

HW1.

#5. If $n_1 = 4096$, $f(n) = 512$ ms. $\downarrow 16$.

If $n_2 = 16384$, $f(n_2) = 8192$ ms.

$f(n) = ?$ $O(\quad)$



$(\log n)$

$O(n)$

$O(n^2)$, $O(n^3)$...

$O(2^n)$

$O(n!)$

\vdots

$$n_2 = 16384 = 4 * n_1$$

$$f(n_2) = 16 * f(n_1)$$

$$\Rightarrow f(4 * n_1) = 16 * f(n_1) \quad - \textcircled{1}$$

3. $\log(n)$

$O(\log(n))$

1000
(constant)

smaller = ? $O(1)$.

$O(1)$

Recursive Algorithms.

- 1) Prefix to Postfix Conversion
- 2) Towers of Hanoi.

Arithmetic expression can be represented in 3 diffⁿ formats - Eg.

- 1) Infix $A + B$
- 2) Prefix $+ AB$
- 3) Postfix $AB +$

$$\begin{array}{c} \downarrow \quad \downarrow \\ A + \{ B * C \} \\ + \underline{A} * \underline{BC} \\ \underline{A} \underline{BC} * + \end{array}$$

Goal: Use recursion to analyze the order in which expressions are to be evaluated.

$* AB$ $\xrightarrow[\text{(using recursion)}]{\text{prefix to postfix conversion}}$ $AB *$

↓ nested expressions may be present.

1st operand 2nd operand

$$\boxed{A * B} + C - \boxed{E / F} \rightarrow \text{Infix.}$$

Prefix: $- + * AB C / EF$

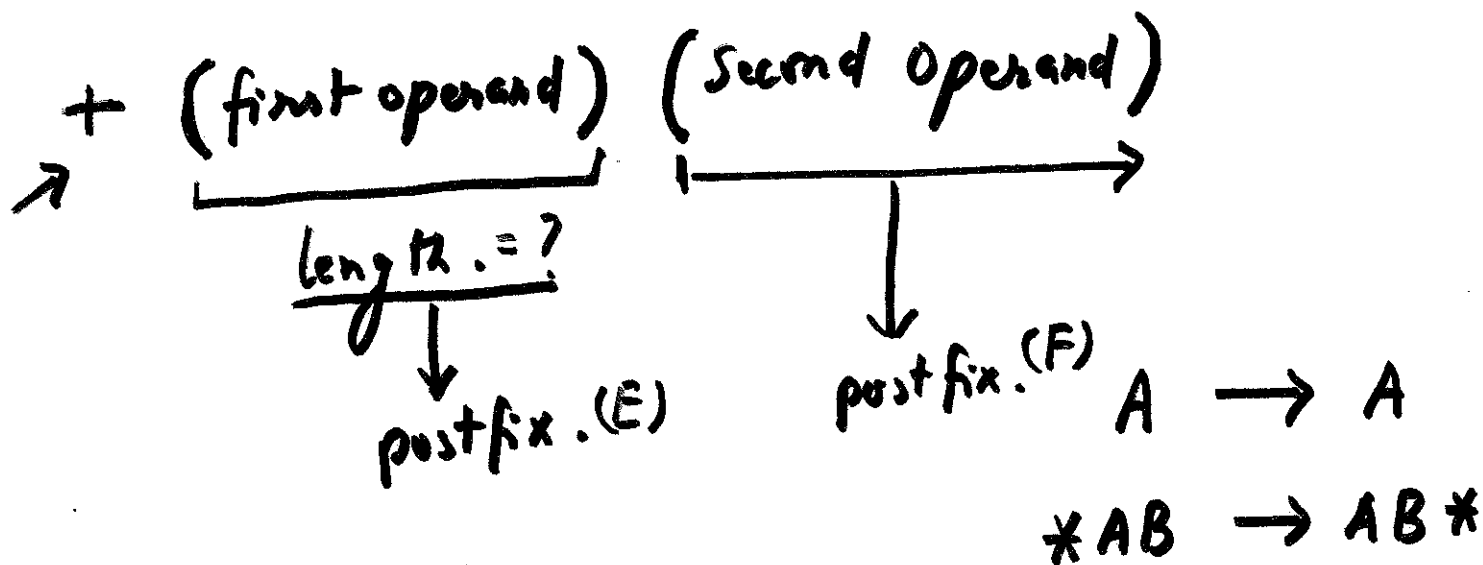
There are only four possible operands } $- , + , * , /$ (Assumption)

postfix: $\underline{AB * C} + \underline{EF} / -$

Q. What is the base case?

finding an operand.

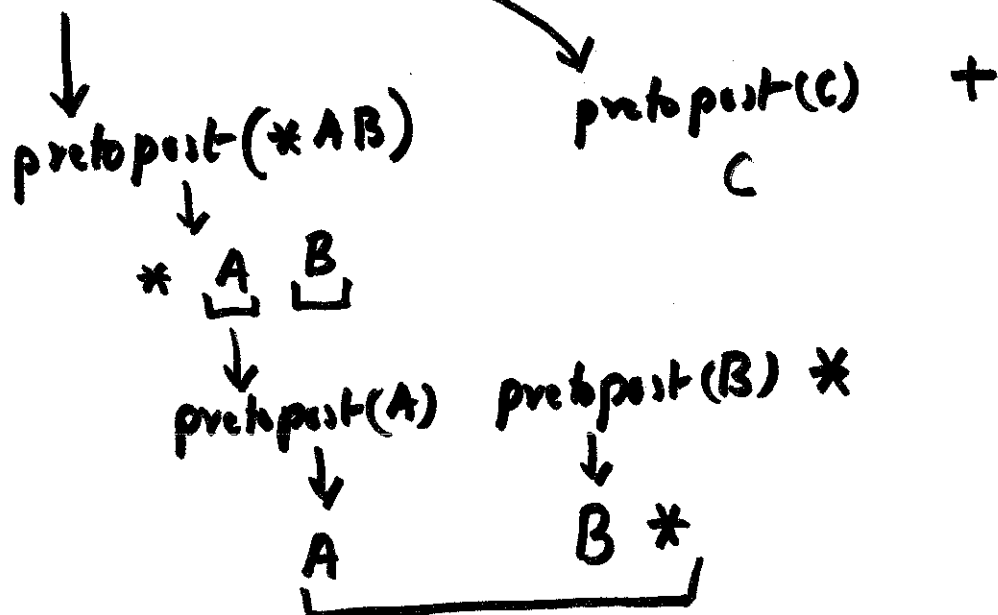
General Case: finding an operator with its left and right operands.



$E F +$

Ex. $+ * A B C$

$+ (*AB) (C)$



Pre to Post Algorithm.

① (Base case) if (expr[0] == operand)
[equivalent to length(expr) = 1]
return expr[0];

② (General case)

1. $oper = expr[0] \leftarrow$

1. Oper = operand 1
2. Decompose the remaining expressions into operand 1 and operand 2.

(Use f^n to compute length)

3. PF 1 = prepost (operand 1)
PF 2 = prepost (operand 2)

4. return [PF1 \rightarrow PF2 \rightarrow oper];

Ex. * A, B
 $\text{len } 1 = \text{find len } (A, B) \rightarrow \text{return } 1$
 $\text{len } 2 = \text{find len } (B) \rightarrow \text{return } 1.$
 $+ \frac{1}{3}$

Ex.

$$A * B * C + D$$

$$+ * * A B C D \longrightarrow A B * C * D +$$

$$\text{findLength}(* * A B C D) \quad (5)$$

$$\text{gcase: } * \quad \text{findLength}(* A B C D) \rightarrow \text{len1}$$

$$\rightarrow \underline{\text{len2}}$$

$$\underline{\text{expr} + 1 + \text{len1.}}$$

Ex.

$$A * B + C$$

$$+ * A B C \longrightarrow \underline{A B * C +}$$

$$\downarrow$$
$$\underline{\text{length} \rightarrow \text{return } 3}$$

$$\underline{* A B} \rightarrow \text{postfix} \rightarrow \underline{\text{PF1}}$$

first operand

$$\text{second operand} = C \rightarrow \text{postfix} \rightarrow \underline{\text{PF2}}$$

$$\text{PF2} = \underline{C}$$

$$* A B \rightarrow \text{PF1} = \underline{A B *}$$

$$\text{PF1} \rightarrow \text{PF2} \rightarrow +$$

$$A B * C +$$