

arr[i] = 4

three address code representation for the given code portion:

$$t_1 = 4$$

$$\text{arr}[i] = t_1$$

- \*  $x = & y$ ;  $y$  এর address যেটি যেটি অর্থের  $x$  এর actual value.
- $x = *y$ ; pointer.
- $*x = y$ ; means the variable which is being pointed by  $x$ , that will be equal to  $y$ .

Three Address Code Representation:

- Quadruples

- Triples

- Indirect Triples.

# '- c' denotes unary minus

Quadruples:

- Quad means ট্রিপল, four.
- Operator  $\rightarrow$  প্রত্যেক operator এর জন্য creating a column.

Write the three address code for the following input string using quadruples representation.

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

$$\Rightarrow t_1 = -c \text{ or } t_1 = \text{uminus } c \text{ or } t_1 = \text{minus } c.$$

Three Address Code:

$$\left\{ \begin{array}{l} t_1 = \text{minus } c. \\ t_2 = b * t_1 \\ t_3 = \text{uminus } c \\ t_4 = b * t_3 \\ t_5 = t_2 + t_4 \\ a = t_5 \end{array} \right.$$

	OP	arg1	arg2	result
0	minus	c		t1
1	*	b	t1	t2
2	minus	c		t3
3	*	b	t3	t4
4	+	t2	t4	t5
5	=	t5		a

OP denotes operator  
 ↗ 0,1,2,3,4,5  
 Quadruples representation number  
 → assign কোরি সেইসব কোরি জুন্নি.

Triples.

- Triples = ৩ টি, three.
- Must use instruction number as there's no result column.  
 ↳ means basically instruction set এর denote পদ্ধতি হচ্ছে.
- Assign operation ও triple representation এর দ্বারা different.

	OP	arg1	arg2
0	minus	c	
1	*	b	(0)
2	minus	c	
3	*	b	(2)
4	+	(1)	(3)
5	=	a	(4)

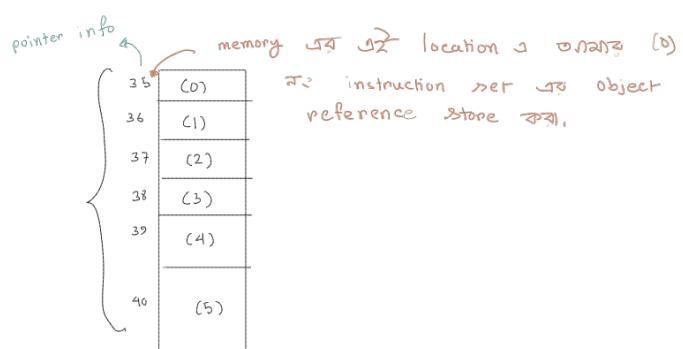
instruction set no denote কোরি  
 ↳ এই instruction set এর অবস্থা হচ্ছে.

রেজিস্টার এর value কোরি assign কোরি অর্থের arg1  
 এর value assign কোরি কোরি কোরি কোরি কোরি  
 instruction set no arg2 এর

Indirect Triples

- basically triples  $\rightarrow$  just কোরি instruction set memory এর  
 কোরি memory address এ point কোরি কোরি কোরি reference  
 কোরি

	OP	arg1	arg2
0	minus	c	
1	*	b	(0)
2	minus	c	
3	*	b	(2)
4	+	(1)	(3)
5	=	a	(4)



Question Type:

input string এর syntax tree মেঘের representation draw কোরি.

## Static Single Assignment

### • Definition:

With the help of Static Single Assignment, we can understand whether a variable's value has already changed or not.

$$\begin{aligned} t &= u \text{ minus } c \\ t &= b + t \\ t &= u \text{ minus } c \\ t &= b + t \\ t &= t + t \\ a &= t \end{aligned}$$

কিন্তু যদি মান না পরিবর্তন করে,

$$\left. \begin{aligned} t_1 &= u \text{ minus } c \\ t_2 &= b * t_1 \\ t_3 &= u \text{ minus } c \\ t_4 &= b * t_3 \\ t_5 &= t_2 + t_4 \\ a &= t_5 \end{aligned} \right\} \begin{array}{l} \text{statistically} \\ \text{singularity} \\ \text{that's good!} \end{array}$$

চলো এই variable র মান কখনোই পরিবর্তন করবেন।

### Syntax Directed Definition for 3 Address Code

$$S \rightarrow id = E ; S \cdot code = E \cdot code \parallel gen (top \cdot get (id \cdot lexeme) '=' E \cdot addr)$$

$$E \rightarrow E_1 + E_2 ; E \cdot code = new \text{ Temp}()$$

$$E \cdot code = E_1 \cdot code \parallel E_2 \cdot code \parallel gen (E \cdot addr = E_1 \cdot addr + E_2 \cdot addr)$$

$$\rightarrow - E_1 ; E \cdot code = new \text{ Temp}()$$

$$E \cdot code = E_1 \cdot code \parallel gen (E \cdot addr = 'minus' E_1 \cdot addr)$$

$$\rightarrow (E_1) ; E \cdot code = E_1 \cdot code$$

$$E \cdot code = E_1 \cdot code$$

$$\rightarrow id ; E \cdot code = '$$

• || ফিল্ড concat করায়।

$$E_1 \cdot addr \parallel E_2 \cdot addr$$

$$\begin{matrix} \downarrow & \downarrow \\ t_1 & t_2 \\ t_1, t_2 \end{matrix}$$

• gen function কিন্তু এটি generate করে।

↳ Three address code generate করে।

• top stack ঘোল id \ lexeme গুরুত্ব করে।

$$E \cdot code = t_1 = t_2 + t_3$$

$$x = t_4$$

$$\bullet new \text{ Temp}() \rightarrow E \cdot code = E_1$$

↳ temp class object এর সঙ্গে এটি একটি-করে temp val change

হয়।