

Assigned: Thursday 8 October

Due: Friday 16 October 10:30 pm

Collaboration: This assignment must be completed individually.

Presentation: Number problems in order and present code in a fixed-width font including comments that make each instruction's purpose clear.

Submission: Upload a PDF with your name, box number, and all solutions to PioneerWeb.

1. Show the machine code (in binary and hex) for the following MIPS code segment.

```
lw $s0, 2($s7)
andi $t0, $s0, 0x8
srl $t0, $t0, 3
```

For the remaining problems, translate the C code segments into MIPS assembly.

2. Assume that ints a, b, c, and d are located in registers \$s0, \$s1, \$s2, and \$s3, respectively.

```
if (a == b)
    d = c;
else if (a == c)
    d = b;
else
    d = a;
```

3. Assume that ints n, i, and sum are located in registers \$s0, \$s1, and \$s2, respectively.

```
sum = 0;
for (i=1; i < n; i++)
    sum += i;
```

4. Assume that the address of array int A[] is located in register \$s7 and that ints n, i, and sum are located in registers \$s0, \$s1 and \$s2.

```
sum = 0;
for (i=0; i < n; i++)
    sum += A[i];
```

5. Translate the following C function into MIPS assembly.

```
int arraysum(int array[], int size) {
    int i, sum;
    sum = 0;
    for (i=0; i < size; i++)
        sum += array[i];
    return sum;
}
```

6. Translate the following C function call into assembly. Assume \$s5 contains the address of the array int numbers[], \$s6 contains the size int n, and the int value total will be stored in \$s7.

```
total = arraysum(numbers, n);
```

7. Translate the following recursive C function into MIPS assembly.

```
int termial(int n) {
    if (n < 1)
        return 0;
    else
        return (n + termial(n - 1));
}
```

Acknowledgments This derivative work of Janet Davis and Marge Coahran, used under the Creative Commons [Attribution-Noncommercial-Share Alike 3.0 United States License](https://creativecommons.org/licenses/by-nc-sa/3.0/) was developed in collaboration with Charlie Curtsinger.