6. In what language is most of UNIX written?

It is written in C

7. What is the disadvantage of having too many features in a language?

It becomes too complex and can make learning language harder. Also reduces readability

8. How can user-defined operator overloading harm the readability of a program?

If it is unexpected or inconsistent with the standard meaning. For instance, using * is used for anything except multiplication

9. What is one example of a lack of orthogonality in the design of C?

Arrays cannot be directly assigned to each other but pointers can.

10.What language used orthogonality as a primary design criterion?

ALGOL 68

11.What primitive control statement is used to build more complicated control statements in languages that lack them?

Goto statement is often used

12.What does it mean for a program to be reliable?

If the program performs as intended, handle errors and avoids unexpected behaviors

13.Why is type checking the parameters of a subprogram important?

It helps to prevents errors which improves reliability and error dectection.

14.What is aliasing?

It is condition when two ro more variables reference to the same memory location

15.What is exception handling?

It allows a program to detect and respond to runtime error or other exceptional conditions.

16.Why is readability important to writability?

Because code that is easier to read is easier to write, debug and maintain.

20. What two programming language deficiencies were discovered as a result of the research in software development in the 1970s?

a. Insufficient support for modularization and abstraction
b. Poor support for detecting errors early.

21. What are the three fundamental features of an object-oriented programming language?

Inheritance, encapsulation, and polymorphism

22. What language was the first to support the three fundamental features of object- oriented programming?

Small talk

23. What is an example of two language design criteria that are in direct conflict with each other?

Reliability vs efficiency: runtime checks for errors often involves additional overhead which can conflict the high execution efficiency goals.

24. What are the three general methods of implementing a programming language?

Compilation, interpretation, and Hybrid implementation

25. Which produces faster program execution, a compiler or a pure interpreter?

Complier is faster.

29. What are the advantages in implementing a language with a pure interpreter?

Immediate execution of code,

faster development cycle as write-test-debug cycle is quicker

easy to debug as errors are detected at runtime