Homework Lecture 5 (solutions)

4.1
24 bits

4.3
Lowest: 0
Highest 2^{18} -1
(Note that it is not a byte addressable architecture)

4.5
a) $[-2^{23}, 2^{23} - 1]$ bytes
a) $[-2^{21}, 2^{21} - 1]$ words

4.7
See table 5-1
I added the underscores for readability (distinction of the fields).

a) sethi 0xABCD, %r12

SETHI ➔ op=00
`00----------------------------`

OP2=100
`00-----100----------------------`

RD=12
`0001100100----------------------`

IMM22= 000000 1010 1011 1100 1101

00_01100_100_00000010101011110011101
$1900ABCD_{16}$

b) call label_d
The next address is 64 bytes ahead, this is 64/4=16 words ahead (see page 120)
01_00000000000000000000000010000
40000010_{16}

c) orcc %r15, 255, %r22
10_10110_010010_01111_1_0000011111111
$AC93E0FF_{16}$

d) be label_d
000_0001_010_00000000000000010000
02800010_{16}

e) st %r25, [%r9 + 128] (see page 107)
11_11001_000100_01001_1_0000010000000
$F2226080_{16}$

f) srl %r8, 31, %r9
10_01001_100110_01000_1_0000000011111
$9332201F_{16}$

4.8

exercise_4_8.asm

File Edit

Assemble | Show Asm File | Show Lst File | Show Binary File | Bin->Sim

```
! exercise 4.8
      .begin
      .org 0
      sethi 0xABCDEF, %r12
      call 0xFFFB
      or %r15, 0x1FFF, %r22
      be -4
      st %r25, [%r9+128]
      srl %r8, 31, %r9
      halt
      .end
```

```
ARC Parser (ARCTools Version 2.1.2)
2 errors, see .lst file (above) for details...
.bin file will not be generated...
```

Line Number:1 Column:15

test.lst

File Edit

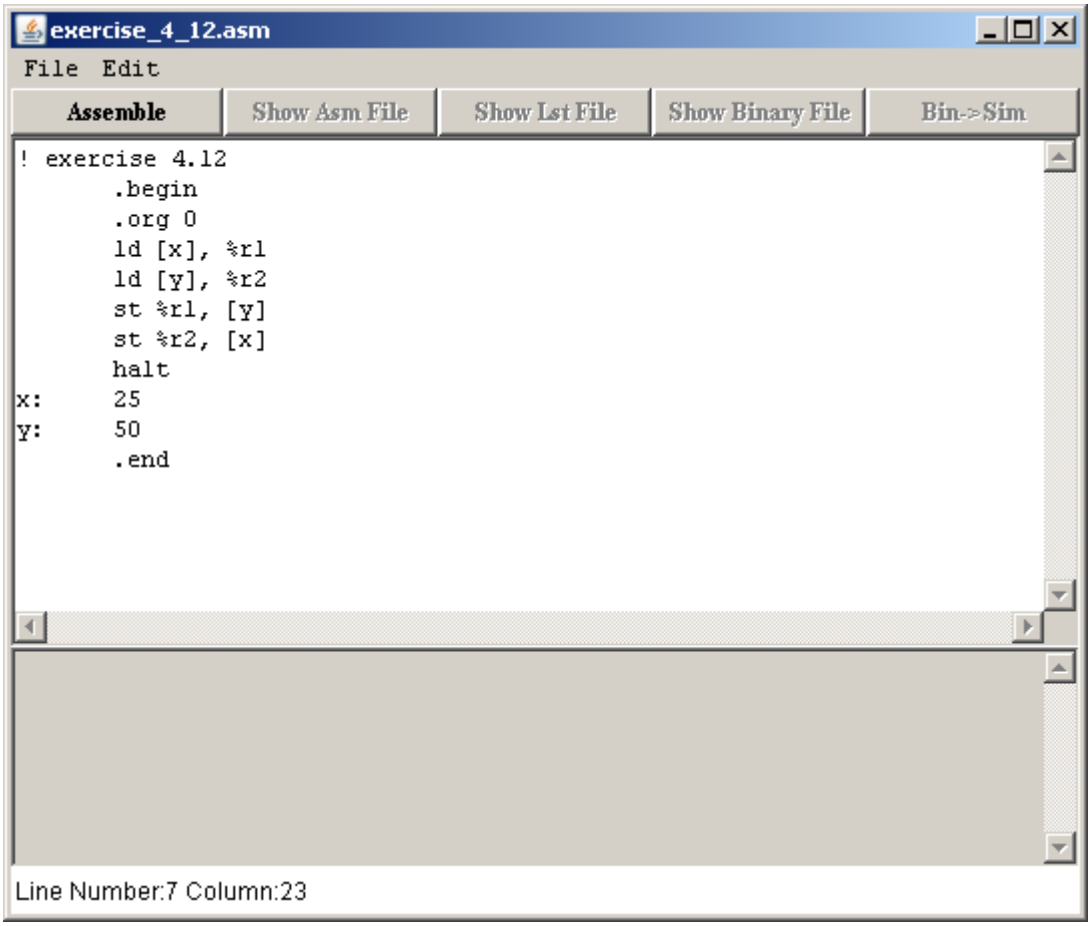
Assemble | Show Asm File | Show Lst File | Show Binary File | Bin->Sim

```
(ARCTools Version 2.1.2)
 HexLoc     DecLoc     MachWord   Label        Instruction
                                               .org 0
00000000  0000000000  192bcdef                 sethi 11259375, %r12
ERROR on line above: Constant is too large for 22 bit representation.
00000004  0000000004  40003ffd                 call 16381
ERROR on line above: Constant  Not aligned on boundry divisible by 4.
00000008  0000000008  ac13ffff                 or %r15, -1, %r22
0000000c  0000000012  02bffffc                 be -4
00000010  0000000016  f2226080                 st %r25, %r9, [128]
00000014  0000000020  9332201f                 srl %r8, 31, %r9
00000018  0000000024  ffffffff                 halt
0000001c  0000000028  0000000f  x:
00000020  0000000032  00000009  y:
00000024  0000000036  00000000  z:
```

```
ARC Parser (ARCTools Version 2.1.2)
2 errors, see .lst file (above) for details...
.bin file will not be generated...
```

Line Number:11 Column:60

4.12



4.13

```
! exercise 4.13
        .begin
        .org 0
Y:      ld      [k], %r1
        addcc   %r1, -4, %r1
        st      %r1, [k]
        bneg    X
        ld      [%r1 + a], %r2
        ld      [%r1 + b], %r3
        addcc   %r2, %r3, %r4
        st      %r4, [%r1 + c]
        ba      Y
X:      halt
k:      40
a:      1,2,3,4,5,6,7,8,9,10
b:      1,2,3,4,5,6,7,8,9,10
c:      .dwb 10
        .end
```

The contents of memory address k is stored in %r1 and is decremented with 4 and stored at memory address k. The decremented value (%r1) is added to the memory locations a, b and c. So the content of de memory addresses a+%r1 and b+%r1 is stored at memory address c+%r1. This process is repeated 10 times. The elements of the arrays a and b are added and the result is stored in an array of 10 elements (in the program space is reserved using **`.dwb 10`**).

4.18

The SPARC is big-endian and the Pentium is little-endian. Therefore the file needs to be “byte swapped” before it can be used by the other architecture.