

LC/3 instructions

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ADD ⁺	0001			DR			SR1			0	00		SR2			
ADD ⁺	0001			DR			SR1			1	imm5					
AND ⁺	0101			DR			SR1			0	00		SR2			
AND ⁺	0101			DR			SR1			1	imm5					
BR	0000			n	z	p	PCoffset9									
JMP	1100			000			BaseR			000000						
JSR	0100			1	PCoffset11											
JSRR	0100			0	00		BaseR			000000						
LD ⁺	0010			DR			PCoffset9									
LDI ⁺	1010			DR			PCoffset9									
LDR ⁺	0110			DR			BaseR			offset6						
LEA ⁺	1110			DR			PCoffset9									
NOT ⁺	1001			DR			SR			111111						
RET	1100			000			111			000000						
RTI	1000			000000000000												
ST	0011			SR			PCoffset9									
STI	1011			SR			PCoffset9									
STR	0111			SR			BaseR			offset6						
TRAP	1111			0000			trapvect8									
reserved	1101															

LC/3 Translation of C function

```
int f(int x, int y) {  
    return x + g(y) ;  
}
```

```
        .ORIG      x3000  
;; Activation record for f  
;;  
;; offsets to R5 during call  
;; -1 - Saved R1 (last local)      - R6 points here during call  
;;  0 - Saved R0 (first local)    - R5 points here during call  
;;  1 - dynamic link  
;;  2 - return PC  
;;  3 - return value              - R6 points here at exit  
;;  4 - x (1st arg)                - R6 points here on entry  
;;  5 - y (2nd arg)  
  
;; Create activation record  
f      STR  R7,R6,#-2  
      STR  R5,R6,#-3  
      ADD  R5,R6,#-4  
      ADD  R6,R5,#-1  
;; Save registers  
      STR  R0,R5,#0  
      STR  R1,R5,#-1  
;; Get x  
      LDR  R0,R5,#4                ;; R0 <- x  
;; Call g(y)  
      LDR  R1,R5,#5  
      STR  R1,R6,#-1  
      ADD  R6,R6,#-1  
      JSR  g  
      LDR  R1,R6,#0                ;; R1 <- g(y)  
      ADD  R6,R6,#2  
;; Put x+g(y) on stack  
      ADD  R0,R0,R1  
      STR  R0,R5,#3  
;; Restore registers  
      LDR  R0,R5,#0  
      LDR  R1,R5,#-1  
;; Delete activation record and return  
      LDR  R7,R5,#2  
      ADD  R6,R5,#3  
      LDR  R5,R5,#1  
      RET  
      .END
```