

```

2.3 sub $t0, $s3, $s4
      add $t0, $s6, $t0
      lw  $t1, 16($t0)
      sw  $t1, 32($s7)

```

**2.4**  $B[g] = A[f] + A[1+f];$

## 2.7

Little-Endian		Big-Endian	
Address	Data	Address	Data
12	ab	12	12
8	cd	8	ef
4	ef	4	cd
0	12	0	ab

## 2.12

**2.12.1** 50000000

**2.12.2** overflow

**2.12.3** B0000000

**2.12.4** no overflow

**2.12.5** D0000000

**2.12.6** overflow

## 2.19

**2.19.1** 0xBABEFEF8

**2.19.2** 0xAAAAAAAA0

**2.19.3** 0x00005545

```
2.20 srl $t0, $t0, 11
      sll $t0, $t0, 26
      ori $t2, $0, 0x03ff
      sll $t2, $t2, 16
      ori $t2, $t2, 0xffff
      and $t1, $t1, $t2
      or  $t1, $t1, $t0
```

```
2.27  addi $t0, $0, 0
        beq  $0,  $0, TEST1
LOOP1:  addi $t1, $0, 0
        beq  $0,  $0, TEST2
LOOP2:  add  $t3, $t0, $t1
        sll  $t2, $t1, 4
        add  $t2, $t2, $s2
        sw   $t3, ($t2)
        addi $t1, $t1, 1
TEST2:  slt  $t2, $t1, $s1
        bne  $t2, $0, LOOP2
        addi $t0, $t0, 1
TEST1:  slt  $t2, $t0, $s0
        bne  $t2, $0, LOOP1
```

**2.39** Generally, all solutions are similar:

```
lui $t1, top_16_bits
ori $t1, $t1, bottom_16_bits
```