

NOIDA INSTITUTE OF ENGINEERING **AND TECHNOLOGY**



COMPILER DESIGN LAB (KCS - 552)

Department of Computer Science and Engineering

Submitted by:

Daravi Joshi (1901330100086)

Submitted to:

Mrs Arti Bahuguna

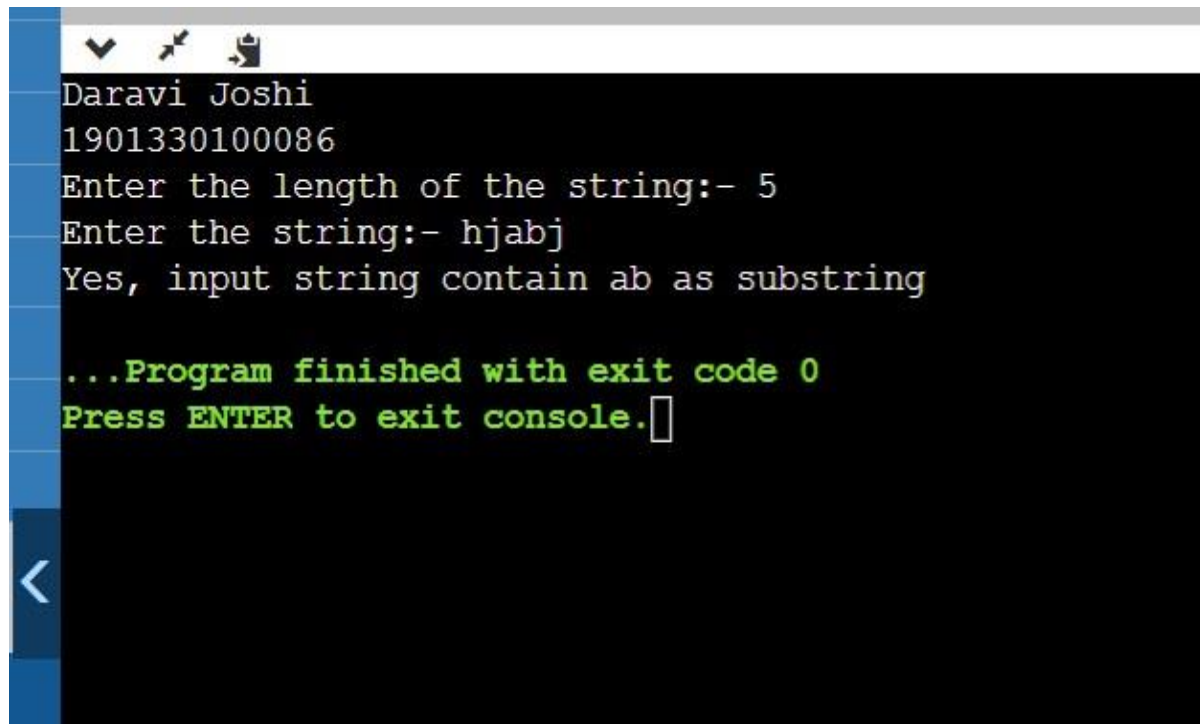
S.NO	Name of Practical	Date	Remark
	1. Write a program to check whether the string 'ab' substring or not.		
	2. Write a program to check whether the string starts with 'ab' or not.		
	3. Write a program to check whether string starts with 'a' and ends with 'b' or not.		
	4. Write a program to check whether the string is constant or not.		
	5. Write a program to check whether a string is keyword or not.		
	6). Write a program to check whether a string is identifier or not.		
	7). Write a program to check whether a string is a comment or not.		
	8). Write a program to construct recursive decent parser for the following grammar:-		
	9. Write a program to count spaces in string.		
	10: Write a program to find out identifiers, operators from a string.		
	11: Write a program to remove left factoring		
	12: Write a program to eliminate left recursion.		

Q1:- Wap to check whether the string contain ab as a substring or not.

CODE

```
#include<stdio.h>
#include<conio.h>
#include<string.h> void
main()
{
int n,i,j=0; char s[20];
printf("Daravi Joshi");
printf("\n1901330100086");
printf("\nEnter the length of the string:- ");
scanf("%d",&n);
printf("Enter the string:- "); for(i=0;i<=n;i++)
scanf("%c",&s[i]); for(i=0;i<=n;i++) if(s[i]=='a' &&
s[i+1]=='b') j++; if(j==1) printf("Yes, input string contain ab
as substring"); else printf("NO, input string does not
contain ab as substring"); getch();
}
```

Output

A screenshot of a C++ IDE's console window. The window has a dark background with a light blue sidebar on the left. The console text is as follows:
Daravi Joshi
1901330100086
Enter the length of the string:- 5
Enter the string:- hjabj
Yes, input string contain ab as substring

...Program finished with exit code 0
Press ENTER to exit console.
The text is in a monospaced font. The first four lines are in white, and the last two lines are in green. There is a white cursor at the end of the last line.
The sidebar on the left has a white back arrow icon.
The window title bar is at the top with standard OS icons (minimize, maximize, close).

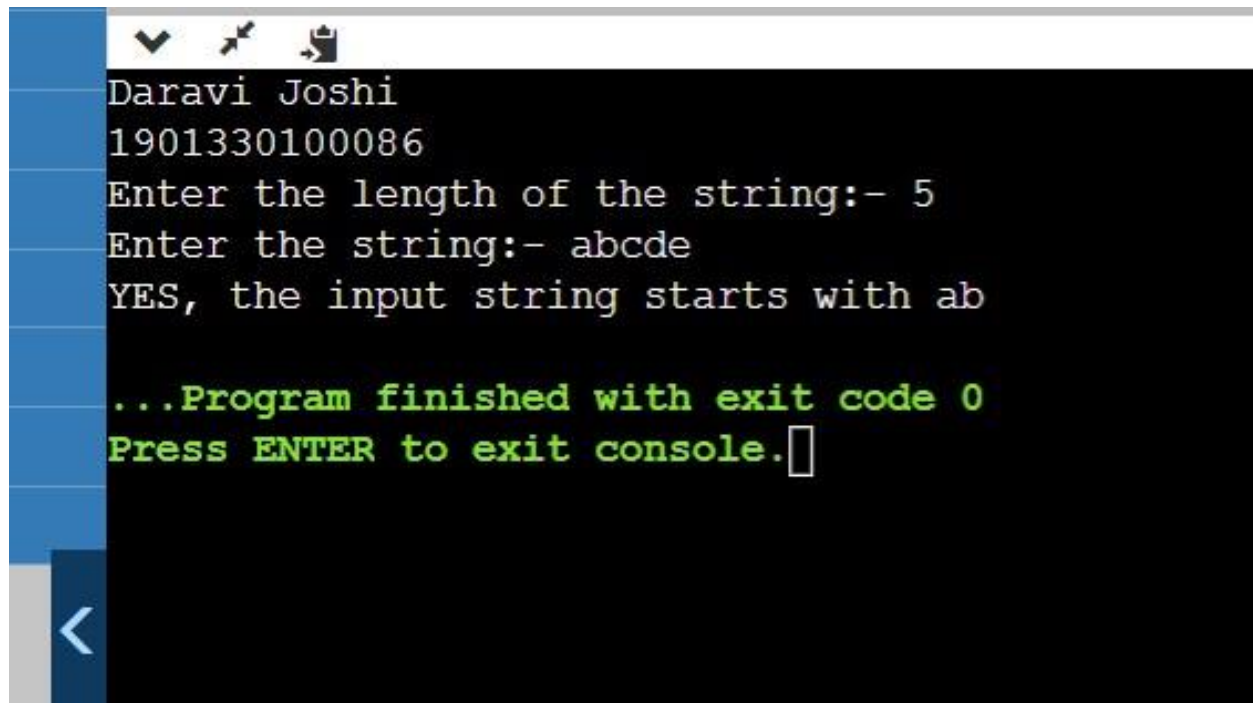
Q2:- Wap to check whether the given string start with ab or not .

CODE

```
#include<stdio.h>
#include<conio.h>
#include<string.h> void
main()
```

```
{  
int n,i;  
char s[20]; printf("Daravi  
Joshi");  
printf("\n1901330100086");  
printf("\nEnter the length of the string:- ");  
scanf("%d",&n);  
printf("Enter the string:- "); for(i=0;i<=n;i++)  
scanf("%c",&s[i]); if(s[1]=='a' && s[2]=='b')  
printf("YES, the input string starts with ab");  
else printf("NO, the input string not starts with  
ab"); getch();  
}
```

Output



```
Daravi Joshi
1901330100086
Enter the length of the string:- 5
Enter the string:- abcde
YES, the input string starts with ab

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

Q3:- Wap to check the string starts with 'a' and end with 'b'. [CODE](#)

```
#include<stdio.h>
```

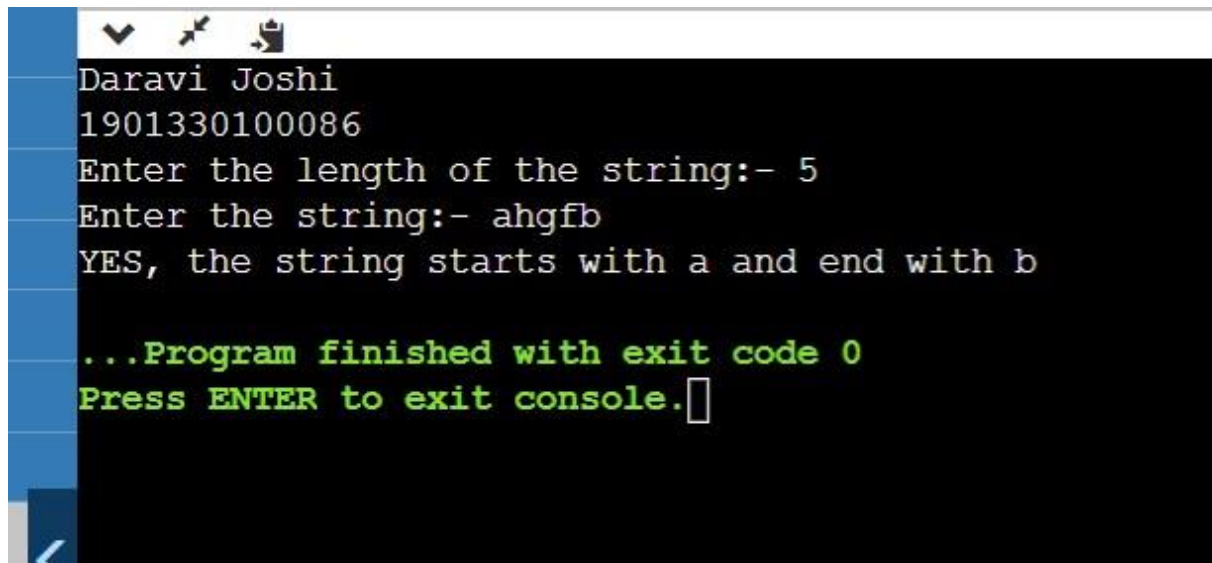
```
#include<conio.h>
```

```
#include<string.h> void
```

```
main()
```

```
{
int n,i;
char s[20]; printf("Daravi
Joshi");
printf("\n1901330100086");
printf("\nEnter the length of the string:- ");
scanf("%d",&n);
printf("Enter the string:- "); for(i=0;i<=n;i++)
scanf("%c",&s[i]); if(s[1]=='a' && s[n]=='b')
printf("YES, the string starts with a and end with b");
else printf("NO"); getch();
}
```

OUTPUT

A screenshot of a C++ console application. The window has a title bar with standard icons. The output text is as follows:

```
Daravi Joshi
1901330100086
Enter the length of the string:- 5
Enter the string:- ahgfb
YES, the string starts with a and end with b

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

Q-4: Wap to check whether string is constant or not. CODE

```
#include<stdio.h>

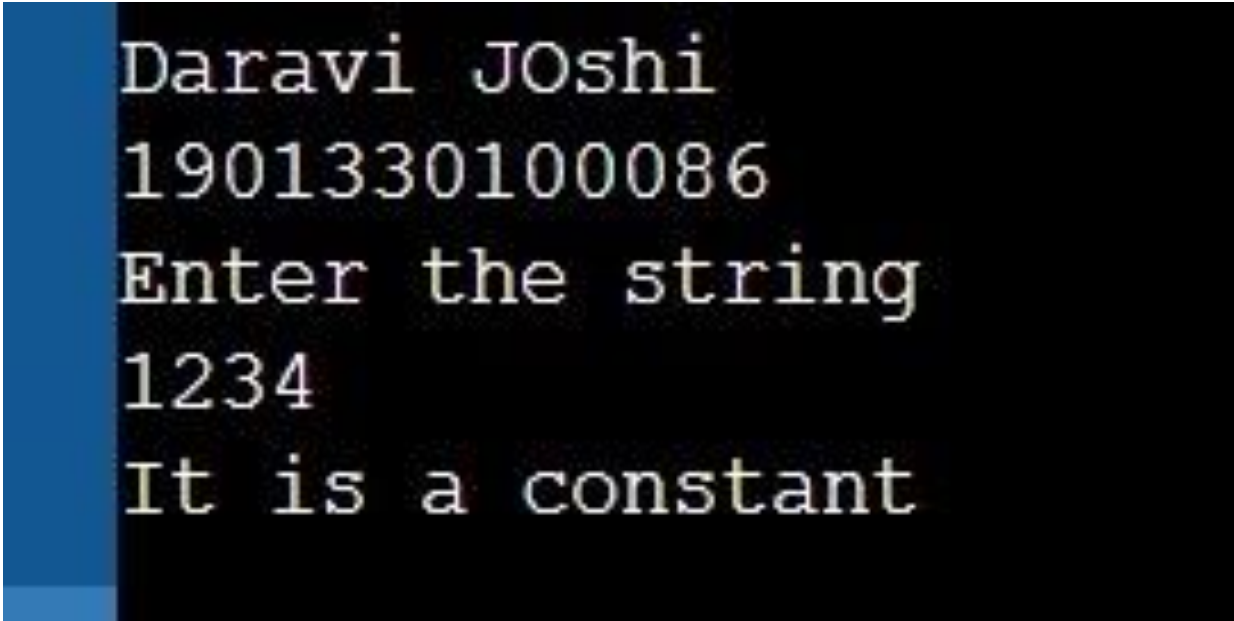
#include<conio.h>

#include<string.h> #include<ctype.h> void
main(){ char str[10]; int isConstant=0, a=0;
printf("Daravi Joshi \n 1901330100086\n");
printf("Enter the string\n"); gets(str); a=0;
while(str[a]){
if(isdigit(str[a]))
```



```
{  
isConstant= 1; break;  
}  
a++;  
}  
if(isConstant) printf("It is a constant"); else  
printf("String does not contain a constant");  
getch();  
}
```

OUTPUT

A screenshot of a terminal window with a black background and a blue vertical bar on the left. The text is displayed in a light green, monospaced font. The output shows the name 'Daravi JOShi', a long numeric string '1901330100086', a prompt 'Enter the string', the input '1234', and the final output 'It is a constant'.

```
Daravi JOShi  
1901330100086  
Enter the string  
1234  
It is a constant
```

Q-5: Wap to check whether string is keyword or not. **CODE**

```
#include <stdio.h>

#include <string.h> int

main() {

char keyword[32][10]={

"auto","double","int","struct","break","else","long",

"switch","case","enum","register","typedef","char",

"extern","return","union","const","float","short",

"unsigned","continue","for","signed","void","default",

"goto","sizeof","volatile","do","if","static","while"

};

char str [10];

printf("Daravi Joshi\n1901330100086\nEnter the enter the string");

gets(str); int flag=0,i;

for(i = 0; i < 32; i++) {

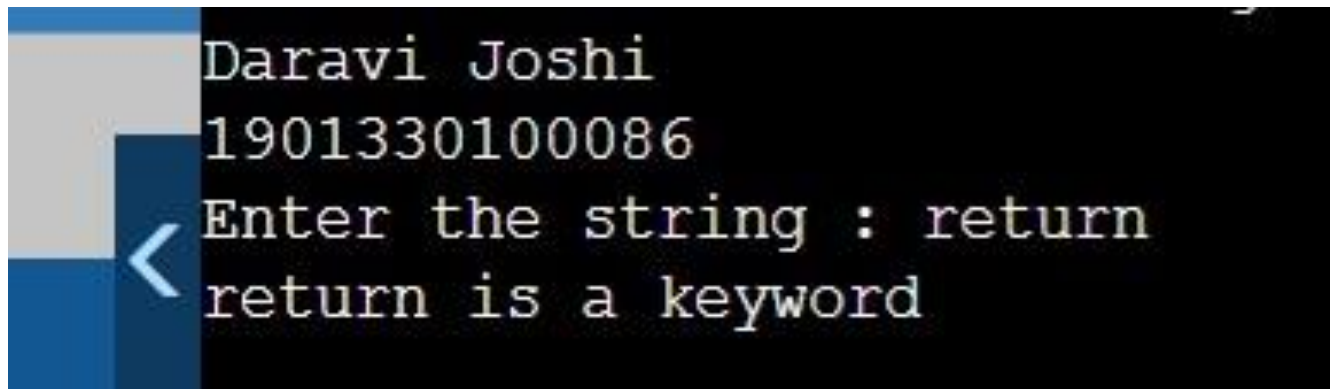
if(strcmp(str,keyword[i])==0) { flag=1;

}

}
```

```
if(flag==1) printf("%s is a  
keyword",str); else printf("%s is  
not a keyword",str);  
}
```

OUTPUT

A screenshot of a terminal window with a black background and yellow text. On the left side, there is a vertical bar with blue and grey segments. The text in the terminal reads: 'Daravi Joshi', '1901330100086', 'Enter the string : return', and 'return is a keyword'. A blue arrow points to the prompt 'Enter the string :'.

```
Daravi Joshi  
1901330100086  
< Enter the string : return  
return is a keyword
```

6).Write a program to check whether a string is identifier or not.

```
#include<stdio.h> #include<string.h> void main(){
    char str[100],i;  int count=0;  int t=0;  printf("Name = Daravi
Joshi");  printf("\nRoll no. = 1901330100086");  printf("\nCS-B");
printf("\nDate:-5/10/21");  printf("\nEnter a string:-");  gets(str);
if(!((str[0]>='a' &&
str[0]<='z') || (str[0]>='A'&&str[0]<='Z') || (str[0]=='_')) ) {
printf("\nGiven String is not valid identifer");    count=1;    t=1;
}
if(t==0){  for(i=1;i<strlen(str);i++){      if(!((str[i]>='a' &&
str[i]<='z') || (str[i]>='A'&&str[i]<='Z') || (str[i]>='0'&& str[i]<='9') || (str[i]=='_')) ) {
printf("\nGiven string is not valid identifer");    count=1;    break;
}
}
}
if(count==0) {    printf("\nGiven
string is valid identifier");
} }
```

OUTPUT:-

```
Name = Daravi Joshi
Roll no. = 1901330100086
CS-B
Date:-5/10/21
Enter a string:-KKKK

Given string is valid identifier
```

7).Write a program to check whether a string is a comment or not.

```
#include<stdio.h> #include<string.h> void main()

{   char str[100];   printf("Name = Daravi joshi");   printf("\nRoll no.-
1901330100086 ");   printf("\nCS-B");   printf("\nDate-26/10/21");

printf("\nEnter string:-");   gets(str);   int n = strlen(str);

if((str[0]=='/'&&str[1]=='/') || ((str[0]=='/' && str[1]=='*')&&(str[n-2]=='*&str[n-
1]=='/'))

{

    printf("\nGiven string is a comment");

}

else

{

    printf("\nGiven string is not a
comment");

}

}
```

OUTPUT:-

```
Name = Daravi joshi  
Roll no.- 1901330100086  
CS-B  
Date-26/10/21  
Enter string:-AJAJA  
  
Given string is not a comment
```

8).Write a program to construct recursive decent parser for the following grammar:-

$E \rightarrow TE'$

$E' \rightarrow +TE' / @$

$T \rightarrow FT'$

$T' \rightarrow *FT' / @$

$F \rightarrow (E) / ID$

Where @->null character.

```
#include<stdio.h>

#include<string.

h>

#include<ctype.

h> char

input[10]; int

i,error; void E();

void T(); void

Eprime(); void

Tprime(); void

F(); void main()

{ i=0; error=0; printf("\nName- Daravi Joshi"); printf("\nRoll

no.- 1901330100086 "); printf("\nCS-B"); printf("\nDate-

2/11/21"); printf("\nRecursive descent parsing for the

following grammar:-"); printf("\nE->TE'\nE'->+TE'/@\nT-

>FT'\nT'->*FT'/@\nF->(E)/ID"); printf("\nEnter the string to

be checked:"); gets(input); E();

if(strlen(input)==i&&error==0)

{

    printf("\nString is Accepted");

}

else

{

    printf("\nString is Rejected");

}

}
```

```

void E()
{
    T();
    Eprime();
}
void Eprime()
{
    if(input[i]=='+')
    {
        i++;
        T();
        Eprime();
    }
}
void T()
{
    F();
    Tprime();}
void Tprime() {

    if(input[i]=='*')
    {    i++;    F();
    Tprime();}}
void F()
{
    if(isalnum(input[i]
    )) i++; else
    if(input[i]=='(')

```



```
{ i++; E();  
if(input[i]=='')  
' i++; else  
error=1;}  
else error=1;  
}
```

OUTPUT:-

```
Name- Daravi Joshi  
Roll no.- 1901330100086  
CS-B  
Date-2/11/21  
Recursive descent parsing for the following grammar:-  
E->TE'  
E'->+TE'/@  
T->FT'  
T'->*FT'/@  
F->(E)/ID  
Enter the string to be checked:E  
String is Accepted
```

9. Write a program to count spaces in string.

```
#include<stdio.h>
#include<conio.h>
#include<ctype.h>
#include<string.h>

void main()
{
char ch[10]; int flag,count=0,i;
printf("Daravi Joshi");
printf("\nRoll no.-
1901330100086 "); printf("Enter
a string : "); gets(ch); for
(i=0;ch[i]!='\0';i++)
{
    if( ch[i]==' ') count++;
}
if(count>0) printf("String has spaces.\nNo. of
spaces are = %d",count); else printf("Spaces are
not present");
}
```

OUTPUT:-

Daravi Joshi

Roll no.- 1901330100086 Enter a string : J A V A

String has spaces.

No. of spaces are = 3

Write to

Program - 10: a program find out identifiers, operators from a string.

```
#include<stdio.h>

#include<conio.h> #include<conio.h>

void main(){ int i=0,j=0,k=0; char
str[20],id[30],op[30]; printf("Daravi
Joshi"); printf("\nRoll no.-
1901330100086 "); printf("Enter string
:"); gets(str); while(str[i]!='\0')
{
    if(isalnum(str[i]))
    { id[j]=str[i];
j++;} else
{
    op[k]=str[i];
op[k+1]=','; id[j]=',';
j++;
k+=2;
}
i++;
} id[j-1]='\0'; op[k-1]='\0';
printf("Identifiers are : ");
puts(id); printf("Opertors
are : "); puts(op);
}
```

OUTPUT:-

Write to

```
Daravi Joshi
Roll no.- 1901330100086 Enter string : A+B
Identifiers are : ,A,
Opertors are : ,+
```

Program -11: a program remove left factoring

```
#include<stdio.h> #include<string.h> int main(){ char
gram[20],part1[20],part2[20],modifiedGram[20],newGram[20],tempGram[20]; int i,j=0,k=0,l=0, pos;
printf("Daravi Joshi"); printf("\nRoll no.- 1901330100086 "); printf("Enter Production : A->"); gets(gram);
for(i=0;gram[i]!='|';i++,j++) part1[j]=gram[i]; part1[j]='\0'; for(j=++i,i=0;gram[j]!='\0';j++,i++)
part2[i]=gram[j]; part2[i]='\0'; for(i=0;i<strlen(part1) || i<strlen(part2);i++){ if(part1[i]==part2[i]) {
modifiedGram[k]=part1[i];
k++;
pos=i+1;
}}
for(i=pos,j=0;part1[i]!='\0';i++,j++){
newGram[j]=part1[i];
}
```

Write to

```
newGram[j++]='|';  
for(i=pos;part2[i]!='\0';i++,j++){  
    newGram[j]=part2[i];  
}  
modifiedGram[k]='X';  
modifiedGram[++k]='\0';  
newGram[j]='\0'; printf("\n A-  
>%s",modifiedGram); printf("\n X-  
>%s\n",newGram);  
}
```

OUTPUT:-

```
Daravi Joshi  
Roll no.- 1901330100086 Enter Production : A->aAB/aB
```

Write to

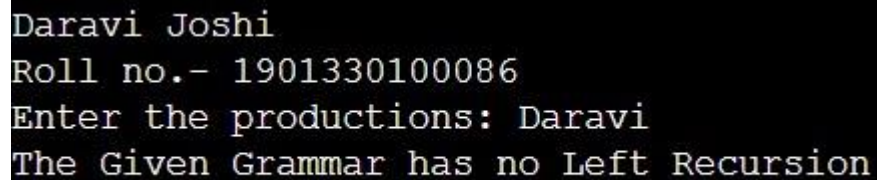
Program -12: a program eliminate left recursion.

```
#include<stdio.h>
#include<string.h>
void main()
{
char
input[100],l[50],r[50],temp[10],tempprod[20],productions[25][50
]; int i=0,j=0,flag=0,consumed=0; printf("Daravi Joshi");
printf("\nRoll no.- 1901330100086 "); printf("\nEnter the
productions: "); scanf("%1s->%s",l,r); printf("%s",r);
while(sscanf(r+consumed,"%[^|]s",temp) == 1 && consumed <=
strlen(r)) {
    if(temp[0] == l[0]) { flag = 1;
sprintf(productions[i++],"%s-
>%s%s'\0",l,temp+1,l);
}
else sprintf(productions[i++],"%s'-
>%s%s'\0",l,temp,l); consumed +=
strlen(temp)+1;
```

Write to

```
} if(flag == 1) { sprintf(productions[i++], "%s->\0", l);  
printf("\nThe productions after eliminating Left Recursion  
are:\n"); for(j=0; j<i; j++) printf("%s\n", productions[j]);  
}  
else printf("The Given Grammar has no Left  
Recursion");  
}
```

OUTPUT:-

A screenshot of a terminal window with a black background and white text. The text shows the output of a program: the name 'Daravi Joshi', a roll number '1901330100086', the prompt 'Enter the productions: Daravi', and the final output 'The Given Grammar has no Left Recursion'.

```
Daravi Joshi  
Roll no.- 1901330100086  
Enter the productions: Daravi  
The Given Grammar has no Left Recursion
```