

Infijo a posfijo

Escribir un programa que haciendo uso del **TAD PILA** y/o **TAD COLA** (#include <stack>, #include <queue> de solución al problema de recibir una expresión en notación infija y pasarla a notación postfija.

Explicación

La mayoría de los lenguajes para computadores tienen las expresiones aritméticas en notación in-fija debido a que el usuario utiliza esta notación, la cual no es la más apropiada para su evaluación por el computador. Por ello, normalmente se hace implícita o explícitamente la transformación a la notación posfija en el proceso de generación del lenguaje de máquina, para que el computador en el momento de la ejecución pueda evaluar la expresión de manera correcta y eficiente. Un método para hacer esta transformación de manera simplificada es utilizando el concepto de pila y cola.

Se supone que existe:

- Una cola de entrada, la cual contiene la expresión en notación infija
- Una cola de salida donde aparecerá la expresión en notación posfija
- Una pila (stack) para almacenar los operadores

Se debe aplicar las reglas sobre prioridad de operadores para las expresiones aritméticas:

Prioridad	Componentes de la expresión
5	Variables (mayor prioridad)
4	Signo exponencial
3	Signo de multiplicación y división
2	Signo de suma y resta
1	asignación

1. Las variables y los operadores provienen de la cola de entrada y se tienen uno por uno en notación infija para determinar en donde se almacenan
2. Cuando entra una variable, se coloca directamente en la cola de salida. Cuando entra un operador, va a la pila de operadores, saca de esta todos los operadores que tengan mayor o igual prioridad y los envía a la cola de salida. Esta acción termina porque:
 - Encuentra un operador de menor prioridad
 - Encuentra un paréntesis izquierdo
 - Encuentra el final de la pila

3. El operador que entra se coloca en la cabeza de la pila de operadores
4. Cuando entra un paréntesis izquierdo, pasa a la cabeza de la pila
5. Cuando entra un paréntesis derecho desocupa la pila hasta encontrar el paréntesis izquierdo correspondiente
6. Todos los operadores que salen de la pila de operadores pasan a la cola de salida. Al terminar la expresión la pila se desocupa y envía los operadores a la cola de salida, y termina el proceso.
7. Cuando se encuentran desocupadas la cola de entrada como la pila de operadores, en la cola de salida estará la expresión en posfijo.

Ejemplo 01

Expresión: $X = A * B - C / D * E$

Al iniciar el proceso en la cola de entrada está la expresión en infijo y la pila de operadores y la cola de salida están vacías. El proceso será:

Cola de Entrada	Pila de Operadores	Cola de salida
$X=A*B-C/D*E$	Vacía	Vacía
$A*B-C/D*E$	=	X
$B-C/D*E$	*=	XA
$C/D*E$	--=	XAB*
$D*E$	/--=	XAB*C
E	*--=	XAB*CD/
Vacía	Vacía	XAB*CD/E*--=

Otra forma de efectuar la transformación de notación infija a posfija es asignar prioridades de entrada y de salida a los operadores y los paréntesis. El programa se encarga de verificar las diferentes prioridades para determinar si un elemento permanece o sale de la pila de operadores.

Ejemplo 02

Expresión: $(X + Z) * W / T ^ Y - V$

Paso	Cola de Entrada	Símbolo analizado	Pila de Operadores	Cola de salida
0	$(X+Z)*W/T^Y-V$	Ninguno	Vacía	Vacía
1	$X+Z)*W/T^Y-V$	((Vacía
2	$+Z)*W/T^Y-V$	X	(X
3	$Z)*W/T^Y-V$	+	+(X
4	$) * W / T ^ Y - V$	Z	+(XZ
5	$* W / T ^ Y - V$)	(XZ+
)	Vacía	XZ+
6	$W / T ^ Y - V$	*	*	XZ+
7	$/ T ^ Y - V$	W	*	XZ+W
8	$T ^ Y - V$	/	/	XZ+W*
9	$^ Y - V$	T	/	XZ+W*T

10	$\gamma - V$	\wedge	$\wedge /$	$XZ + W^*T$
11	$-V$	γ	$\wedge /$	$XZ + W^*TY$
12	V	$-$	$/$	$XZ + W^*TY^\wedge$
	V	$-$	$-$	$XZ + W^*TY^\wedge /$
13	vacía	V	$-$	$XZ + W^*TY^\wedge / V$
14	vacía	ninguno	vacía	$XZ + W^*TY^\wedge / V -$