

05/10/2020

```
#define size 10
char stack[size]; top = -1
```

```
push(char elem)
{
    stack[++top] = elem;
}
```

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

```
int pre(char symbol)
{
    if (symbol == 'n')
        return 3;
    else if (symbol == '*' || symbol == '/')
        return 2;
    else if (symbol == '+' || symbol == '-')
        return 1;
    else
        return 0;
}
```

```
main()
```

```
{
    while ((ch = infix[i]) != '\0')
    {
        if (exp[i] is operand
```

```
        postfix =
        res ← res + exp[i]
```

```
    else if (exp[i] is operator
```

```
        push(exp[i])
```

```
    while (!s.empty() && has higher pre
           (s.top() exp[i]))
    {
        res ← res + s.top()
        s.pop()
```

```
    }
    s.push(exp[i])
```

```
    }
}
```

```
else if (isOpeningParenthesis(exp[i]))  
    s.push(exp[i])
```

```
else if (isClosingParenthesis(exp[i]))  
{ while (!s.empty() && !isOpeningParenthesis(s.top()))  
    { res ← res + s.top()  
      s.pop()  
    }  
  s.pop()  
}
```

```
while (!s.empty())  
{ res ← res + s.top()  
  s.pop()  
}
```

```
return res  
}
```