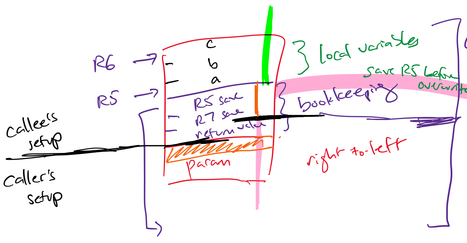
C-to-LC3 for functions

- local variables
- parameters
- bookkeeping
  - RS save (cfp)
  - R7 save (ra)
  - return value

lexicographical alphabetical order

$x = 5 * x / (3 + y) * a;$




```
int multiply(int a, int b);

a = multiply(b, c);
```

a	int	R5	0
b	"	R5	-1
c	"	R5	-2

caller setup

```
ADD R6, R6, #-2   ← # of parameters
LDR R3, R5, #-2   } push c
STR R3, R6, #1    }
LDR R3, R5, #-1   } push b
STR R3, R6, #0    }
JSR MULTIPLY
```

caller teardown

```
LDR R3, R6, #0
STR R3, R5, #0    a ← rv
ADD R6, R6, #3    ← # of params + 1 (rv)
```

callee setup MULTIPLY LV+3

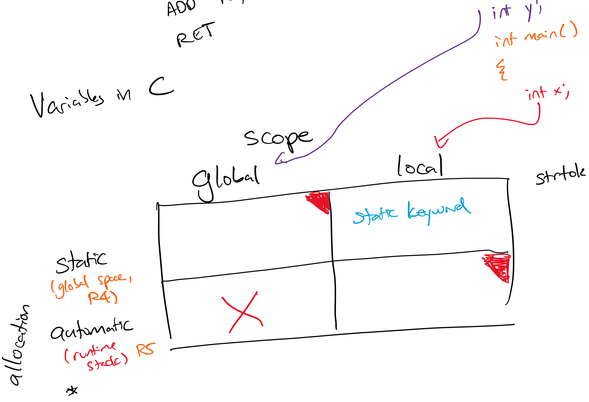
```
ADD R6, R6, #-6
STR R5, R6, #3    LV
ADD R5, R6, #2    LV-1
STR R7, R5, #2
```

[magic happens] result → R0

caller teardown

```
STR R0, R5, #3
LDR R7, R5, #2
LDR R5, R5, #1
ADD R6, R6, #5    LV+2
RET
```

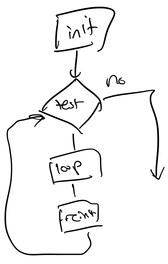
Variables in C

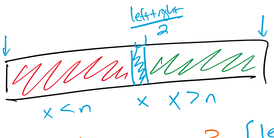



```
i = 2;
printf("%d", i++);   /* 2 */
printf("%d", ++i);   /* 4 */
```

```
for (i=0; i<10; ___)
for (i=10; --i; ___)
```

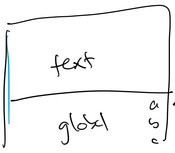
int test reint





1 2 3 [left, right]

arr[mid] = 2 < n = 3



a	R4	#0
b	R4	#1
c	R4	#2