Name: _____

3 questions in 70 minutes

Question 1 (20%)

Give the sequences of car's and cdr's needed to get `x` in the following expressions; for convenience name the list under discussion as `lst` – the first one is answered to clarify the question:

(a) `(a x b d)`
`(car (cdr lst))`

(b) `(a (b (x)) d)`

(c) `(a (b (x d)))`

(d) `(a (b (d) x c))`

(e) `(((a (b (x c) d))))`

Question 2 (40%)

A given set *A* is a **subset** of another set *B* if and only if all the members of *A* are also a member of *B*. Two sets are **equivalent**, if and only if they are subsets of each other. For this problem you will represent sets via lists.

(a) Define a procedure `SUBSETP` that takes two list arguments and decides whether the first is a subset of the second.

(b) Define a procedure `EQUIP` that takes two list arguments and decides whether the two are equivalent.

(c) Define a procedure `IDENP` that takes two list arguments and decides whether the two have the same elements in the same order – do not directly compare the lists with `EQUALP`, you are required to do a element by element comparison.

Question 3 (40%)

Assume you have data that pairs employees' last names with their monthly salaries. Define a procedure that takes as input employee dat and (ii) a threshold salary (an integer), and returns in a list the last names of all the employees that earn above the threshold salary. Define two versions, one with, and one without an accumulator.