LC-3 Reference -- CS 350: Computer Org & Assembler Lang Pgmg (v. 2016-10-12)
(You can bring this sheet to Quiz 2 & Exam 2)

OPCODES SORTED BY MNEMONIC

Op	Hex	Bin	Arguments
ADD	1	0001	Dst Src1 0 00 Src2
ADD	1	0001	Dst Src1 1 Immed5
AND	5	0101	Dst Src1 0 00 Src2
AND	5	0101	Dst Src1 1 Immed5
BR	0	0000	NZP PCoffset9
err	D	1101	(unused opcode)
JMP	C	1100	000 Base 000000
JSR	4	0100	1 PCoffset11
JSRR	4	0100	000 Base 000000
LD	2	0010	Dst PCoffset9
LDI	A	1010	Dst PCoffset9
LDR	6	0110	Dst Base Offset6
LEA	E	1110	Dst PCoffset9
NOP	0	1110	000 0..0 (BR w/000 mask)
NOT	9	1001	Dst Src1 111111
RET	C	1100	000 111 000000 (JMP R7)
RTI	8	1000	0000 0000 0000
ST	3	0011	Src PCoffset9
STI	B	1011	Src PCoffset9
STR	7	0111	Src Base Offset6
TRAP	F	1111	0000 TrapVec8

OPCODES SORTED BY OPCODE NBR

Hex	Bin	Op	Arguments
0	0000	BR	NZP PCoffset9
0	0000	NOP	000 0..0 (BR w/000 mask)
1	0001	ADD	Dst Src1 0 00 Src2
1	0001	ADD	Dst Src1 1 Immed5
2	0010	LD	Dst PCoffset9
3	0011	ST	Src PCoffset9
4	0100	JSR	1 PCoffset11
4	0100	JSRR	000 Base 000000
5	0101	AND	Dst Src1 0 00 Src2
5	0101	AND	Dst Src1 1 Immed5
6	0110	LDR	Dst Base Offset6
7	0111	STR	Src Base Offset6
8	1000	RTI	0000 0000 0000
9	1001	NOT	Dst Src1 111111
A	1010	LDI	Dst PCoffset9
B	1011	STI	Src PCoffset9
C	1100	JMP	000 Base 00000
C	1100	RET	000 111 000000 (JMP R7)
D	1101	err	(unused opcode)
E	1110	LEA	Dst PCoffset9
F	1111	TRAP	0000 TrapVec8

Trap Vectors (Note: TRAP, JSR, JSRR modify R7)

```
---------------------------------------------------------------------------
x20 - GETC   Read character from keyboard into R0[7..0]; clear R0[15..8].
x21 - OUT    Print character in R0[7..0].
x22 - PUTS   Print string of ASCII chars starting at location pointed to
             by R0 (one char per location; stop at word = x0000).
x23 - IN     Like x20 but prints a prompt on the screen first.
x25 - HALT   Halt execution (turn off running flag)
x24 - PUTSP  (unused) Like x22 but each location contains two characters; the
             one at 7..0 is printed first then the one at 15..8.  Stop at x0000.
```

Condition Code: set by ld, ldi, ldr, lea, add, and, not, various TRAPs

Assembler Directives (below, n can be in decimal or hex)

```
---------------------------------------------------------------------------
.ORIG  n        Load program starting at address n (typically hex constant)
.FILL  n        Allocate 1 word of memory initialized to n
.FILL  label    Allocate 1 word of memory initialized to address of label
.BLKW  n        Allocate n words of memory initialized to 0. (Like n .FILL 0's)
.STRINGZ "str"  Allocate M+1 words for M characters plus terminal null char
                (Some backslash escapes work.)
.END            Last line of assembler program
```

ASCII: Space = 32; Newline = 10; '0'= 48 = x30; 'A'= 65 = x41; 'a'= 97 = x61
Multiples of 16: 32 48 64 80 96 112 128 144 160 176 192 208 224 240 256