1. Why do we study programming languages? (Give three reasons)

2. Why in tarnation would we have programming languages when that dang ol computer understands zeroes and ones?

3. Define programming language:

4. What is the fetch-execute cycle?

5. What was the first high level language?

6. Who invented Fortran?

7. When was Fortran invented?

8. What is the only high level structure in Fortran?

9. What was the purpose of the do-loop with regards to efficiency?

10. Define a compiler

11. Define an Interpreter:

12. How does a compiler differ from an interpreter?

13. What did the loader do in the first assignment (psuedo-code interpreter)?

14. Explain how psuedo-code interpreters are related to virtual computers.

15. How many IO statements did Fortran have?

16. How was floating point calculation handled in the earliest computer systems?

17. Name each advance related to the IBM 704 as discussed in class.

18. How were Fortran subprograms compiled with respect to the entire program?

19. Explain how Fortran implemented recursion.

20. What is static allocation?

21. Why are GOTO statements considered harmful?

22. Define the Static Structure Principle

23. Define the Zero, One, Infinity Principle

24. Derive the Fortran 1-D addressing equation

25. Derive a 2-D addressing equation from the previously derived equation.

26. What is the origin of Fortran control structures?

27. FORTRAN Syntax/Semantics

   (a) Do loop

   (b) Arithmitic IF

   (c) GOTO statement

   (d) Common block

   (e) EQUIVALENCE statement

28. ALGOL Statistics

   (a) Year?

   (b) Who invented it?

   (c) Name three objectives/goals for the newly created ALGOL.

   (d) Explain the main difference between ALGOL-60 and ALGOL-58.

   (e) Practically nobody wrote programs for ALGOL and its definitly not used today. Why is it such a big deal?

   (f) Why did nobody write programs for ALGOL?

29. BNF

    (a) What does BNF stand for? (hint: its not Best New Friend)

    (b) Who invented BNF

    (c) Where was it first used?

    (d) What is the purpose of BNF?

30. Define variable scope within the context of programming languages

31. Explain the dangling-else problem.

32. How long does each of the below scopes bind a variable to an address?
    (a) Global

    (b) Function

    (c) Block

33. What is the name of the area in memory that stores variables?

34. Isn't a compound statement just a fancy word for a block? Why/Why not? What is a block anyway?

35. Define feature interaction in the context of programming languages.

36. Name one feature interaction in FORTRAN.

37. What is dynamic scoping?

38. Define Grammar in the context of programming languages.

39. How do we define a Grammar for a programming language?

40. What is the syntax of the rewrite rule?

41. Produce a parse tree of the below grammar using a top down technique:

    $\langle statement \rangle \rightarrow \langle expr \rangle$