

Aplicaciones del TAD Pila

Ejercicios

- ▶ **Ejercicio 1:** Escribir una rutina que dada una **cadena de caracteres**, determine **si ésta es un palíndromo**. (Un palíndromo es un texto que se lee igual en los dos sentidos, ejemplos: Anita lava la tina, Dábale arroz a la zorra el Abad, radar, oso, ojo, Ana, etc.).

Algoritmo: Usar una PILA para introducir los caracteres (no introducir espacios en blanco). Al llegar al final del texto, ir sacando los caracteres y compararlos con el texto original; enviar el mensaje correspondiente.

Ejercicio 2:

Validar paréntesis correctos

- ▶ El compilador siempre sabe cuando se ha escrito un paréntesis, o una llave de mas (*revisión de sintáxis*).

¿Como lo hace?

Con el uso de pilas, expresiones escritas:

- $(a+b))$ *Mal*
- $((a+b) * c / 4*g-h)$ *OK*
- Se puede reconocer los paréntesis que no coinciden
- *¿Como lograr esta aplicación de la pila?*

- ▶ $7 - ((X * ((X + Y) / (J - 3)) + Y) / (4 - 2.5))$
- ▶ Cuando los paréntesis coinciden:
 - Al final de la expresión
 - Total paréntesis izq = Total paréntesis der y
 - En todo momento, en cualquier punto de la expresión
 - Cada paréntesis der. esta precedido de uno izq
 - Acum. paréntesis der. siempre es \leq que Acum.

• P.g.

$(A+B)) + 3$

En este punto de la expresión, ya se sabe que es incorrecta

Acum (= 1

Acum) = 2

No se cumple la regla

Al final de la expresión:

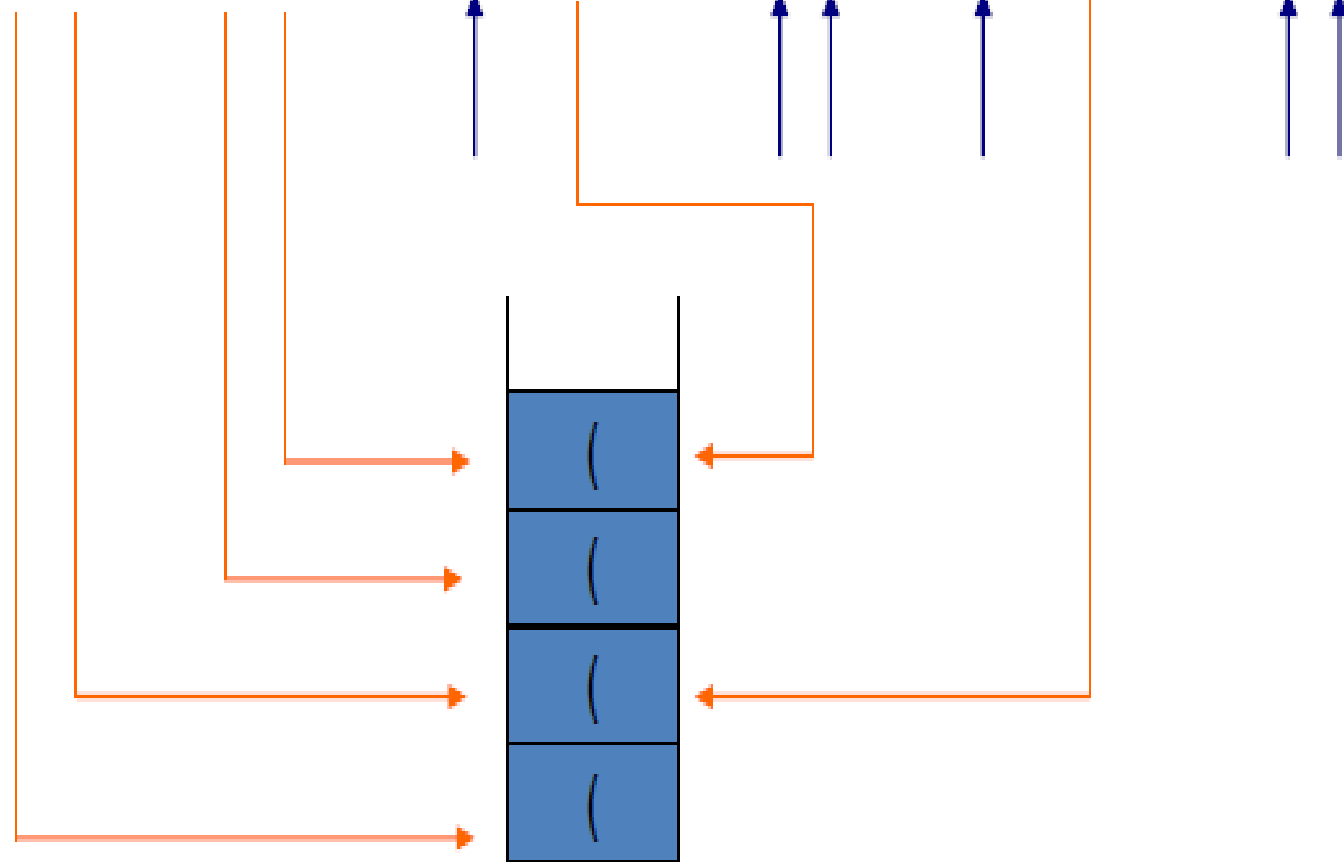
Total (= 1

Total) = 2

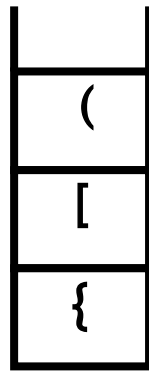
paréntesis izq

$$7 - ((X * ((X + Y) / (J - 3)) + Y) / (4 - 2))$$

Todos los),
encontraron su (



▶ $\{2^3 + [1 + 8 * (10 - 3)]\}$



()
[]
{ }

▶ $A * ((B - C) / 2)$

Revisión de Sintaxis:

Revisar la sintaxis de una expresión aritmética consistente de: operandos y operadores aritméticos, paréntesis anidados de tres tipos: `()`, `[]`, `{}`.

Determinar si la expresión está balanceada en cuanto al número de paréntesis (tantos paréntesis de apertura “ `(` , `[` , `{` ” como de cerradura “ `)` , `]` , `}` ”).

Emplear una pila para almacenar paréntesis de apertura; cuando se encuentre paréntesis de cerradura), retirar un elemento de la pila (paréntesis de apertura) y checar que sea el correspondiente al paréntesis de cerradura.

► Ejercicio 3: Conversión de una expresión de notación infija a postfija

Expresiones aritméticas

Una expresión aritmética contiene constantes, variables y operaciones con distintos niveles de precedencia.

Operaciones básicas:

\wedge potencia

$*/$ multiplicación, división

$+,-$ suma, resta

Expresiones infijas, prefijas y postfijas

Notación infija:

Los operadores aparecen en medio de los operandos.

$A + B$, $A - 1$, E/F , $A * C$, $A ^ B$, $A + B + C$, $A+B-C$

Notación prefija

El operador aparece antes de los operandos.

$+ AB$, $- A1$, $/EF$, $*AC$, AB , $+AB+C$, $+AB-C$

Notación postfija:

El operador aparece al final de los operandos.

$AB+$, $A1-$, $EF/$, AC^* , $AB^$, $AB+C+$, $AB+C-$

Conversión de una expresión de notación infija a postfija

- ▶ Condiciones
- ▶ Solamente se manejarán los siguientes operadores :
 - \wedge potencia
 - $*$ / multiplicación y división
 - $+$ $-$ suma y resta
- ▶ Los operadores de más alta prioridad se ejecutan primero
- ▶ Si hubiera una expresión con dos o más operadores de prioridad, entonces se procesan de izquierda a derecha.
- ▶ Las subexpresiones parentizadas tendrán más prioridad que cualquier operador.

Ejemplos

1) $X + Z * W$

paso 0: $X + Z * W$

1: $X + Z W *$

2: XZW^*+

2) $(X + Z) * W / T \wedge Y - V$

paso 0: $(X + Z) * W / T \wedge Y - V$

1: $XZ + *W / T \wedge Y - V$

2: $XZ + *W / TY \wedge - V$

3: $XZ + W^* / TY \wedge - V$

4: $XZ + W^* TY \wedge / - V$

5: $XZ + W^* TY \wedge / V -$

Ejercicios

1. $A * B / (A + C)$

2. $4 * (5 + 6 - (8 / 2^3) - 7) - 1$

3. $6 * B + 8 * 45$

4. $6 * 5^{(4 + 3 * 6)} + 5 * 8^{(4 * 7)}$

Ejemplos

1) $X + Z * W$

paso 0: $X + Z * W$

1: $X + Z W *$

2: XZW^*+

2) $(X + Z) * W/T \wedge Y - V$

paso 0: $(X + Z) * W/T \wedge Y - V$

1: $XZ + *W/T \wedge Y - V$

2: $XZ + *W/TY \wedge - V$

3: $XZ + W^*/TY \wedge - V$

4: $XZ + W^*TY \wedge / - V$

5: $XZ + W^*TY \wedge / V -$

SOLUCION

EJERCICIO 1) $A*B / (A+C)$

= $A*B / AC+$

= $AB^* / AC+$

= $AB^* AC+ /$

- Conversión de una expresión Infija a Postfija mediante una pila

Ejemplo : expresión infija: $A*B / (A+C)$

expresión postfija: $AB*AC+ /$

Algoritmo “infija_a_postfija “

1. Inicializar la pila

2. Repetir hasta que no haya caracteres en la expresión de entrada

2.1 Leer un carácter de la expresión

2.2 Si es un operando se pasa a la expresión postfija de salida

2.3 Si el elemento es un operador distinto de ‘)’ entonces:

2.3.1 Si la pila está vacía se mete en la pila.

2.3.2 Si la pila NO está vacía

- Si la prioridad del operador es mayor que la prioridad del operador de la cima de la pila \Rightarrow se mete en la pila
- Si la prioridad del operador es menor o igual que la prioridad del operador de la cima de la pila \Rightarrow se saca el operador de la cima y se coloca en la expresión postfija. Volvemos a 2.3

2.4 Si el elemento es el operador ‘)’ entonces:

2.4.1 Se sacan operadores de la pila hasta encontrar el ‘(’ paréntesis que se elimina (las expresiones postfijas no llevan paréntesis)

3. Al finalizar el recorrido por la expresión aritmética se pasa todo el contenido de la pila a la expresión postfija

Conversión de Infija a Postfija

Ejemplo : expresión infija: $A*B / (A+C)$

expresión postfija: $AB*AC+/$

– Prioridad de los operadores

Operador	Prioridad en la expresión infija	Prioridad en la pila
\wedge	4	3
$*$	2	2
$/$	2	2
$+$	1	1
$-$	1	1
$($	5	0
$)$	no definida	no definida

– Notas:

- La prioridad de la potencia es menor en la pila que en la expresión infija para evaluar varios operadores de potenciación de derecha a izquierda (se evalúa primero lo último encontrado)
- Esta variación no afecta a los otros operadores ya que la prioridad de la potencia siempre es mayor
- El paréntesis izquierdo pasa a tener prioridad cero ya que sólo se extrae de la pila (para eliminarlo) cuando aparece un paréntesis derecho.
- La prioridad del paréntesis derecho no está definida porque nunca entra a formar parte de las comparaciones

Conversión de Infija a Postfija

Ejemplo 1 : expresión infija: $A * B / (A + C)$

expresión postfija: $AB * AC + /$

Cima

Carácter leído	Acción	Pila	Expresión postfija
A	Pasar a postfija		A
*	Meter('*')	*	A
B	Pasar a postfija	*	AB
/	$PInfija('/') \leq PPila('*')$ \Rightarrow Sacar \rightarrow ('*') y pasamos a postfija Como la pila está vacía \Rightarrow Meter ('/')	/	AB*
($PInfija('(') > PPila('/') \Rightarrow$ Meter('(')	(, /	AB*
A	Pasar a postfija	(, /	AB*A
+	$PInfija('+') > PPila('(') \Rightarrow$ Meter ('+')	+, (, /	AB*A
C	Pasar a postfija	+, (, /	AB*AC
)	Pasamos los elementos de la pila a postfija hasta encontrar '(' que se elimina	/	AB*AC+
Fin entrada	Pasamos todo lo que queda en la pila a la expresión postfija		AB*AC+ /

Conversión de Infija a Postfija

Ejemplo 2 : expresión infija: $4*(5+6-(8/2^3)-7)-1$

Carácter leído	Acción	Pila	Expresión postfija
4	Pasar a postfija		4
*	Meter('*')	*	4
($Plnfija('(') > PPila('*') \Rightarrow \text{Meter}('(')$	(, *	4
5	Pasar a postfija	(, *	45
+	$Plnfija('+') > PPila('(') \Rightarrow \text{Meter}(' +')$	+, (, *	45
6	Pasar a postfija	+, (, *	456
-	$Plnfija('-') \leq PPila(' +')$ $\Rightarrow \text{Sacar} \rightarrow (' +')$ y pasamos a postfija $Plnfija('-') > PPila('(') \Rightarrow \text{Meter}(' -')$	-, (, *	456+
($Plnfija('(') > PPila('-') \Rightarrow \text{Meter}('(')$	(, -, (, *	456+
8	Pasar a postfija	(, -, (, *	456+8
/	$Plnfija('/') > PPila('(') \Rightarrow \text{Meter}('/')$	/, (, -, (, *	456+8
2	Pasar a postfija	/, (, -, (, *	456+82
^	$Plnfija('^') > PPila('/') \Rightarrow \text{Meter}('^')$	^, /, (, -, (, *	456+82
3	Pasar a postfija	^, /, (, -, (, *	456+823
)	Pasamos los elementos de la pila a postfija hasta encontrar '(' que se elimina	-, (, *	456+823^/
-	$Plnfija('-') \leq PPila('-')$ $\Rightarrow \text{Sacar} \rightarrow (' -')$ y pasamos a postfija $Plnfija('-') > PPila('(') \Rightarrow \text{Meter}(' -')$	-, (, *	456+823^/-
7	Pasar a postfija	-, (, *	456+823^/-7
)	Pasamos los elementos de la pila a postfija hasta encontrar '(' que se elimina	*	456+823^/-7-
-	$Plnfija('-') \leq PPila('*')$ $\Rightarrow \text{Sacar} \rightarrow ('*')$ y pasamos a postfija Como la pila está vacía $\Rightarrow \text{Meter}(' -')$	-	456+823^/-7-*
1	Pasar a postfija	-	456+823^/-7-*1
Fin entrada	Pasamos todo lo que queda en la pila a la expresión postfija		456+823^/-7-*1-

Fuentes de información

- Estructuras de Datos, Cairo – Guardati, Mc Graw Hill
- Estructuras de Datos, Román Martínez, Elda Quiroga, Thomson Learning
- ▶ <https://docplayer.es/2154070-Estructura-de-datos-y-de-la-informacion-pilas-y-expresiones-aritmeticas.html>