University of Delaware
CISC260 Homework 4 Solution

Jiefu Li

April 27, 2018

1 2.4.1

Answer:

a.
LDR R0, [R7,#16]
ADD R0, R0,R1
ADD R0, R0,R2

b.
LDR R0, [R7,#16]
LDR R1, [R0,#0]
SUB R1, R2, R1

2 2.5.1

Answer:

a.
temp = Array[3];
Array[3] = Array[2];
Array[2] = Array[1];
Array[1] = Array[0];
Array[0] = temp;

b.
temp = Array[4];
Array[4] = Array[0];
Array[0] = temp;
temp = Array[3];
Array[3] = Array[1];
Array[1] = temp;

3 2.13.1

Answer:
a.
shift r3 left 4 bits: 0x55555550.
Or with r4:

0x57755778

b.
shift r3 left 4 bits: 0xEADFEED0.
Or with r4:
0xFEFFFEDE

4 2.13.2

Answer:

a.
r3 = 0x55555555
r3 became 0xFFFFFFFE shift r4 left 4 bits: 0x23456780.
AND them we get r5:
0x23456780

b.
r3 = 0xBEADFEED
r3 became 0xFFFFFFFE shift r4 left 4 bits: 0xEADFADE0.
AND them we get r5:
0xEADFADE0

5 2.13.3

Answer:

a.
r3 = 0x55555555
shift r3 right 3 bits: 0xAAAAAAA.
r5 = 0xFFEF (0x0000FFEF)
AND them we get r5:
0x0000AAAA.

b.
r3 = 0xBEADFEED
shift r3 right 3 bits: 0x17D5BFDD.
r5 = 0xFFEF (0x0000FFEF)
AND them we get r5:
0x0000BFCD.

6 2.13.4

Answer:
a.
r0 = 0x0000A5A5, left shit 1 bit, we have: 0x00014B4A
OR with r1 = 0x00005A5A, we get r2:
0x00015B5A
b.
r0 = 0x0000A5A5, right shit 1 bit, we have: 0x000052D2
AND with r1 = 0x00005A5A, we get r2:

0x00005252

7 2.13.5

Answer:
a.
r0 = 0xA5A50000, left shit 1 bit, we have: 0x4B4A0000
OR with r1 = 0xA5A50000, we get r2:
0xEFEF0000
b.
r0 = 0xA5A50000, right shit 1 bit, we have: 0x52D28000
AND with r1 = 0xA5A50000, we get r2:
0x00800000

8 2.13.6

Answer:
a.
r0 = 0xA5A5FFFF, left shit 1 bit, we have:0x4B4BFFFE
OR with r1 = 0xA5A5FFFF, we get r2:
0xEFEFFFFF
b.
r0 = 0xA5A5FFFF, right shit 1 bit, we have: 0x52D2FFFF
AND with r1 = 0xA5A5FFFF, we get r2:
0x0080FFFF

9 2.16.1

a.
R2 has value 0.

b.
R2 has value 0.

10 2.16.2

a.
R2 has value 0.

b.
R2 has value 0.

11 2.16.3

```
MOV r2,#0
CMP r0, r1
BLO ELSE
BHI DONE
```

```
BEQ DONE
ELSE: MOV r2, #2
DONE:
```

12 2.16.4

a.
R2 has value 0.

b.
R2 has value 0.

13 2.16.5

a.
R2 has value 0.

b.
R2 has value 0.

14 2.17.4

a.20 b.40

15 2.18.2

a, b, i are in registers r0, r1, r2.
register r3 holds the base address of the array D

a.

```
MAIN:
CMP r2, #10
BEQ DONE
ADD r0, r0, r1
ADD r2, r2, #1
B MAIN

DONE:
```

b.

```
MAIN:
CMP r0, #10
BLT LOOP
B DONE

LOOP:
ADD r4, r0, r1
MUL r5, r0, #4
STR r4, [r3, r5]
```

```
ADD r0, r0, #1
B MAIN
DONE:
```