4. Develop a Program in C for converting an Infix Expression to Postfix Expression
   Program should support for both parenthesized and free parenthesized expressions with the operators :
   + - * / % ^ and alphanumeric operands

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

#define SIZE 50

char s[SIZE],elem;
int top=-1;

void push(char elem){
        s[++top]=elem;
}

char pop(){
        return s[top--];
}

int pr(char elem){
        switch(elem){
                case '#':
                        return 0;
                case '(':
                        return 1;
                case '+':
                case '-':
                        return 2;
                case '*':
                case '/':
                case '%':
                        return 3;
                case '^':
                        return 4;
        }
        return 0;
}

void main(){
        char infx[SIZE],pofx[SIZE],ch;
        int i=0,k=0;

        printf("\nRead the Infix Expression\n> ");
        scanf("%s",infx);
        push('#');
        while((ch=infx[i++])!='\0'){
                if(ch=='(')
                        push(ch);
                else if(isalnum(ch))
                        pofx[k++]=ch;
                else if(ch==')'){
                        while(s[top]!='(')
                                pofx[k++]=pop();
                        elem=pop();
                }else{
                        while(pr(s[top])>=pr(ch))
                                pofx[k++]=pop();
                        push(ch);
                }
        }
        while(s[top]!='#')
                pofx[k++]=pop();
        printf("\nGiven Infix expression : %s",infx);
        printf("\nPostfix expression     : %s\n",pofx);
}
```