**Homework 4**

Trevor Lund, CprE 381

1. Simulating MIPS Code
   1. Input: Output
      1. 1: 1
      2. 3: 6
      3. 4: 10
   2. The program takes input greater than or equal to 1, and returns (n being input) n + (n-1) + (n-2) + … + 1
   3. Included in the attachments
   4. Included in the attachments
2. First MIPS Applications
   1. Included in the attachments
   2. Included in the attachments. To optimize the program, check to see which number is smaller and use that one as the counter ($s0 in this case)
3. Application Benchmarking
   1. This code finds the summation of the common elements the second array has in common with each element of the first array. $v0 = A2 elements in common with A1[0] + A2 elements in common with A1[1] + …
   2. This code will take 4 + 3000(5) + 3000\*3000\*12 + 3000\*4 = 108,027,004 cycles. With each cycle taking 1/(2.4) = .4167 seconds, the code will take 45,011,251.67 seconds at its worst case.
4. Computer Arithmetic
   1. 1 Clock cycle per step, so hardware will take 3 \* 32bits \* 7tu = 672tu and software will take 4 \* 32bits \* 7tu = 896tu.