Name: _____

CIS 351 Sample AL2 Problem

31 October 2025

AL2: Assembly Branches

(a) Convert the following line of Java code to assembly: `t0 = t1 + t2 + t3 - t4 + t5`

(b) Convert the following line of Java code to assembly: `t0 = (t1 ^ t2) & (t3 | !t4)`

(c) Convert the following Java code to assembly. Your answer *must* use `slt` and either `beq` or `bne`. Do not use any pseudoinstructions. Note: This is not a function; it is simply a section of code. Set your code up as if there are more instructions following the block of code (i.e., don't use `jr $ra`).

```
if (t1 - 6 < t2) {
  t0 = t1;
} else {
   t0 = t2 + 4;
}
t1 = t1 + 7
```

Click here for solutions: `al2_sample_solutions.pdf`

CIS 351 Sample AL2 Problem Solutions

Fri 31st Oct, 2025

AL2: Assembly Branches

(a) Convert the following line of Java code to assembly: `t0 = t1 + t2 + t3 - t4 + t5`

There are many correct answers. Here is one:

```
add $t0, $t1, $t2
add $t0, $t0, $t3
sub $t0, $t0, $t4
add $t0, $t0, $t5
```

(b) Convert the following line of Java code to assembly: `t0 = (t1 ^ t2) & (t3 | !t4)`

There are many correct answers. Here is one:

```
xor $t0, $t1, $t2   # computes t1 ^ t2
xori $t5, $t4, -1    # computes !t4
or $t5, $t5, $t3    # computes t3 | !t4
and $t0, $t0, $t5
```

(c) Convert the following Java code to assembly. Your answer *must* use `slt` and either `beq` or `bne`. Do not use any pseudoinstructions. Note: This is not a function; it is simply a section of code. Set your code up as if there are more instructions following the block of code (i.e., don't use `jr $ra`).

```
if (t1 - 6 < t2) {
  t0 = t1;
} else {
   t0 = t2 + 4;
}
t1 = t1 + 7
```

There are many correct answers. Here is one:

```
addi $t3, $t1, -6
slt $t4, $t3, $t2
beq $t4, $zero, else
addi $t0, $t1, 0  # does a move
j done
else: addi $t0, $t2, 4
done: addi $t1, $t1, 7
```