NAME: AGNIV PRAMANICK **SECTION:** A **USN:** 1NT21IS017 **DATE:** 29/12/22

Q. Write a program to Convert PREFIX to POSTFIX.

THEORY

Prefix expression:

Another way to describe anything is with a prefix notation, which does not require knowledge about precedence or associativity but does when used with an infix notation. It is also known as **polish notation**. In prefix notation, an operator comes before the operands.

The syntax of prefix notation is given below:
<operator> <operand> <operand>

Example:

a b - c d -/

Postfix Expression:

When the operator is written after the operands, then it is known as **postfix notation**. Operand does not have to be always a constant or a variable; it can also be an expression itself.

The syntax for postfix notation is given below:

<operand> <operand> <operator>

Example:
/ - a b - c d

ALGORITHM

Step-1: Scan the prefix expression from right to left, i.e., reverse.
Step-2: If the incoming symbol is an operand, then push it into the stack.
Step-3: If the incoming symbol is an operator, then pop two operands from the stack. Once the operands are popped out from the stack, we add the incoming symbol after the operands. When the operator is added after the operands, then the expression is pushed back into the stack.
Step-4: Once the whole expression is scanned, pop and print the postfix expression from the stack.

CODE

```
//code starts

#include<stdio.h>

#include<conio.h>

#include<string.h>       //library function inserted

#include<stdlib.h>

# define MAX 20        // max size is 20

char str[MAX],stack[MAX];   //string of max size and char data type is used
```

```
int top=-1;
void push(char c)                //push function is defined
{
  stack[++top]=c;           //top is incremented
}
char pop()        //pop function is defined
{
  return stack[top--];   //top is decremented
}
void pretopost()    //converting from pre to post
{
  int n,i,j=0; char c[20];
  char a,b,op;
  printf("Enter the prefix expression\n");        //prefix expression is asked
  gets(str);
  n=strlen(str);
  for(i=0;i<MAX;i++)
  stack[i]='\0';
  printf("Postfix expression is:\t");
  for(i=0;i<n;i++)
  {
    if(str[i]=='+'||str[i]=='-'||str[i]=='*'||str[i]=='/')       //all four symbols are included
    {
      push(str[i]);     push function called
    }
    else
    { c[j++]=str[i];
      while((top!=-1)&&(stack[top]=='#'))
      {
         a=pop(); c[j++]=pop();        //pop function called
      }
```

```
        push('#');
      }
   }
   c[j]='\0';
   printf("%s",c);
}
main()      //main function starts from here
{
    pretopost();      //conversion function is called
}
//code ends
```

OUTPUT