

Infix to postfix using Stack

$a - b * d + c$

Infix \rightarrow

a	-	b	*	d	+	c	\0
---	---	---	---	---	---	---	----

postfix \rightarrow

--	--	--	--	--	--	--	--

```

char * InToPo(char *infix) {
    struct stack * sp; // Init the stack
    char * pos = (char *) malloc(strlen(infix+1) * sizeof(char));
    int i = 0; // Infix Scanner
    int j = 0; // Postfix fill
    while(infix[i] != '\0') {
        if (!isoperator(infix[i])) {
            postfix[j] = infix[i];
            i++; j++;
        }
        else {
            i++; j++;
        }
        else {
            if (prec(infix[i]) > prec(stackTop(st))) {
                push(sp, infix[i]);
            }
            else {
                postfix[j] = pop(sp);
                j++;
            }
        }
    }
}
    
```

```
        j++;  
    }  
    }  
    while (!isEmpty(sp)) {  
        postfix[j] = pop(sp);  
        j++;  
    }  
    postfix[j] = '\\0';  
    return postfix;  
}
```