## CSC 103 • How Computers Work Homework 2: PIPPIN Due 5:00 PM on Friday, 24 September 2010 Please submit your work electronically via Moodle (Paper submissions due 1:00 PM)

## Part One: Analyzing Simple Programs

For the PIPPIN code segments below, determine an equivalent algebraic expression or statement that would perform the same task. The first has been done for you as an example. The last is included as a challenge; if you have trouble figuring out what is going on, try running a few simulations with small values in X and Y, to see what the result is. For example, try X = 7 and Y = 2, X = 3 and Y = 3, X=10 and Y = 4. Record the results of these simulations, even if you cannot figure out a formula for what is being computed.

| A. (              | 0 | LOD X   | E. ( | )  | LOD #5 |
|-------------------|---|---------|------|----|--------|
| /                 | 2 | SUB #2  | 2    | 2  | MUL X  |
| 2                 | 4 | STO Y   | 4    | 4  | STO T1 |
|                   |   |         | (    | 5  | LOD Y  |
| Answer: $Y = X-2$ |   |         | 8    | 8  | SUB Z  |
|                   |   |         |      | 10 | DIV T1 |
| В. (              | 0 | LOD X   |      | 12 | STO W  |
| /                 | 2 | DIV #3  |      |    |        |
| 2                 | 4 | STO Y   | F. 0 | )  | LOD #1 |
|                   |   |         | 2    | 2  | MUL X  |
| C. (              | 0 | LOD #64 | 4    | 4  | STO Z  |
| 4                 | 2 | DIV Y   | (    | 5  | LOD Y  |
| 2                 | 4 | STO W   | 5    | 8  | SUB #1 |
|                   |   |         |      | 10 | JMZ 18 |
| D. (              | 0 | LOD Y   |      | 12 | STO Y  |
| 4                 | 2 | SUB X   |      | 14 | LOD Z  |
| 2                 | 4 | ADD #5  |      | 16 | JMP 2  |
| (                 | 6 | STO Z   |      | 18 | HLT    |
|                   |   |         |      |    |        |

For this project, you will program <u>PIPPIN</u> to compute some simple numeric results. Write assembly code to perform the computations described below, and test your program on the simulator. Submit a screenshot of your program before and after it has completed running on some input.

[Recall that we work in assembly language, but the computer actually sees binary machine language opcodes. You can look at your program in machine language by clicking the 'binary' button at the bottom of the simulator. You can also get more information on the instruction codes for PIPPIN by looking at the <u>PIPPIN User's Guide</u> on the course web page.]

A. W = ((X - 2) \* X) +8)
B. Z = (W-2\*(W/2))
C. if Z equals 0 Y = 2 else Y = 1

D. Can you describe in words the simple concept that B & C together are trying to compute in register Y? (If the concept doesn't seem simple, then you're probably missing it. Again, try using the simulator to test out the result for different values of W – 1, 2, 3, 4, 5, etc. In what cases is W-2\*(W/2) ever nonzero? The point here is that a concept that seems simple may actually involve significant computation to implement.)