

Infix to Postfix

```
#include <stdio.h>
#include <string.h>
#define SIZE 100
char stack[MAX], postfix[MAX], infix[MAX];
int top = -1;
void push(char symbol);
char pop();
int isEmpty();
int precedence(char symbol);
void display();
void inToPost();

int main() {
    printf("Enter Infix expression");
    gets(infix);
    inToPost();
    display();
    return 0;
}

void inToPost() {
    int i = 0, j = 0;
    char symbol, next;
    for (i < strlen(infix); i++) {
        symbol = infix[i];
        switch(symbol) {
            case '(': push(symbol);
            break;
            case ')':
                while ((next = pop()) != '(') {
                    postfix[j++] = next;
                }
                break;
        }
        postfix[j++] = symbol;
    }
}
```

case '+';
case '-';
case '*';
case '/';
case '^';

if (!is Empty()) & precedence (Stack [top]) >=

precedence (symbol) {

postfix[j++] = symbol;

}

push (symbol);

break;

default: postfix[j++] = symbol;

}}

while (!is Empty()) {

postfix[j++] = symbol;

postfix[j] = '\0';

}

void push (char symbol) {

top++;

Stack[top] = symbol;

}

char pop() {

char next = Stack[top];

top--;

return next;

}

int is Empty() {

return top == -1;

}

int precedence (char symbol) {

case '^': return 3;

case '/': case '*': return 2;


```
case '+': case '-': return 1;  
default: return 0;
```

```
}
```

```
void display() {
```

```
for (int i = 0; i < strlen(postfix); i++) {
```

```
printf("%c", postfix[i]);
```

```
}
```

```
}
```

Q: Enter infix exp:

44-9+6*2

447-62*+

Enter infix expression: 224+12-8*6/11

22412+86*11/-

Process returned 0 (0x0) execution time : 13.106 s

Press any key to continue.