CSCI 330-54

Ganzorig Arvinbileg

Hwork01

Question 1.

2.2. Those are the balanced ones: ((A) (B)), (A (B (C))), (((A) (B)) (C))
2.4. The parenthesis notation for the given cons cell structure is: ((BOWS ARROWS) (FLOWERS CHOCOLATES))
2.6. () – NIL
(()) – (NIL)
((())) – ((NIL))
(() ()) – (NIL NIL)
(() (())) - (NIL (NIL))
2.13. FUN – CAAAR

IN – CADAR

THE – CADAR

SUN – CADDR

2.15.

Function	Result
CAR	(A B)
CDDR	(E F)
CADR	(C D)
CDAR	(B)
CADAR	B
CDDAR	NIL
CAAR	A
CDADDR	(F)
CADADDR	F

2.16. When you apply for CAAR to (FRED NIL) it will cause an error because FRED is not a list.

Question 2.

6. UNIX is mostly written on C programming language.

7. Having too many features in a programming language can make things feel way more complicated than they need to be. It's harder to learn because there's so much to wrap your head

around. Plus, reading someone else's code can be a real headache if they use stuff, you're not familiar with, especially if they learned a different part of the language than you did.
8. Letting people define their own operators can make code harder to follow because it messes with what you'd normally expect those operators to do. Instead of sticking to the usual meanings, you must figure out how someone decided to change them, which can be confusing.

9. One example of a lack of orthogonality in the design of C is an array element can be any data type except void or a function.

10. Ada uses orthogonality as a primary design criterion.

11. The goto statement is a basic tool that's used to create more complex things like loops and conditionals in languages that don't already have those built in.

12. A program is reliable if it performs to its specifications under all conditions.

13. Checking the types of parameters in a subprogram is super important because it makes sure the values you pass in match the type of the subprogram is expecting. This helps avoid errors and keeps things running smoothly.

14. Aliasing has two or more distinct names in a program that can be used to access the same memory cell.

15. Exception handling is basically how a program deals with things going wrong while it's running. Instead of crashing, the program can catch the problem, figure out what happened, and either fix it, show a helpful error message, or do something else to keep running smoothly. It's like having a backup plan for when things don't go as expected.

16. Readability affects reliability in both the writing and maintenance phases of the life cycle. Programs that are difficult to read are difficult both to write and modify.

20. Back in the 1970s, researchers realized that programming languages had some big issues, like not properly checking types and relying too much on things like goto for control flow. These problems pushed developers to improve things by creating structured programming and making type safety a bigger priority.

21. The three fundamental features of object-oriented programming language are data abstraction, inheritance, and dynamic binding.

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

23. Two criteria that conflict are reliability and cost of execution. For example, conflicting criteria that lead directly to design trade-offs, consider the case of APL. Because of the large number of operators, a significant number of new symbols had to be included in APL representing the operators.

24. The three general methods of implementing a programming language are compiler implementation, source language, and parse trees.

25. Compiler is faster than the interpreter.

29. Using a pure interpreter has its perks, like making it easier to handle debugging at the source level. Since everything is executed directly from the source code, any runtime error messages can point straight to the specific lines or parts of the code where things went wrong.