continue(0/1)?");	{	z]+) {printf("%s is a valid email", yytext);}	scanf("%s",production[i]);	#include <ctype.h></ctype.h>	for (j = 0; j < 5; j++)	[6-9][0-9]{9} {printf("%s, is a mobile number", yytext);}	for(i=0;i<=len;i++)	#include <string.h></string.h>	{	else error=1;	yylex();
scanf("%d%c",&z,&ch);	if(a[k][2]=='\$') follow(a[i][0]);	.* {printf("It is not a valid	} for(int i=0;i <num;i++){< td=""><td>#include<stdlib.h></stdlib.h></td><td>{</td><td>.+ {printf("The Mobile</td><td>{</td><td>#include<ctype.h></ctype.h></td><td>i++;</td><td>}</td><td>printf("number of</td></num;i++){<>	#include <stdlib.h></stdlib.h>	{	.+ {printf("The Mobile	{	#include <ctype.h></ctype.h>	i++;	}	printf("number of
}	else	email");}	printf("\nGRAMMAR	int main()	if (expr[i] == operators[j])	Number is Invalid");}	printf("\n");	char input[10];	T();		identifiers = %d\n", count id);
while(z==1);	if(islower(a[k][2]))f[m++]=a[	%%	:::%s",production[i]); non_terminal=production[i]	{	{	%%	for (i = 0; i < len; i++)	int i,error;	Eprime();	else error=1;	printf("number of keywords
}	k][2];	int main()	[0]; if(non terminal==productio	char expr[100];	printf("%c\t %p\t\t operator ", expr[i],&expr[i]);	int main()	{	void E();	}	}	= %d\n", count_key);
void follow(char c)	else first(a[k][2]);	{ printf("\n Enter the	n[i][index]) {	char operators[5] = {'+', '- ', '*', '/','='};	printf("\n");	{ printf("\n Enter the mobile	if ((expr[i] >= 'a' &&	void T();	} void T()	3. digit [0-9]	fclose(yyin);
{	}	printr("\n Enter the email:");	alpha=production[i][index+ 1]:	int len, i, j;	)	printt("\n Enter the mobile number:");	expr[i] <= 'z')    (expr[i] >= 'A' && expr[i] <= 'Z'))	void Eprime();	void I()	letter [A-Za-z]	}
if(a[0][0]==c)f[m++]='\$';	1	yylex();	printf(" is left	printf("Enter an	}	yylex();	{	void Tprime(); void F():	f();	%{	SAMPLE INPUT:
for(i=0;i <n;i++)< td=""><td>3. %{</td><td>}</td><td>recursive.\n"); while(production[i][index]!=</td><td>expression: ");</td><td>}</td><td>}</td><td>printf("%c\t %p\t\t</td><td></td><td>Tprime();</td><td>int count_id,count_key; %}</td><td>#Include<stdio.h></stdio.h></td></n;i++)<>	3. %{	}	recursive.\n"); while(production[i][index]!=	expression: ");	}	}	printf("%c\t %p\t\t		Tprime();	int count_id,count_key; %}	#Include <stdio.h></stdio.h>
{	#include <stdio.h></stdio.h>	int yywrap()	0 &&	scanf("%s", expr);	printf("\n");	int yywrap()	identifier", expr[i], &expr[i]);	main()	ipililety,	76) 96%	void main()
for(j=2;j <strlen(a[i]);j++)< td=""><td>%}</td><td>€</td><td>production[i][index]!=' ')</td><td>len = strlen(expr);</td><td>}</td><td>{</td><td>printf("\n");</td><td>i=0:</td><td>void Tprime()</td><td>(stdio.h conio.h) {</td><td>(</td></strlen(a[i]);j++)<>	%}	€	production[i][index]!=' ')	len = strlen(expr);	}	{	printf("\n");	i=0:	void Tprime()	(stdio.h conio.h) {	(
{	%%	return 1;	index++; if(production[i][index]!=0) {	printf("Symbol\t Address\t\t Type");	}	return 1;	}	error=0:	{	printf("%s is a standard	int a,b,c = 30; printf("hello");
if(a[i][j]==c)	[A-Z] {printf("%s is a capital	}	beta=production[i][index+1]	printf("\n");	3. %{	}	}	printf("Enter an	if(input[i]=='+')	library\n",yytext); }	printit tieno ),
1	letter", yytext);}		;		%}			arithmetic expression : ");			,

4. %{ int vow count=0;	gets(string); if(string[0]=='a') {	}	printf("\n\$%s\t%s\$\t%sid",s	if(stk[z]=='E' && stk[z+1]=='+' &&	printf("\n\$%s\t%s\$\t%s",stk	[0-9]+ {printf("%s is a digit", yytext);}	/*stack + - * / ^ i ( ) \$ */	switch(c)	<pre>if(stack[top- t]!=handles[i](t])</pre>	for(j=i;j <l;j++) printf("%c",*(input+j));</l;j++) 	dispstack();
= '		,	tk,a,act);	stk[z+2]=='E')	,a,ac);	[a-z A-Z]+ {printf("%s is an			{	printi( /ec , (input+j//,	
int const_count =0;	flag=0;	}	check();	{	i=i-2;	alphabet", yytext);}	/* +*/ '>'.	case '+':return 0;	found=0:	}	printf("\t");
%}	for (count=1;string[count-	}	}	stk[z]='E';	}	-7	5,,4,,4,,4,,4,,5,,5,	case '-':return 1;	break;	void main()	
%%	1]!='\0';count++) {	2. #include <stdio.h></stdio.h>	else	stk[z+1]="\0";	}	%%		case '*':return 2;	break,	{	dispinput();
[aeiouAEIOU]		#include <string.h></string.h>	{	stk[z+2]="\0";	3. %{	int main()	/* -*/ '>',	case '/':return 3;	3	int j;	
{vow_count++;}	if(string[count]=='b'	int k=0,z=0,i=0,j=0,c=0;	stk[i]=a[i];		int vow count=0;	1	'>','<','<','<','<','>','>','>',	case '^':return 4;	}	input=(char*)malloc(50*siz	printf("\tReduced: E->%s".lasthandle):
[a-zA-Z] {const_count++;}	){	char	stk[i+1]='\0';	printf("\n\$%s\t%s\$\t%s",stk	int const count =0;	printf("\n Enter the input		case 'l':return 5;	if(found==1)	eof(char));	1
%%	fi	a[16],ac[20],stk[15],act[10];		,a,ac);	- '	string:");	/* **/ '>',	case '(':return 6;	{	printf("\nEnter the	,
int yywrap(){}	flag=1;	void check();	a[j]=' ';	i=i-2;	%}	vvlex();	'>','>','>','<','<','>','>',	case ')':return 7;	stack[top-t+1]='E';	string\n");	,
int main()	continue:	int main()	printf("\n\$%s\t%s\$\t%ssym	}	%%	1		case 'S':return 8:	top=top-t+1;	scanf("%s",input);	}
{	1	{	bols",stk,a,act);	for(z=0; z <c; td="" z++)<=""><td>[aeiouAEIOU]</td><td>, lat</td><td>/* /*/ '&gt;',</td><td>case 3 Jetum 6,</td><td></td><td>input=strcat(input,"\$");</td><td>if(strcmp(stack,"\$E\$")==0)</td></c;>	[aeiouAEIOU]	, lat	/* /*/ '>',	case 3 Jetum 6,		input=strcat(input,"\$");	if(strcmp(stack,"\$E\$")==0)
printfi"Enter the string of	else ?	puts("GRAMMAR is E-	check();	if(stk[z]=='E' &&	{vow_count++;}	int yywrap()	'>','>','>','<','<','<','>',	}	strcpy(lasthandle,handles[i]	l=strlen(input);	printf("\nAccepted;");
vowels and consonants:");	if((flag==1)&&(string[count]	>E+E \n E->E*E \n E->(E) \n	3	stk[z+1]=='+' &&	[a-zA-Z] {const_count++;}	{		}	);	strcpy(stack,"\$");	else
yylex();	=='a')) {	E->id");	,	stk[z+2]=='E')	%%	return 1;	/* ^ */ '>',	int shift()	stack[top+1]="\0';	printf("\nSTACK\tINPUT\tA	printf("\nNot
printf("Number of vowels	printf("The string	puts("enter input string	,	{	int yywrap(){}	}	'>','>','>','<','<','>','>',	{	return 1;//successful	CTION");	Accepted;");
are: %d\n", vow_count);	does not belong to the	1,	3	stk[z]='E';	int main()			stack[++top]=*(input+i++);	reduction	while(i<=I)	}
printf("Number of	specified grammar");	gets(a);	void check()	stk[z+1]="\0";	{	SET 8:	/* i */ '>',	stack[top+1]="\0";	}	{	2#include <stdio.h></stdio.h>
consonants are: %d\n",		c=strlen(a);	{	stk[z+1]="\0";	printf("Enter the string of	1. #include <stdio.h></stdio.h>	'>','>','>','>','e','e','e','>','>',	}	}	shift();	int main() {
const_count);	break;	strcpy(act,"SHIFT->");	strcpy(ac,"REDUCE TO		vowels and consonants:");	#include <string.h></string.h>		int reduce()	}	printf("\n"):	int temp = 1;
return 0;	} else if(string[count]=='a')	puts("stack \t input \t	E");	printf("\n\$%s\t%s\$\t%s",stk	yylex();	char *input;	/* ( */ '<',		return 0;	dispstack():	char op1, op2, a1, a2,
}	eise ii(striiig(count) a )	action");	for(z=0; z <c; td="" z++)<=""><td>,a,ac);</td><td>printf("Number of vowels</td><td>int i=0:</td><td>'&lt;','&lt;','&lt;','&lt;','&lt;','&gt;','e',</td><td>1</td><td>}</td><td>printf("\t"):</td><td>res1, res2;</td></c;>	,a,ac);	printf("Number of vowels	int i=0:	'<','<','<','<','<','>','e',	1	}	printf("\t"):	res1, res2;
	continue; else	for(k=0,i=0; j <c;< td=""><td>if(stk[z]=='i' &amp;&amp;</td><td>i=i-2;</td><td>are: %d\n", vow_count);</td><td>,</td><td></td><td>int i,len,found,t;</td><td>void dispstack()</td><td></td><td>printf("Enter an</td></c;<>	if(stk[z]=='i' &&	i=i-2;	are: %d\n", vow_count);	,		int i,len,found,t;	void dispstack()		printf("Enter an
SET7:	if((flag==1)&&(string[count]	k++,i++,j++)	stk[z+1]=='d')	}	printf("Number of	char lasthandle[6],stack[50],han	/* ) */ '>',	for(i=0;i<5;i++)//selecting handles		dispinput();	expression: ");
1. #include <stdio.h></stdio.h>	='\0')) {	{	{	for(z=0; z <c; td="" z++)<=""><td>consonants are: %d\n", const count);</td><td>dles[][5]={")E(","E*E","E+E",</td><td>'&gt;','&gt;','&gt;','e','e','e','&gt;','&gt;',</td><td>f</td><td>int j;</td><td>printf("\tShift");</td><td>scanf(" %c %c %c %c %c",</td></c;>	consonants are: %d\n", const count);	dles[][5]={")E(","E*E","E+E",	'>','>','>','e','e','e','>','>',	f	int j;	printf("\tShift");	scanf(" %c %c %c %c %c",
#include <string.h></string.h>		if(a[j]=='i' && a[j+1]=='d')	stk[z]='E';	if(stk[z]=='(' &&	return 0;	"I","E^E"};		1		if(prec[getindex(sta	&res1, &op1, &a1, &op2, &a2);
int main() {	printf("String not accepted!!!!"):	{	stk[z+1]='\0';	stk[z+1]=='E' && stk[z+2]==')')	retuiro,	//(E) becomes )E( when	/* \$ */ '<', '<','<','<','<','<','>','<','>',	len=strlen(handles[i]);	for(j=0;j<=top;j++)	ck[top])][getindex(input[i])] =='>')	if (op2 == '*') {
char string[50];		stk[i]=a[j];		Stk[2+2]== ) )	1	pushed to stack	<,<,<,<,<,<,<,>,	if(stack[top]==handles[i][0]	printf("%c",stack[j]);		
int flag,count=0;	break;	stk[i+1]=a[j+1];	printf("\n\$%s\t%s\$\t%s",stk ,a,ac);	{	4.%{	int top=0,I;		&⊤+1>=len)	}	,	printf("t%d = %c %c %c\n", temp, a1, op2, a2);
printf("The	}	stk[i+2]='\0';	j++;	stk[z]='E';	#include <stdio.h></stdio.h>	char prec[9][9]={	};	{		while(reduce())	temp++:
grammar is: S->aS, S->Sb, S-	else {	a[i]=' ';	,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,	stk[z+1]="\0";	%}			found=1:	void dispinput()	{	
>ab\n");		a[j+1]=' ';	}	stk[z+1]="\0";	%%	/*input*/	int getindex(char c)	for(t=0;t <len;t++)< td=""><td>{</td><td></td><td>printf("t%d = %c %c t%d\n", temp, res1, op1,</td></len;t++)<>	{		printf("t%d = %c %c t%d\n", temp, res1, op1,
printf("Enter the	printf("String accepted"):	,	for(z=0; z <c; td="" z++)<=""><td></td><td>if else {printf("%s is a</td><td></td><td>{</td><td>i i (i-o,icien,irr)</td><td>int j;</td><td>printf("\n");</td><td>temp-1);</td></c;>		if else {printf("%s is a		{	i i (i-o,icien,irr)	int j;	printf("\n");	temp-1);
string to be checked:\n");					keyword", yytext);}			1			} else {

printf("t%d = %c %c	.+ {printf("\n%s is not		int main()		printf("ADD\n");	3. %%	{	}	printf("%c\t",	char stack [5][2];	install(prod[i],
%c\n", temp, res1, op1, a1);	mobile number",yytext);}	else if(f2==-1)	{	ans=op1/op2;	break;	((http) (ftp))s?:\/\/[a-zA-Z0-	return 1;	int main() {	arr[i][j]);	int top = -1;	res[i][j]);
temp++;	%%	{	yylex();	}	case MULTIPLY:	9](.[a-z])+(.[a-zA-Z0- 9+=?]*)* {printf("\nURL	}	int i = 0, j;	}		break;
printf("t%d = t%d %c	int yywrap(){}		}			94=1) (printit (nort	SET 10:	char pro, re, pri = ' ';	getch();	void install(char pro. char	} else if (res[i][j] == 'E'
%c\n", temp, temp - 1, op2, a2):	int main()	op2=atof(yytext);	void eval()	break;	printf("MULTIPLY\n");	.+ {printf("\nURL	1. #include <conio.h></conio.h>		printf("\n\n");	re) {	res[i][j] == 'F'    res[i][j] == 'T') {
1	{	f2=1;	f f	default:	break;	Invalid\n");}	#include <stdio.h></stdio.h>	for (i = 0; i < 6; ++i) {	for (i = 0; i < 18; ++i) {	int i;	if (res[i][j - 1] == '+'
,	while(yylex());	}	1		default:	%%			if (pri != arr[i][0]) {	for (i = 0; i < 18; ++i) {	res[i][j - 1] == '*'
return 0;	3	if((f1==1) &&	switch(oper)	printf("operation not available"):	fprintf(stderr, "Error:	void main()	char arr[18][3] ={{'E', '+', 'F'},{'E', '*', 'F'},{'E', '(', 'F'},	for (j = 0; j < 3 && res[i][j] != '\0'; ++i) {	pri = arr[i][0];	if (arr[i][0] == pro &&	res[i][j - 1] == '('     res[i][j -
}	,	(f2==1))	{		Unknown token type\n");	I	{'E', ')', 'F'},{'E', 'I', 'F'},{'E',	if (res[i][i] == '+'	printf("\n\t36c -> ".	arr[i][1] == re) {	1] == ')'
3. %%		{	case '+':	break;	break;	nrintf("\nEnter URL	'\$', 'F'},	res[i][] == '*'    res[i][] ==	printi( \n\tau \sec -> , pri);	}	res[i][j - 1] == 'i'    res[i][j - 1] == '\$') {
[\t]+;	SET 9:	eval();		}	3	printt("\nEnter UKL : ");	{'F', '+', 'F'},{'F', '*', 'F'},{'F',	'('    res[i][j] == ')'	3	1	
[0-9]+[0-9]*\.[0-9]+{	1.%{	f1=0;	ans=op1+op2;	printf("The answer	,	yylex();	'(', 'F'},('F', ')', 'F'},{'F', 'I', 'F'},{'F', 'S', 'F'}, {'T', '+', 'F'},	res[i][j] == 'i'    res[i][j] == '\$') {	if (arr[i][2] == 'T')	, ++top:	install(prod[i], res[i][i-1]);
printf("\n%s is NUMBER", vytext);}	#undef yywrap	f2=0:		is = %If",ans);	1	printf("\n"):	{'T', '*', 'F'}, {'T', '(', 'F'), {'T',	install(prod[i].			break;
	#define yywrap() 1	12=0;	break;	}	int main() {	printi( (ri );	')', 'F'},{'T', 'i', 'F'},{'T', '\$',	res[i][j]);	printf("%c ", arr[i][1]);	arr[i][2] = 'T';	1
#.* { printf("\n%s is COMMENT", yytext);}	int f1=0,f2=0;	}	case '-':	2. #include <stdio.h></stdio.h>	// Example expression: 2 * (3 + 4)	}	'F'}};	break:	1		1
[a-zA-Z]+ { printf("\n%s is	char oper;	}		typedef enum {	Token tokens[] = {	int yywrap()	char prod[] = "EETTFF";	l l	getch();		}
WORD", vytext);}	float op1=0,op2=0,ans=0;	{OP}{	ans=op1-op2;			{	char res[6][3] ={ {'E', '+', 'T'},	,	getch();	stack[top][0] = pro;	}
\n { ECHO;}	void eval();	oper=(char)	break;	NUMBER,	{NUMBER, 2},	}	{'T', '\0'}, {'T', '*', 'F'}, {'F',	}	}	stack[top][1] = re;	}
%%	%) %)	*yytext;	case '*':	ADD,	{NUMBER, 3},	4. %{	'\0'}, {'(', 'E', ')'}, {'i', '\0'});	}	2. #include <conio.h></conio.h>	}	
int main()	%}	f2=-1;	case ·:	MULTIPLY,	{NUMBER, 4},	#include <stdio.h></stdio.h>	char stack [5][2];	while (top >= 0) {	#include <stdio.h></stdio.h>		while (top >= 0) {
int main()		}	ans=op1*op2:	} TokenType;	{ADD, 0},	%}	int top = -1;	pro = stack[top][0];	char arr[18][3] ={{'E', '+',	int main() {	pro = stack[top][0];
{	DIGIT [0-9]			typedef struct {	{MULTIPLY, 0},	%%	void install(char pro, char	re = stack[top][1];	'F'}, {'E', '*', 'F'}, {'E', '(', 'F'),		re = stack[top][1];
while( yylex());	NUM {DIGIT}+(\.{DIGIT}+)?	[\n] {	break;	TokenType type;	}:	[0-9][0-9]\/[0-1][0-9]\/[1-	re) {	top;	('E', ')', 'F'), ('E', 'I', 'F'), ('E', '\$', 'F'), ('F', '+', 'F'),	int i = 0, j;	-top;
}	OP [*/+-]	if(f1==1 && f2==1)	case '/':	int value;	for (int i = 0: i <	[0-9][0-9]\/[0-1][0-9]\/[1- 2][0-9]{3} {printf("%s, is a	int i;	for (i = 0; i < 6; ++i) {	{E', 'S', 'F'}, {F', 'F', 'F'}, {'F', 'F'},	char pro, re, pri = ' ';	
int yywrap( )		(12-13412-1)	,.	} Token:	sizeof(tokens) /	Valid DOB", yytext);}	for (i = 0; i < 18; ++i) {	if (res[i][0] == pro &&	{'F', ')', 'F'}, {'F', 'I', 'F'},		for (i = 0; i < 6; ++i) {
{	%%		if(op2==0)	, ,	sizeof(tokens[0]); i++) {	.+ {printf("DOB is Invalid");}	if (arr[i][0] == pro &&	res[i][0] != prod[i]) {	{'F', '\$', 'F'}, {'T', '+', 'F'},		for (j = 2; j >= 0;j) {
return 1;		eval;	{	void generateOptimizedCode(To		%%	arr[i][1] == re) {	install(prod[i], re);	{'T', '*', 'F'}, {'T', '(', 'F'),	for (i = 0; i < 6; ++i) {	if (res[i][0] == pro && res[i][0] != prod[i]) {
1	{NUM} {	f1=0;		ken token) {	generateOptimizedCode(to kens[i]);	int main()	arr[i][2] = 'T';	1	{'T', ')', 'F'}, {'T', 'I', 'F'},	for (j = 2; j >= 0;j) {	
4.%{		f2=0;	printf("ERROR");	switch (token.type) {	1		break;	,	{'T', '\$', 'F'},		install(prod[i], re);
	if(f1==0)	}		case NUMBER:	,	1	}	,	};	if (res[i][j] == '+'	break:
#include <stdio.h></stdio.h>	{	}	return;	printf("PUSH %d\n".	return 0;	printf("\n Enter the DOB:");	}	,	char prod[6] = "ETF";	res[i][j] == '+'    res[i][j] ==	
%}		%%	}	token.value);	}	yylex();	++top;	for (i = 0; i < 18; ++i) {	char res[6][3] ={ {'E', '+',	'('    res[i][j] == ')'    res[i][j] == 'l'    res[i][j] ==	} else if (res[i][0] != "\0") break:
%%	op1=atof(yytext);		else	break;		}	stack[top][0] = pro;	printf("\n\t");	'T'}, {'T', '\0', '\0'}, {'T', '*', 'F'}, {'F', '\0', '\0'}, {'(',	'\$') {	1-,
[6-9][0-9]{9} {printf("\n%s is	f1=1;		{	case ADD:		int yywrap()		for (j = 0; j < 3; ++j)	'E', 'J'}, {'i', \0', \0'},};		,
mobile number",yytext);}	}						stack[top][1] = re;				}

```
count and ignore comment lines
                                                                                                                               break;
                                                                               eval();
 for (i = 0; i < 18; ++i) {
                                return 1;
                                                                                                                                     case '-':
                                                                                                                                                         #include <stdio.h>
   printf("\n\t");
                                                                               f1=0;
                                                                                                                                                                                          int main()
                                                                                                                                                         // This is a single-line
                                4. %{
                                                                               f2=0;
   for (j = 0; j < 3; ++j)
                                                                                                                               ans=op1-op2;
                                                                                                                                                                                          yyin = fopen("input.c",
"r");
     printf("%c\t",
                                #undef yywrap
                                                                                                                                                         int main() {
                                #define yywrap() 1
                                                                                                                                                          /* This is
}
                                                                                                                                                                                          yyout = fopen("outp.c",
"w");
                                                                 {OP} {
                                                                                                                                     case '*':
                                int f1=0,f2=0;
                                                                                                                                                           a multi-line
 printf("\n\n");
                                char oper;
                                                                        oper=(char)
                                                                                                                                                           comment */
                                                                                                                                                                                            yylex();
                                                                                                                               ans=op1*op2;
 for (i = 0; i < 18; ++i) {
                                float op1=0,op2=0,ans=0;
                                                                                                                                                           printf("Hello, World!\n");
                                                                                                                                                                                            printf("%d\n",c);
   if (pri != arr[i][0]) {
                                                                        f2=-1:
                                                                                                                                                         // Another single-line comment
                                void eval();
                                                                                                                               break;
                                                                                                                                                                                            fclose(yyin);
    pri = arr[i][0];
                                                                                                                                     case '/':
                                                                                                                                                                                            fclose(yyout);
                                                                 [\n] {
     printf("\n\t%c -> ",
                                DIGIT [0-9]
                                                                                                                                                                                            return 0;
                                                                        if(f1==1 && f2==1)
                                                                                                                                                           return 0;
                                                                                                                               if(op2==0)
                                NUM {DIGIT}+(\.{DIGIT}+)?
 }
                                OP [*/+-]
   if (arr[i][2] == 'T')
                                                                                                                                                        [10:19 AM, 11/16/2023]
Harish Padipps: input file
                                %%
     printf("%c ",
                                                                                                                               printf("ERROR");
                                                                               f1=0;
arr[i][1]);}
                                {NUM} {
                                                                                                                                                        [10:36 AM, 11/16/2023]
Harish Padipps: %{
                                                                               f2=0;
}
                                       if(f1==0)
                                                                                                                               return;
3. %{
                                       {
                                                                                                                                                         int c = 0;
#include<stdio.h>
                                                                                                                                              else
                                                                  %%
                                        op1=atof(yytext);
%}
                                                                                                                                                         %}
                                                                                                                               ans=op1/op2;
                                                                {
[0-9]+ {printf("%s, is a digit",
                                                                                                                                                         %%
yytext);}
                                                                        yylex();
                                                                                                                                                         \\\[^\n]* { c++; };
                                                                                                                                                         // Match and ignore single-
line comments
.+ {printf("it is not a digit");}
                                        else if(f2==-1)
                                                                                                                               break;
%%
                                       {
                                                                 void eval()
                                                                                                                                                        V\([^]|(\+[^V]))\+V { c++;
};// Match and ignore
multi-line comments
                                        op2=atof(yytext);
                                                                                                                               printf("operation
                                                                         switch(oper)
                                                                                                                        not available");
break;
                                              f2=1;
printf("\n Enter the digit
                                                                                                                                                         %%
                                                                                                                               }
                                                                               case '+':
yylex();
                                                                                                                                                         int yywrap()
                                                                                                                               printf("The answer
                                       if((f1==1) &&
                                                                        ans=op1+op2;
                                                                                                                        is = %If",ans);
                                (f2==1))
```