

EXPERIMENT 5

Infix to Postfix Conversion

Program:

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

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

```
char stack[100];
```

```
int top = -1;
```

```
void push(char x)
```

```
{
```

```
    stack[++top] = x;
```

```
}
```

```
char pop()
```

```
{
```

```
    if(top == -1)
```

```
    {
```

```
        return -1;
```

```
    }
```

```
    else
```

```
    {
```

```
    return stack[top--];  
    }  
}
```

```
int priority(char x)  
{  
    if(x == '(')  
    {  
        return 0;  
    }  
    if(x == '+' || x == '-')  
    {  
        return 1;  
    }  
    if(x == '*' || x == '/')  
    {  
        return 2;  
    }  
    return 0;  
}
```

```
int main()
{
    char exp[100];
    char *e, x;
    printf("Enter the expression : ");
    scanf("%s",exp);
    printf("\n");
    e = exp;

    while(*e != '\0')
    {
        if(isalnum(*e))
        {
            printf("%c ",*e);
        }
        else if(*e == '(')
        {
            push(*e);
        }
        else if(*e == ')')
        {

```

```
while((x = pop()) != '(')
{
    printf("%c ", x);
}
else
{
    while(priority(stack[top]) >= priority(*e))
    {
        printf("%c ",pop());
    }
    push(*e);
}
e++;
}

while(top != -1)
{
    printf("%c ",pop());
}
}
```

Output:

```
infix_to_postfix
Enter the expression : ((A+B)*(C+D))/(D+(E*F))
A B + C D + * D E F * + /
Press any key to continue . . .
```