1. Abstract data types (ADTs) should be well-defined. How can the ADT called the “stack” be well-defined when it doesn’t specify whether the stack should be an array, linked-list, ordered tree, etc.?

2. Demonstrate the 2-stacks algorithm on a boolean expression like
   \[ 1 \& ( 0 \mid 0 ) \mid 1 \]
   Same for arithmetic expressions.

3. ```c
   int I = 1;
   int J = 2;
   int *P = &I; /*This declares P to be of type pointer-to-int.*/
   printf("Enter an int:");
   scanf("%d", P);
   printf("%d %d %d\n", I, J, P);
```
   What does this program print when the user interaction looks like:

   Enter an int: 57
   ??? ??? ???

4. Post offices have machines that print postage labels to paste on packages after the customer pays for the postage. Choose the best completions for explaining object oriented programming:
   - The machine is like a(n) ______________________ (class, object, computer).
   - The label is like a(n) ______________________ (class, object, computer).

5. **Computer Architecture I**: Draw a sketch of the structure of a typical computer. Explain the function of a computer as emphasized in this course.

6. List as many layers from the layered view of computer architecture as you can (more points for more layers in the correct order.)

7. An arithmetic expression expresses a number. (True?, False?, Debatable?) Explain your answer.

8. What do we call the activity of writing, testing and debugging code to implement an algorithm? What is the result? (The answer must be more specific than “implementation.”)

9. What was the “dictionary” or “symbol table” used for in Project 1? Ditto for an assembler.

    - There is one function and many ways to call it.
    - There are many functions with the same name.
11. Where must C++ or Java code be in order for that code to access the private data or function members of a class?

12. What is an include guard and how do you use it in C/C++?

13. The following diagram shows one PC/Windows host in the Science Library G0012 and one Sun/Solaris Unix host in the ITSUNIX server room.

Suppose you have started the X Server on the Windows machine, started ssh to log into the ITSUNIX server with a tunnelled X connection. Also, you are running Emacs on the ITSUNIX server with the Emacs window on your Windows desktop. You compiled and linked a C program, and got it to run.

The diagram shows that the Emacs window and the Windows OS are running in the PC. Draw the boxes which represent the other processes inside the hosts in which they are running. (A process is “the running of a program.”)

14. Given numbers in binary with a fixed word size (say 8 bits) demonstrate: addition (with carries), formation of the 2-s complement, adding a number to its 2-s complement to produce 0, subtracting two numbers by forming the 2-s complement of one of them and adding, etc.

15. Demonstrate left and right bit shifting and bit rotating. Making use of the LC-3 reference manual, how can you compose operations from the LC-3 Instruction Set Architecture (ISA) to implement shift and rotate which are not in this ISA. Also, distinguish between arithmetic right shift and logical right shift. How do you write C/C++ code to guarantee that a right shift is logical (does not extend the sign bit)?

16. What type of data must be the second and later arguments to C’s `scanf()` function to input data? What symbol do you use to express this about a variable `I` so you can read data into `I`?

17. The first argument to `scanf()` is called the ____________________________.
18. What does \(*P = 32; \) do in C/C++?

19. What is a bit?

20. What is a representation? List and explain a few from this course so far.

21. Why do some standardizers prefer to call 1024 bytes a “kibibyte” (for kilobinary) and not kilobyte, even though it sounds like a kind of cat food? Would you rather have a kilobuck or a kibibuck? (“Buck” is US slang for dollar.)

22. Demonstrate the algorithm to convert a decimal number to binary by successive subtraction of powers of 2.

23. Demonstrate the results of the C/C++ bitwise operators & , | , ~ , << and >> on integers given in binary, hexadecimal or binary.

24. Given the C/C++ program fragment

   \[
   \begin{align*}
   \text{unsigned int } & I; \\
   \text{signed int } & J; \\
   \text{int } & K; \\
   & I = 0xFFFFFFFFA; \\
   & J = 0xFFFFFFFFA; \\
   & K = 0xFFFFFFFFA;
   \end{align*}
   \]

   What is the value, in 32 bit Hex, of

   \[
   (I >> 4) \\
   (J >> 4) \\
   (K >> 4)
   \]

25. Demonstrate how to code access to individual bits by (1) bitwise AND with a mask and (2) by shifts.

26. Given 2 half-adder subcircuits and one OR-gate,

   \[
   \begin{array}{c}
   \text{a} \quad \text{H1} \\
   \text{b} \\
   \text{sum} \\
   \text{carry} \\
   \end{array} 
   \]

   \[
   \begin{array}{c}
   \text{a} \quad \text{H2} \\
   \text{b} \\
   \text{sum} \\
   \text{carry} \\
   \end{array}
   \]

   construct a full-adder circuit from them.

27. Express the half-adder’s sum and carry output functions of \(a\) and \(b\) using AND, OR and NOT operations.
28. Write the truth tables to determine whether or not two Boolean expressions with variables are equivalent.

29. In von Neumann’s fundamental model of a computer, is the program in the control unit or the memory? The data is in the memory. So how can the program be in the memory also?

30. Explain one of DeMorgan’s rules (your choice) in two ways: (1) Analyze a personal situation such as being rich and/or happy. (2) Calculate two truth tables.

31. Why and how do we apply the idea of 2-s complement to program the LC-3 to subtract, or to test if two 16-bit values are equal?

32. Given some short and possibly nonsensical sequences of LC-3 instructions, either in binary or in assembly language, demonstrate what they do to the registers, memory and condition flags when they are run.

33. Sketch the top level pseudo-code for the Project 4 (pattern match count) assignment.

34. Some modern (and currently) expensive architectures have 64-bit ALUs. How many stages must their barrel shifters have?

35. Code simple loops, calculations or decisions in LC-3 assembly language and translate it into binary.

36. What must be done with R7 in order for an LC-3 function to call another LC-3 function? How do you program LC-3 functions to return? Which register does the LC-3 RET instruction use?

37. What is a combinational logic circuit? What additional capability can a logic circuit with a cycle have, which no combinational circuit can have? Why is this crucial in a computer?

38. Explain and use all the addressing modes in the LC-3 ISA.

39. Given one LC-3 instruction, sketch how the LC-3 datapath is used step-by-step to carry it out.

40. Explain and code loops that are
   - counter controlled and
   - sentinel controlled.

   Which of these loops is used to process all the characters of a C-string?

41. Demonstrate what the assembler does with an assembly language program.

42. Explain, code, demonstrate, etc. the callee-save protocol that our LC-3 functions will use so they can call each other without losing the return address or other values in registers.