

# Dept. of Computer Science & Engineering



আন্তর্জাতিক ইসলামী বিশ্ববিদ্যালয় চট্টগ্রাম  
الجامعة الإسلامية العالمية شيتاغونغ  
International Islamic University Chittagong

**Course Code:** CSE-3528

**Course Title:** Compiler Lab

## Spring-22

**Submitted By:**

**Name:** Farida Nusrat

**ID:** C201242

**Semester:** 5<sup>th</sup>

**Section:** 5AF

**Email:** [c201242@ugrad.iiuc.ac.bd](mailto:c201242@ugrad.iiuc.ac.bd)

**Submitted To:**

Mrs. Israt Binteh Habib

Dept. Of CSE, IIUC.

**Signature:**

Farida Nusrat

**Submission:** 11.11.22

**OBJECTIVE:**

Implement a program to calculate First and Follow sets of given grammar.

**RESOURCE:**

Code-blocks.

**Algorithm:****First ()-**

If  $x$  is a terminal, then  $FIRST(x) = \{ 'x' \}$

If  $x \rightarrow \epsilon$ , is a production rule, then add  $\epsilon$  to  $FIRST(x)$ .

If  $X \rightarrow Y_1 Y_2 Y_3 \dots Y_n$  is a production,

$FIRST(X) = FIRST(Y_1)$

If  $FIRST(Y_1)$  contains  $\epsilon$  then  $FIRST(X) = \{ FIRST(Y_1) - \epsilon \} \cup \{ FIRST(Y_2) \}$

If  $FIRST(Y_i)$  contains  $\epsilon$  for all  $i = 1$  to  $n$ , then add  $\epsilon$  to  $FIRST(X)$ .

**Follow ()-**

$FOLLOW(S) = \{ \$ \}$  // where  $S$  is the starting Non-Terminal

If  $A \rightarrow pBq$  is a production, where  $p$ ,  $B$  and  $q$  are any grammar symbols, then everything in  $FIRST(q)$  except  $\epsilon$  is in  $FOLLOW(B)$ .

If  $A \rightarrow pB$  is a production, then everything in  $FOLLOW(A)$  is in  $FOLLOW(B)$ .

If  $A \rightarrow pBq$  is a production and  $FIRST(q)$  contains  $\epsilon$ ,

then  $FOLLOW(B)$  contains  $\{ FIRST(q) - \epsilon \} \cup FOLLOW(A)$

**PROCEDURE:**

Go to debug -> run or press CTRL + F9 to run the program.

**PROGRAM:**

```
#include<bits/stdc++.h>
```

```
using namespace std;
```

```
int n,m=0,p,i=0,j=0;
```

```
char a[10][10],f[10];
```

```
void follow(char c);
```

```
void first(char c);
```

```
int main() {
```

```
    int i,z;
```

```
    char c,ch;
```

```

printf("Enter the no of prooductions:");

scanf("%d",&n);

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

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

    scanf("%s%c",a[i],&ch);

do {

    m=0;

    printf("Enter the elemets whose fisrt & follow is to be found:");

    scanf("%c",&c);

    first(c);

    printf("First(%c)={",c);

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

        printf("%c",f[i]);

    printf("}\n");

    strcpy(f," ");

    m=0;

    follow(c);

    printf("Follow(%c)={",c);

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

        printf("%c",f[i]);

    printf("}\n");

    printf("Continue(0/1)?");

    scanf("%d%c",&z,&ch);

} while(z==1);

```

```
    return(0);

}

void first(char c)

{

    int k;

    if(!isupper(c))

        f[m++]=c;

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

    {

        if(a[k][0]==c)

        {

            if(a[k][2]=='$')

                follow(a[k][0]);

            else if(islower(a[k][2]))

                f[m++]=a[k][2];

            else

                first(a[k][2]);

        }

    }

}

void follow(char c)

{

    if(a[0][0]==c)

        f[m++]='$';
```

```

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

{

    for(j=2; j<strlen(a[i]); j++)

    {

        if(a[i][j]==c)

        {

            if(a[i][j+1]!='\0')

                first(a[i][j+1]);

            if(a[i][j+1]=='\0' && c!=a[i][0])

                follow(a[i][0]);

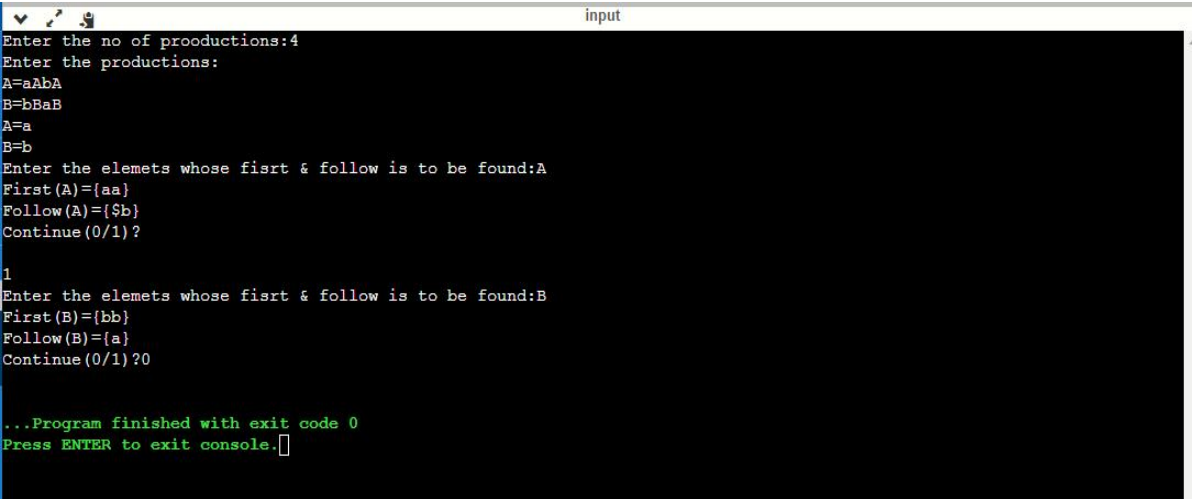
        }

    }

}

```

### **INPUT & OUTPUT:**



```

input
Enter the no of prooductions:4
Enter the productions:
A=aAbA
B=bBaB
a=a
b=b
Enter the elemets whose fisrt & follow is to be found:A
First(A)={aa}
Follow(A)={$b}
Continue (0/1)?
1
Enter the elemets whose fisrt & follow is to be found:B
First(B)={bb}
Follow(B)={a}
Continue (0/1)?0

...Program finished with exit code 0
Press ENTER to exit console.

```

### **OUTCOMES:**

The implementation is done successfully.