

## CSCI330-54|Hwork1

### Zoljargal Enkhbayar

#### Q1.

2.2. The ones with properly balanced parenthesis are, ((A) (B)), (A (B (C))), (((A) (B)) (C)).

2.4. The parenthesis notation for the cons cell structure is ((BOWS ARROWS) (FLOWERS CHOCOLATES))

2.6. () – NIL

(( )) – (NIL)

(( ( )) ) – ((NIL))

(( ) ( )) – (NIL NIL)

(( ) ( ( )) ) – (NIL (NIL))

2.13. Step      Result

*Start*      (((FUN)) (IN THE) (SUN))

C..AR      ((FUN))

C.AAR      (FUN)

CAAAR      FUN

Step      Result

*Start*      (((FUN)) (IN THE) (SUN))

C..DR      (IN THE)

C..ADR      IN

CADAR      IN

Step      Result

*Start*      (((FUN)) (IN THE) (SUN))

C..DR      (IN THE)

C.DDR      (THE)

CADAR      THE

Step      Result

*Start*      (((FUN)) (IN THE) (SUN))

C..DDR      ((SUN))

C.ADDR (SUN)

CADDR SUN

2.15.	<u>Function</u>	<u>Result</u>
	CAR	(A B)
	CDDR	(E F)
	CADR	(C D)
	CDAR	(B)
	CADAR	B
	CDDAR	NIL
	CAAR	A
	CDADDR	(F)
	CADADDR	F

2.16. Applying CAAR to (FRED NIL) would cause an error because FRED is an element not a list. Therefore, CAR would not be applied.

## Q2.

6. Most of the UNIX is written in the C programming language.

7. The disadvantage of having too many features in a programming language is that it can make things more complicated and harder to read. Learning the language becomes more challenging because there's so much to take in. Reading code can also be tricky if the person who wrote it uses features, you're not familiar with, especially if they learned a different part of the language than you did.

8. User-defined operator overloading can harm readability by making code behavior unclear or unexpected, forcing readers to understand custom implementations instead of relying on familiar operator meanings

9. One example of a lack of orthogonality in the design of C is record can be returned from functions but cannot.

10. The language that uses orthogonality as a primary design criterion is Ada.

11. The goto statement is the primitive control statement used to build more complicated control statements, such as loops and conditionals, in languages that lack these higher-level constructs.

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

13. Type checking parameters of a subprogram is important because it ensures that the actual parameters passed to subprogram match the expected types of the formal parameters defined in subprogram.

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

15. Exception handling is a language feature that allows for the interception of run-time errors, enables corrective actions to be taken, and ensures that the program can continue running, which significantly enhances reliability.

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. Research in the 1970s identified incomplete type checking and inadequate control statements, such as excessive reliance on goto, as a major deficiency in programming languages, prompting advancements in structured programming and type safety.

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

22. The first language to support the three fundamental features of OOP was smalltalk.

23. Two criteria that conflict are reliability and cost of execution. For example, Java ensures reliability by checking array index bounds, increasing execution cost. C skips these checks prioritizing efficiency but sacrificing reliability.

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

25. Compiler executes the program faster than the interpreter.

29. The advantages in implementing a language with pure interpreter are allowing easy implementation of many source-level debugging operations, because all run time error messages can refer to source-level units.