1. Using the assembly language for the hypothetical machine used in lab, starting over for each part, give a statement or series of statements that would
   i) Put the value currently stored in R into the memory location with label Y.

   ii) Double the value that is currently into the memory location with label X.

   iii) Possibly change the value in the CCR.

   iv) Put the value 1 into the memory location with label X.

2. Circle the statement types whose execution phase could possibly change the value in R:
   ADD, JUMPGT, COMPARE, STORE, LOAD, DECREMENT, IN,
   OUT, CLEAR

Assembly Language Table for Exam 3

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>0000</td>
<td>1000</td>
<td>LOAD X</td>
<td>JUMP X</td>
</tr>
<tr>
<td>0001</td>
<td>1001</td>
<td>STORE X</td>
<td>JUMPGT X</td>
</tr>
<tr>
<td>0010</td>
<td>1010</td>
<td>CLEAR X</td>
<td>JUMPEQ</td>
</tr>
<tr>
<td>0011</td>
<td>1011</td>
<td>ADD X</td>
<td>JUMPLT</td>
</tr>
<tr>
<td>0100</td>
<td>1100</td>
<td>INCREMENT X</td>
<td>JUMPNEQ X</td>
</tr>
<tr>
<td>0101</td>
<td>1101</td>
<td>DECREMENT X</td>
<td>IN X</td>
</tr>
<tr>
<td>0110</td>
<td>1110</td>
<td>COMPARE X</td>
<td>OUT X</td>
</tr>
<tr>
<td>0111</td>
<td>1111</td>
<td></td>
<td>HALT</td>
</tr>
</tbody>
</table>

3. Using the assembly language for the hypothetical machine (above):
   a. Write a complete assembly language program, including pseudo-operations, for a program that allows the user to enter 2 numbers, say X, and Y, and then outputs the value of \( 3X - 2Y + 2 \)
b. For this problem, you are to use the techniques covered in class to convert the following pseudocode program to an assembly language program. The program allows the user to enter a list of numbers (terminated by 0) and outputs the largest of the list.

**Pseudocode**

Get Num  
Set Large to Num  
While Num ≠ 0 Do  
    If Num > Large Then  
        Set Large to Num  
    Get Num  
End-Of-Loop  
Print Large  
Stop

**Assembly Language**
4. Assume that the first few memory cells of our hypothetical lab machine have the following contents:

<table>
<thead>
<tr>
<th>Address (decimal)</th>
<th>Contents (binary)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0000000000000111</td>
</tr>
<tr>
<td>1</td>
<td>1000000000000101</td>
</tr>
<tr>
<td>2</td>
<td>0011000000000110</td>
</tr>
<tr>
<td>3</td>
<td>000100000001001</td>
</tr>
<tr>
<td>4</td>
<td>010000000001000</td>
</tr>
<tr>
<td>5</td>
<td>1111000000000000</td>
</tr>
<tr>
<td>6</td>
<td>000000000001011</td>
</tr>
<tr>
<td>7</td>
<td>000000000001010</td>
</tr>
<tr>
<td>8</td>
<td>000000000001101</td>
</tr>
</tbody>
</table>

Assume further that the machine is ready to begin the fetch-decode-execute cycle, that the PC holds 000000000000, and that the register R holds 0 (decimal). You may use binary, decimal or instruction formats in your answers (as you please).

a. Tell exactly what will be stored in each of the following locations at the end of the fetch phase (a single fetch phase beginning with the situation described).

PC:
IR:
R:
Memory location 7:

b. Tell exactly what will be in these locations at the end of the execute phase (the single execute cycle following the fetch phase of part a).

PC:
IR:
R:
Memory location 7:

c. Tell exactly what will be stored in each of the following locations at the end of the second fetch phase (a single fetch phase following part b).

PC:
IR:
R:
Memory location 7:
d. Tell exactly what will be in these locations at the end of the next execute phase

PC:
IR:
R:
Memory location 7:

5. Assume that the alphabet consists of the lowercase letters only. Let the key S be 4 for a Caesar cipher. Write the cipher text for each of the following plain texts:

   a. python
   b. hacker

6. State two advantages and one disadvantage of a network architecture that routes every connection between two nodes through a single, central master node.

7. State two advantages and one disadvantage of a point-to-point network architecture.