1) Why should you be interested in learning about Lambda Calculus?
    a. Can encode any computation.
    b. Basis for functional programming.
    c. Now in most languages.
2) How do you encode the concepts of TRUE, FALSE, NOT, AND, OR?
    a. TRUE: λx.λy.x
    b. FALSE: λx.λy.y
    c. NOT: λb.b FALSE TRUE
    d. AND: λx.λy.x y x
    e. OR: λx.λy.x x y

3) What is important about the Lambda Calculus expression called 'Y Combinator'?
    a. It implements recursion in Lambda Calculus.
4) Write the Y Combinator expression in Lambda Calculus.
    a. λf.(λx.(λ(x x))λx.(f(x x)))

5) Where did the language 'Haskell' get its name?
    a. It is named after Haskell B. Curry.
6) In the video it was mentioned that Erlang was used to code what?
    a. WhatsApp.

7) How is 'pattern matching' used?
    a. To determine what definition of a function is used based on the structure of the input.

8) Complete this sentence: "NP problems are hard to solve but easy to _____"
    a. Check.
9) What is the example of an NP problem used in the video?
    a. Factoring numbers.
10) What are the TV shows mentioned in the video?
    a. The Simpsons.
    b. Futurama.

11) Floating point numbers are essentially what?
    a. Scientific notation.
12) Computers perform scientific notation in what base?
    a. 2
13) What is the problem with adding 1/3 + 1/3 + 1/3 using base 10 and ignoring recurring numbers?
    a. You run out of digits. .333 + .333 + .333 = .999 not 1.0.
14) What is 1/10 in base 2?
    a. 0.00111101110011001100110011

15) What is the name of the function discussed in the video?
    a. Ackermann's function.
16) Can Ackermann's function be coded using for or 'DO' loops?
    a. No.
17) What is the value of Ackermann(4,1)?
    a. 65,533
18) How many minutes will the machine in the video take to calculate Ackermann(4,2)?
    a. 3*2^65,533
19) The performance characteristic of Ackermann's function is described as what?
    a. Super Exponential.

20) A loop nested in another loop has the performance characteristic of what?
    a. Polynomial.
21) What was the limitation of Fortran mentioned in the video?
    a. Doesn't allow user level recursion.
22) What real-world use needs complex recursion?
    a. Compilers.

23) There was a need to have a language that could cope with what?
    a. Different widths of objects.
24) C is most powerful when considered as the classical what?
    a. System implementation language.

25) What are the names of the two fields of the ‘THING’ structure?

a. Item
b. next

26) What is the advantage of the ‘Triple Ref Technique’?

a. You are able to remember the previous node in a singularly linked list.

27) What is the procedure used in the video to compare the different structures?

a. Sum all the values of p and compare the times to complete.

28) Why is the reverse array faster on the Atari?

a. There is an instruction for decrement, then branch if not 0 on the 68000 CPU. This reduces multiple instructions into one.

29) What would be the goal of requiring people to be exposed to coding?

a. To demystify computer systems and allow those who wouldn’t have learned it to know what learning options are available.

30) List 3 or more of the different sort algorithms mentioned in the video.

a. Bubble sort.
b. Quick sort.
c. Heap sort.
d. Selection sort.
e. Cocktail sort.

31) What is the ‘Decision Problem’?

a. Is there an automatic way to determine whether certain premises entail a conclusion.

32) An example of an abstraction used in the video is, “A transistor is a type of ___”?

a. Switch.

33) Which video was the most interesting or your favorite?

a. I liked the video that went into detail about how concepts such as TRUE and FALSE are encoded in Lambda Calculus. Before I knew Lambda Calculus was the basis for functional languages but didn't know how to mathematically write or compute functions.