1. Briefly define, in the context of this course, each of the following terms.

   a. problem specification - the careful statement of what problem is to be solved, including domain, range, preconditions, postconditions, and interface [LOC, p. 5]
   b. CASE - computer assisted software engineering [LOC, p. 24]
   c. recursion - a function that calls (a smaller version of) itself [AE, p. 184]
   d. HTML form - a named object consisting of other HTML objects (e.g., text input fields, buttons, textareas) used for interaction [AE, p. 131]
   e. assembly language - a language of mnemonics where each instruction corresponds to a machine (i.e., binary) instruction [AE, p. 210]

2. List two factors that complicate the writing of specifications. [LOC, p. 16]

   * poorly defined goal
   * ambiguous language (e.g., English prose)
   * assumptions made about the experiences of team members
   * distinct problem views
   * incomplete specifications

3. Arrange these terms hierarchically (i.e., as a "contains" relationship):

   * statements
   * functions
   * programs
   * tokens

   A program contains functions, each function contains statements, each statement contains tokens.

4. Convert the following binary numbers to their decimal equivalents:

   * 11000 \(2^4 + 2^3 = 16 + 8 = 24\)
   * 110010 \(2^5 + 2^4 + 2^1 = 32 + 16 + 2 = 50\)
   * 10011 \(2^4 + 2^1 + 2^0 = 16 + 2 + 1 = 19\)
   * 10211 undefined, since 2 is not a valid binary digit
5. Given that \( x \) and \( y \) are numeric parameters, describe what this function returns:

```javascript
function mystery(x, y) { // demonstration: x = 5, y = 3
    x = x - y; // x = 2, y = 3
    y = x + y; // x = 2, y = 5
    x = y - x; // x = 3, y = 5
    return x; // returns 3, the original y value
}
```

The above function should return the original \( y \) value; actually, the function swaps the original \( x \) and \( y \) values.

6. What is the value of result in the following program code?

```javascript
function F(n) { // returns n^3
    var temp = 0;
    var index = 0;
    while (index < n) {
        temp = temp + (n * n);
        index = index + 1;
    }
    return temp;
}

function test() {
    var result = F(3);
}
```

result = 27

7. The discriminant of a quadratic equation

\[ a \ x^2 + b \ x + c = 0 \]

is given by the equation

\[ \text{discriminant} = b^2 - 4 \ a \ c \]

Show the parse tree for the discriminant equation.
Let: \( w \) discriminant, \( x \): a, \( y \): b, and \( z \): c, so

\[
\begin{align*}
\quad \quad w &= b\times b - 4\times a\times c \quad \quad | \quad w &= y\times y - 4\times x\times z \\
\end{align*}
\]

8. Refer to #7 above, now generate the PIPPIN assembly code for the parse tree to determine the discriminant of a quadratic equation.

```assembly
LOD Y
MUL Y
STO T1
LOD #4
MUL X
MUL Z
STO T2 or possibly SUB T1
LOD T1    MUL #1
SUB T2    STO W
STO W
```

9. Find the logic error(s) in the following program fragment.

```javascript
function pounds(x, y)
{
    y = 2.2 * x;
    return y;
}

function weight()
{
    var start = 10;
    return pounds(start);
}
```

A function defined with two parameters cannot be called with only one parameter.
10. Implement the following Javascript loop in PIPPIN assembly language.

```javascript
{  
    var w = 10, x = 0;
    while (x < w)
       x++;
}
```

```assembly
LOD #10 ; L and K are statement labels
STO W
LOD #0
STO X
L: LOD W
   SUB X
   JMZ K
   LOD X
   ADD #1
   STO X
   JMP L
K: HLT
```