1.Write a program for Find First of the given grammar?

#include<stdio.h>

#include<ctype.h>

int main()

{

int i,n,j,k;

char str[10][10],f;

printf("Enter the number of productions\n");

scanf("%d",&n);

printf("Enter grammar\n");

for(i=0;i<n;i++)

scanf("%s",&str[i]);

for(i=0;i<n;i++)

{

f= str[i][0];

int temp=i;

if(isupper(str[i][3]))

{

repeat:

for(k=0;k<n;k++)

{

if(str[k][0]==str[i][3])

{

if(isupper(str[k][3]))

{

i=k;

goto repeat;

}

else

{

printf("First(%c)=%c\n",f,str[k][3]);

}

}

}

}

else

{

printf("First(%c)=%c\n",f,str[i][3]);

}

i=temp;

}

}

Output:-

[1210316530@CSELABServer2 ~]$ vi first.c

[1210316530@CSELABServer2 ~]$ cc first.c

[1210316530@CSELABServer2 ~]$ ./a.out

Enter the number of productions

3

Enter grammar

E->Tf

T->FE

F->(id)

First(E)=(

First(T)=(

First(F)=(

2. Write a aprogram to Find follow of the given grammar?

#include<stdio.h>

main()

{

int np,i,j,k;

char prods[10][10],follow[10][10],Imad[10][10];

printf("enter no. of productions\n");

scanf("%d",&np);

printf("enter grammar\n");

for(i=0;i<np;i++)

{

scanf("%s",&prods[i]);

}

for(i=0; i<np; i++)

{

if(i==0)

{

printf("Follow(%c) = $\n",prods[0][0]);

}

for(j=3;prods[i][j]!='\0';j++)

{

int temp2=j;

if(prods[i][j] >= 'A' && prods[i][j] <= 'Z')

{

if((strlen(prods[i])-1)==j)

{

printf("Follow(%c)=Follow(%c)\n",prods[i][j],prods[i][0]);

}

int temp=i;

char f=prods[i][j];

if(!isupper(prods[i][j+1])&&(prods[i][j+1]!='\0'))

printf("Follow(%c)=%c\n",f,prods[i][j+1]);

if(isupper(prods[i][j+1]))

{

repeat:

for(k=0;k<np;k++)

{

if(prods[k][0]==prods[i][j+1])

{

if(!isupper(prods[k][3]))

{

printf("Follow(%c)=%c\n",f,prods[k][3]);

}

else

{

i=k;

j=2;

goto repeat;

}

}

}

}

i=temp;

}

j=temp2;

}

}

}

Output:-

[1210316530@CSELABServer2 ~]$ vi follow.c

[1210316530@CSELABServer2 ~]$ cc follow.c

follow.c: In function ‘main’:

follow.c:25: warning: incompatible implicit declaration of built-in function ‘strlen’

[1210316530@CSELABServer2 ~]$ ./a.out

enter no. of productions

3

enter grammar

E->TF

T->FE

F->(id)

Follow(E) = $

Follow(T)=(

Follow(F)=Follow(E)

Follow(F)=(

Follow(E)=Follow(T)