HW1

1.

2.2

(A B (C): yes

((A) (B)): no

A B) (C D): no

(A (B (C)) : no

(A (B (C))): yes

(((A) (B)) (C))

2.4

((Bow Arrows) (Flowers Chocolates))

2.6

() = nil

(()) = (nil)

((())) = ((nil))

(() ()) = (nil nil)

(() (())) = (nil (nil))

2.13

Step	Result
Start	(((FUN))(IN THE)(SUN))
Car	Fun
C..DR	((IN THE)(SUN))
C.adr	(IN THE)
Caadr	In

C…DR	((IN THE)(SUN))
C..adr	(IN THE)
C.dadr	(THE)
Cadadr	THE

C…dr	((IN THE)(SUN))
C.ddr	(SUN)
Caddr	SUN

2.15

(E F)

(C)

(B)

CADAR

Nil

CAAR

(F)

CAADDR

2.16

It would give Fred

2.

6. C

7. It makes the language harder to learn

8. Users can have overly complex overloads that make little sense which harms readability.

9. Arrays hamper C’s orthogonality as they are exceptions to most of the languages rules.

10. Algol

11. Functions

12. If the program performs to its specifications under all conditions

13. Bug fixing outside of compilation is difficult. Type checking the parameters in compilation prevents.

14. Having one memory cell get called two different things.

15. Intercepting, correcting and subsequently working around run time errors.

16. A language that is easier to understand is easier to write simply.

20. Incompletenss of type checking and inadequacy of control statements.

21. Data abstraction, inheritance, and dynamic method binding.

22. Smalltalk

23. Reliability and cost of execution

24. Operating system, language implementations, and the machine language interface

25. A compiler is faster.

29. Runtime errors are easier to catch. No need for translation as well.