

```

/*                                     while(    TOP!=-1          &&    precedence(stack[TOP])
Name: Alex Joseph                    ==precedence(ch)){
RollNo: 21121212[j++] = pop();
Date: 2025-07-14
*/ }
    push(ch);
}
#include<stdio.h>
#include<ctype.h>
#define(SIZE)100
postfix[j++] =pop();

char stack[SIZE];
int TOP = -1;

postfix[j] = '\0';
void push(char c){
}
if (TOP ==SIZE-1)
int evalPostfix(char postfix[]){
    else {
        int OPS[32];
        int stack[TOP] = c;
        int i =0;
    }
    int a,b;
    while((ch=postfix[i++]) != '\0'){
char pop(){
    if (isdigit(ch)){
        if(S2[TOP] != ch - '0';
        }return;
    else{
        else{ S2[top--];
        char S2[top] = stack[TOP];
        TOP--;
        return ch;
    }
    case '+': S2[++top] = b+a; break;
    case '-': S2[++top] = b-a; break;
    case '*': S2[++top] = b*a; break;
    case '/': S2[++top] = b/a; break;
    }
    if (c=='^') return 3;
    else if (c=='*' || c=='/') return 2;
    else if (c=='+' || c=='-') return 1;
    else return 0;
    return result;
}

void infixToPostfix(char infix[], char postfix[]){

    char postfix[SIZE] ,infix[SIZE];
    int i = 0, j=0;
    printf("Enter the infix value (don't enter white space)\n");
    for(i=0; infix[i] !='\0'; i++){
        scanf("%s", infix);
        ch = infix[i];
        infixToPostfix(infix, postfix);
        if(isdigit(ch))
            printf("%s\n", postfix);
        else if (ch=='('){
            push(ch);
        }
        else if (ch==')'){
            char poppedChar = pop();
            while(poppedChar !='('){
                postfix[j++] =poppedChar;
                poppedChar = pop();
            }
        }
        else {

```

```
/*
Output:
uploads\infix_to_postfix.c: In function 'pop':
uploads\infix_to_postfix.c:24:3:   warning:   'return'
with no value, in function returning non-void
    return;
    ^~~~~~

uploads\infix_to_postfix.c:21:6: note: declared here
char pop(){
    ^~~

uploads\infix_to_postfix.c: In function 'evalPostfix':
uploads\infix_to_postfix.c:89:9:
*/
```