Hw #2

ECE 4300   Hyosung Kim

2.1)
```
addi $t0, $s2, -5
add $s0, $s1, $t0
```

2.2)
```
f = i + (g+h);
```

2.3)
```
sub $t0, $s3, $s4
sll $t0, $t0, 2
add $t0, $s6, $t0
lw $t1, 0($t0)
add $t2, $s7, 32
sw $t1, 0($t2)
```

2.4)
```
f = A[f];
B[g] = A[f] + A[f+1];
```

2.5)

```
sll $t0, $s0, 2
add $t0, $s6, $t0
lw  $t1, 0($t0)
lw  $t2, 4($t0)
add $t0, $t1, $t2
sll $t1, $s1, 2
add $t1, $s7, $t1
sw  $t0, 0($t1)
```

2.6.1) n = 5

```
for (I=0  i<n-1  i++)
{
    int  min_Indx
    for  (J = i+1  J<n  J++) {
        if (Array [J] < Array [min_indx])
        {
            min Indx = J;
        }
    }
    Temp = Array (minIndx)
    Array [min Indx] = Array [i]
    Array [i] = Temp;
}
```

2.7) 0x abcdef12

4 Bytes 0xab 0xcd 0xef 0x12

Little 0: 0x12
1: 0xef
2: 0xcd
3: 0xab

Big 0: 0xab
1: 0xcd
2: 0xef
3: 0x12

2.9)
```
sll $t0, $s3, 2

add $t0, $s6, $t0
lw $t0, 0($t0)

sll $t1, $s4, 2
add $t1, $s6, $t1
lw $t1, 0, $t1

add $t2, $t0, $t1

li $t3, 32
add $t3, $s7, $t3
sw $t2, 0$t3
```

2.10) int A[2]
int f

A[1] = A[0]
f = A[0] + A[1]

2.12.1) $s0 = 0x5000,0000

.2) There has been no Overflow

.3) 0x8000,0000 - 0xD000,0000
= 0x8000 0000 + 0x300 0 0000
= 0xB000, 0000

.4) There is overflow

.5) 0x5000, 0000 + 0x 80000000

0x3000, 0000

.6) there is no Overflow

2.18.1) Overall size of R-Type instructions remains 32 bits with adjusted field sizes

18.2) Instruction size is 32 bits but register fields increase to 7 bits

18.3) – it could increase the size by adding registers and having wider instructions
– It could decrease due to simpler instructions and more specialized instructions

2.21)
```
li $t0, -1
xor $t1, $t2, $t0
```

2.22)
```
lw $t3, 0($s1)
sll $t1, $t3, 4
```

2.26) Final value of $s2 = 20

26.2)
```
int i = N;
int temp = 0;
while (i >0) {
        temp += 2;
        i--;
}
*B = temp;
```

26.3) 5N +1

2.29)
```
while (i <100) {
      *result += MemArray[i];
      i++;
}
```

2.38) t1 0x1000,0000

t2 0x1000,0010

0x1000,0000 = 0x11223344

stores highest byte $\boxed{0x11}$

2.47) 70 % arith

20 % branch

10 load/store

47.1) 2 × 0.7 1.4
3 × 0.2 0.6
6 × 0.1 0.6

$\boxed{2.6}$

47.2) ~~2.6 · 1.25 = 3.25 − 1.2~~
~~0.75~~ ~~$\frac{2.05}{2}$ = 1.025~~
~~$\boxed{1.025}$~~

47.3) ~~2.6 · 1.5 =~~
2.6 · 0.5 = 1.3
0.1
$\boxed{0.05 \text{ Cycles}}$

47.2) 2.6 · 0.75 = 1.95
−1.2
0.75
$\boxed{0.375}$