Lab Activity 6d
Intro to Computer Science (420-121-VA)
Perry R. James, 1 October 2020

MIPS – Part 4

Fourth Program

Consider the following algorithm for adding the numbers from 1 to `n`:

```
set $sum = 0$
set $i = 1$
while $i \le n$
    add $i$ to $sum$
    increment $i$
output $sum$
```

This can be encoded in MIPS assembler as follows:

```
     .data
n:   .word 4
sum: .word 0

      .text
        #calculate sum = 1 + 2 + … + n
main:   move $t2, $zero         # set $t2 to 0
        move $t0, $zero         # set i ($t0) to 1
        addi $t0, $t0, 1
        lw   $t1, n             # set $t1 to n
loop:   slt  $t3, $t1, $t0      # $t3 = n < i ? 1 : 0
        bne  $t3, $zero, finish # if $t3 ≠ 0 goto finish
        add  $t2, $t2, $t0      # add i to $t2
        add  $t0, $t0, 1        # add 1 to $t0
        j    loop               # goto top of loop
finish: sw   $t2, sum           # store $t2 in sum
```

Note that `$t0` is used to store the value of `i`, `$t1` the value of `n`, and `$t2` the partial value of `sum`.

Answer the following questions:

1. What value is calculated for `sum` (in decimal) when `n` = 4? _____
   When `n` = 100? 1000? _____ _____
2. Modify the program to calculate the factorial of *n* (*n*!)
3. What does your program calculate for 7! ? _____
4. What does your program calculate for 13! ? _____
   Is this value correct? Why or why not?
5. What is the largest value of *n* for which *n*! is calculated correctly? _____
6. Modify your program so that it works correctly for larger values of *n*.
7. What is the largest value of *n* for which *n*! is calculated correctly by your new program? _____