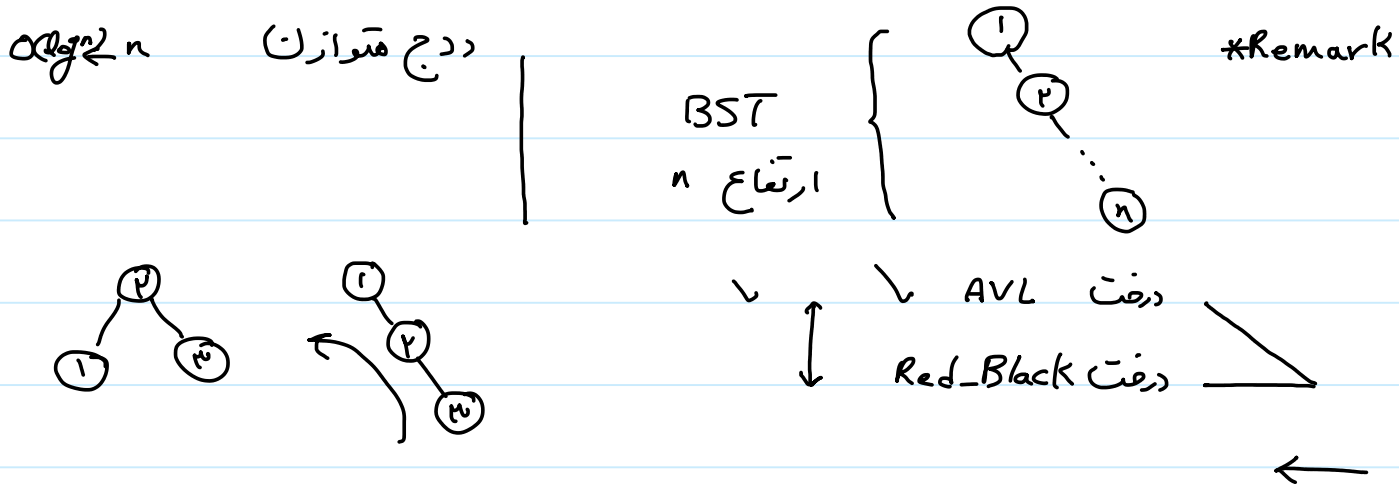


موضوع: درخت عبارت



$$2+5 \quad 2+5*7$$

$$3+(9*7^2)$$

عبارت‌های ریاضی

$$a + b * c + d$$

عملوند عملر عملوند

هر عبارت ریاضی ۲ تا n دارد: عملر و عملوند.

$$(a + b)$$

عبارت میان‌بندی: عملرها بین عملوندها هستند

$$a + b * c$$

* مسطح عبارت میان‌بندی، ابهام:

برای رفع ابهام، اولویت‌ها تعریف شده است:

+ - * / ^

- ۱- توان $a^b \leftarrow \underbrace{a^b}_c$ [راست به چپ]
- ۲- ضرب و تقسیم $a * b * c / d * e$ [چپ به راست]
- ۳- جمع و تفریق $\underbrace{a * (b + c)}_{\text{پارانتز}}$ [چپ به راست]

عبارت میانوندی پرانتزگذاری شده.

$$\left(\left(\underline{(a * b)} + \underline{(c * d)} \right) - \underline{(e^f)} \right) =$$

$$\boxed{2} \quad a + b * c - d * e^f \rightarrow ((a + (b * c)) - (d * (e^f)))$$

به غیر از عبارت های میانوندی، ۲ نوع عبارت دیگر وجود دارد:

$$+ a * b c \leftarrow a + b * c \quad \text{۱- پسوندی} \quad + a b$$

$$\underline{a b c * +} \leftarrow \underline{a + b * c} \quad \text{۲- پسوندی} \quad a b +$$

$$a b c + * \quad \underbrace{a b * c}_+ \quad a * (b + c)$$

$$\boxed{1} \quad \underline{a * (b + c^d)} - \underline{e / f}$$

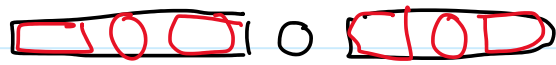
پسوندی: $- * a + b^c d / e f$

پسوندی: $\backslash a b c d^+ * e f / -$

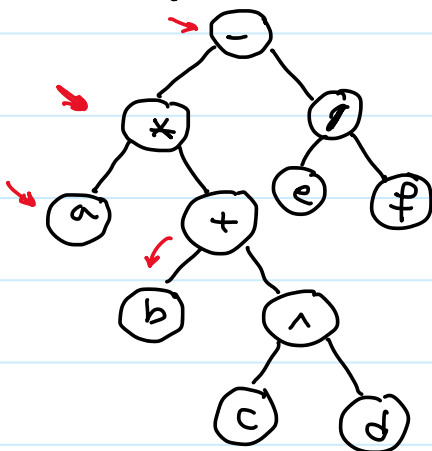


$((a * (b + (c^d))) - (e * f))$

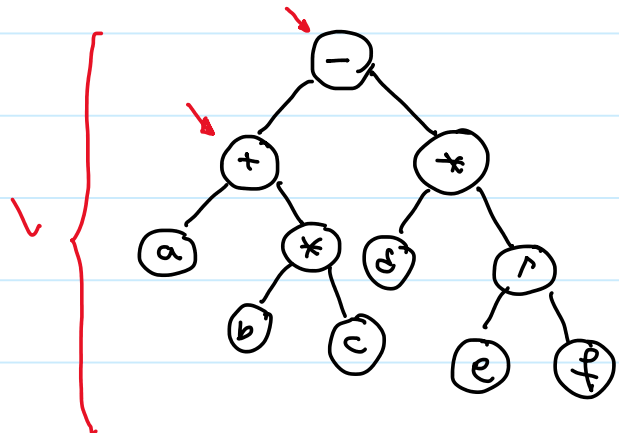
\checkmark } $- * a + b \wedge c d / e f$
 \checkmark } $a b c d \wedge + * e f / -$



$$\boxed{\frac{a \times (b + c \cdot d) - e}{f}}$$



□ $a + b * c - d * e^f$



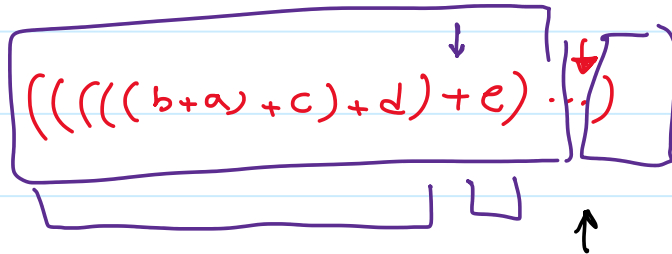
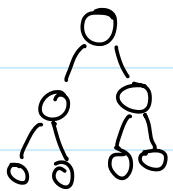
۲- تره های داخلی محللر و لب ها محللرند محللر + محللرند

✓ $- * a + b / c d / e f$: Preorder
 ✓ $a b c * + d e f / * -$

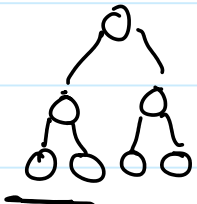
الگوریتم تبدیل عبارت به درخت عبارت [فرض کنید ورودی پرانتز گذاری کامل شده]

۱- عملگر آخر را پیدا و در ریشه قرار می دهیم. $\rightarrow O(n)$

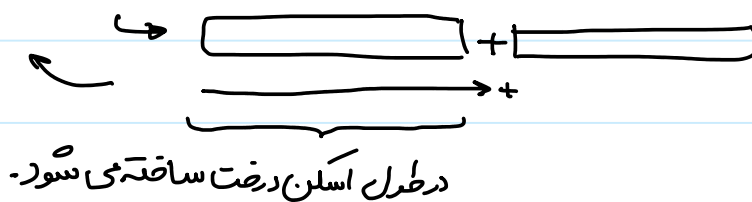
۲- به طور بازگشتی سمت راست را در زیر درخت هاچپ و سمت چپ را در زیر درخت راست می گذاریم \rightarrow



$$\underbrace{T(n) = T(n-1) + O(n)}_{\Omega} = \underbrace{T(n) = O(n^2)}_{\Omega}$$



* این الگوریتم را می توان در $O(n)$ هم پیاده کرد.



میانودی پسوندی
 الگوریتم تبدیل infix به postfix . با استفاده استک .

* ورودی را از چپ به راست بخوان

۱- اگر در ورودی محموله قرار داشت : بده به خروجی

۲- اگر محمله :

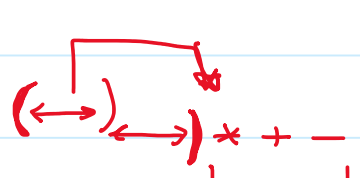
بده

۲. اگر عملگر:

- ا. تا زمانی که اولویت عملگر بالای استک بیشتر یا مساوی عملگر فعلی است، pop، به خروجی بده
- ب. عملگر را وارد استک بنما.

ورودی: $a + b * c - d * e ^ f$

ورودی	استک	خروجی
a		a
+	+	a
b	+	ab
*	^{Top} * +	abc
c	* +	abc
-	-	abc * +
d	-	abc * + d
*	* -	abc * + d
e	* -	abc * + d e
^	^ * -	abc * + d e
f	^ * -	abc * + d e f
		abc * + d e f ^ * -



? ← ()

(← پویش استک

(← ٹاپلائٹر بائریوٹ ، POP کٹم وہ خروچی دھم
لے)