

逆波兰表示法和表达...

逆波兰式(后缀式)

- 1. 后缀式(逆波兰式)
- 2. 抽象语法树AST. 有向无环图 DAG.
- 3. 三地址码
 - 四元式
 - 三元式
 - 间接三元式

逆波兰表示法 (后缀表示法) 表达式 $E = E_1 \text{ op } E_2 \Rightarrow E_1 E_2 \text{ op}$

eg: $\frac{(a+b) * (a-c) - d}{x+y}$ 翻译成逆波兰表示

原式 = $E_1 E_2 = ab+ac-xd-xy+*$

$E_1 = \frac{(a+b) * (a-c) - d}{x+y} = E_3 \frac{d}{x+y} = ab+ac-xd-$

$E_2 = x+y = xy+$

$E_3 = \frac{(a+b) * (a-c)}{x+y} = E_4 E_5 * = ab+ac-x*$

$E_4 = a+b = ab+$

$E_5 = a-c = ac-$

运算符 "自外向内" ↙

存在运算符的运算顺序一般为 \neg, \wedge, \vee

表达式 $\frac{\neg(A \vee B) \wedge (C \vee D)}{E_1 E_2}$ 的逆波兰表示为

$E_1 E_2 \wedge = A \neg B \vee C \vee D \wedge$

$E_1 = \neg A \vee B = (\neg A) \vee B = A \neg \vee B$

$E_2 = C \vee D = C \vee D$

四、试写出表达式 $A * (B * D - C / A) + B$ 的后缀式和四元式中间代码。(8分)

$A * (B * D - C / A) + B = E_1 B + = ABD * CA / - * B +$

$E_1 = A * (B * D - C / A) = A E_2 * = ABD * CA / - *$

$E_2 = \frac{B * D - C / A}{x+y} = E_3 E_4 - = BD * CA / -$

$E_3 = B * D = BD *$

$E_4 = C / A = CA /$

表达式 { 后缀式: 由外到内
四元式: 由内到外 (计算过程)

$A * \frac{B * D - C / A}{x+y} + B$

1 $(*, B, D, t_1)$
2 $(/, C, A, t_2)$
3 $(-, t_1, t_2, t_3)$
4 $(*, A, t_3, t_4)$
5 $(+, t_4, B, t_5)$
6 $(=, t_5, -, y)$

日常使用

5) 后缀式 \neg 对应的中缀表达式是 $a - (-b) * c$ (注: @表示求负运算)。
A. $a - b @ c *$ B. $ab @ - c *$ C. $ab - c @ *$ D. $ab @ c * -$

6) 后缀式 $ab + cd + /$ 可用中缀表达式 B 来表示。
A. $a + b / c + d$ B. $(a + b) / (c + d)$ C. $a + b / (c + d)$ D. $a + b + c / d$

7) 表达式 $(a + b) * c$ 的后缀表达式为 C 。
A. $ab * c +$ B. $abc * +$ C. $ab + c *$ D. $abc + *$

8) 中间代码生成时所依据的是 C 。
A. 语法规则 B. 词法规则 C. 语义规则 D. 等价变换规则

$a - (-b) * c$
 $a b @ c * -$
 $ab + cd + /$
 $(a + b) / (c + d)$
 $(a + b) * c$
 $ab + c *$

课本 P100-P111

四元式

能存放 tmp.

$a := b * -c + b * -c$

op	arg1	arg2	result
(1)	$-c$		T_1
(2)	b	T_1	T_2
(3)	$-c$		T_3
(4)	b	T_3	T_4
(5)	T_2	T_4	T_5
(6)	T_5		a

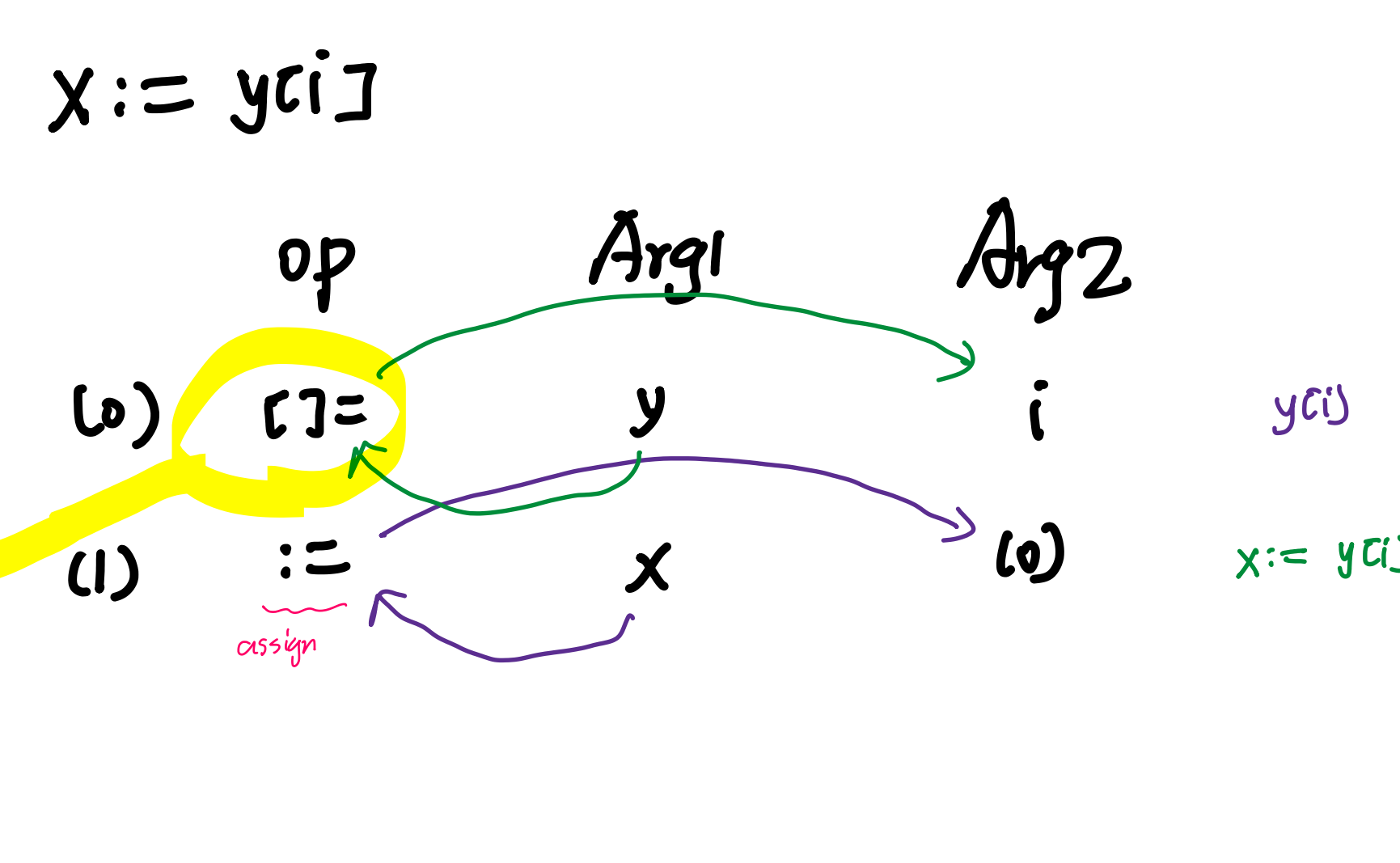
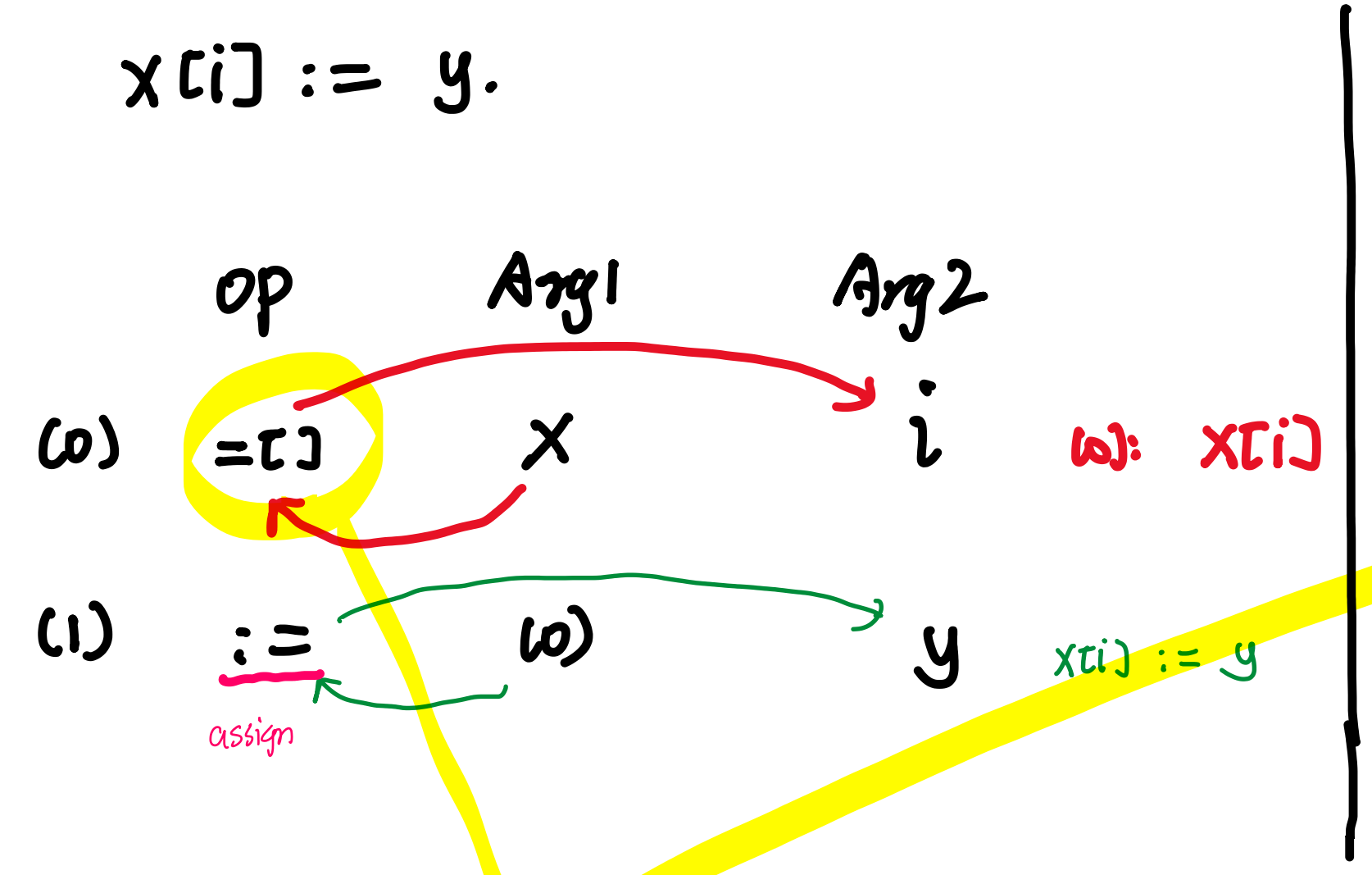
注意: 运算顺序
(有点像 Assembly Language)

三元式

用 pointer 作为 index.

$a := b * -c + b * -c$

op	arg1	arg2
(1)	uminus	c
(2)	$*$	b (1)
(3)	uminus	c
(4)	$*$	b (2)
(5)	$+$	(1) (3)
(6)	assign	a (5)



数组书写时. " $= [i]$ " or " $[i] =$ "
取决于该 array-element 在赋值句中左/右侧。
if $x[i] = y \Rightarrow x[i]$ 在左 \Rightarrow 用 " $= [i]$ "
else $y = x[i] \Rightarrow x[i]$ 在右 \Rightarrow 用 " $[i] =$ "

间接三元式

间接码表.
有利于调序操作.

$x := (A + B) * C$
 $y := D \uparrow (A + B)$

间接码表:

	op	Arg1	Arg2
(1)	(1) $+$	A	B
(2)	(2) $*$	(1)	C
(3)	(3) assign	x	(2)
(4)	(4) \uparrow	D	(1)
(5)	(5) assign	y	(4)