**Quiz 7b**

Question 1 (1 + 1 points)

1. (1 point) Given the following situations let us know whether class A should be a parent of class B in OOP? (answer “Yes” or “No”) **If they get two out of three, it is ½ point and all three are a full point.**
   1. Simulating a bakery. Class A: Bread; Class B: Sourdough **Yes**
   2. Simulating a living room. Class A: Couch; Class B: Furniture **No**
   3. Simulating a classroom. Class A: Teacher; Class B: Student **No**
2. (1 point) Given the following situations should the given variables be a class variable or an instance variable? (answer “Class” or “Instance”)
   1. In a book class, the number of pages in a book. **Instance**
   2. In a guitar class, the number of guitars in the world. **Class**
   3. In a backpack class, the material of the backpack. **Instance**

Question 2 (2 + 3 + 3 points)

We are going to simulate a Dictionary using OOP as described in the lab.

1. (2 points) Define a book class. This class should contain three methods: ***publisher****,* which returns the sentence: (the publisher is <publisher>); ***dog-ear*** which should take in a page number; ***next-page*** which should return the page *after* the one you dog-eared.

(define-class (book publisher)

**(define-class (book publisher)**

**(instance-vars (current-page #f))**

**(method (publisher) (se ‘the ‘publisher ‘is publisher)) .5 points for publisher**

**(method (dog-ear page) (set! current-page page)) 1.5 points for dog-ear, next-page and**

**(method (next-page) (+ current-page 1)))) the instance variable.**

1. (3 points) Define a dictionary class. Every dictionary contains the following information: a publisher, and two lists. One list is a list of words, and the second list is a list of definitions. Both lists must be the same length. Instantiating a dictionary requires three arguments:

(define websters-dictionary (instantiate dictionary

‘Merriam-Webster

(list ‘cat ‘gangnam-style)

(list ‘(favorite animal of reddit) ‘(popular dance))))

The dictionary object accepts two messages: ***word-num*** asks for the number of words in the dictionary; ***define*** takes in a word as an argument and returns the corresponding definition (if the word isn’t in the dictionary, return “I don’t know”).

Make certain your dictionary class can do everything the book class can (but don’t simply rewrite the code).

Also, assume you have a procedure *get-location* that finds and returns the index of a word in a list (returns 0 for the first element, 1 for the second, etc.). *get-location* returns #f if there isn’t a match.

> (get-location ‘hello (list ‘hey ‘hi ‘hello))

2

> (ask websters-dictionary ‘define ‘cat)

(favorite animal of reddit)

(define-class (dictionary publisher words definitions)

**(define-class (dictionary publisher words definitions)**

**(parent (book publisher)) 1 point for parent**

**(method (word-num) (length words)) .5 points for words**

**(method (define wd) (if (member? wd words) 1.5 points for define**

**(nth (get-location wd words) definitions))**

**‘(I don’t know))))**

1. (3 points) Now we define a bookworm class. A bookworm is a bug that, much like bees has a hive mind. This means that given the message, ***word-num***, any individual bookworm can return the number of words that any bookworm has ever digested (repeated words are counted multiple times, i.e. if two bookworms digested the word “dog”, that would increment their collective count by two).

Bookworms are also able to ***digest*** *dictionaries*, although they can only digest *one at a time*. If a bookworm is told to digest a dictionary, it remembers every word and definition in that dictionary. Each time a dictionary is digested, word-num should be incremented by the number of words in the dictionary(it doesn’t matter if they digest the same word).

Bookworms also have a ***define*** method that takes in a word and returns the word’s definition in the same way the book did.

> (define wormy (instantiate bookworm))

> (define worm-bert (instantiate bookworm))

> (define urban-dictionary (instantiate ‘internet-people (list ‘crosstext) (list ‘(sending the wrong text to someone))))

> (ask wormy ‘digest websters-dictionary) ;instantiated above

> (ask worm-bert ‘digest websters-dictionary)

> (ask wormy ‘ define ‘cat)

(favorite animal of reddit)

> (ask wormy ‘digest urban-dictionary)

> (ask wormy ‘define ‘cat)

(I don’t know)

> (ask worm-bert ‘word-num)

5

> (ask worm-bert ‘define ‘gangnam-style)

(popular dance)

**(define-class (bookworm)**

**(class-vars (word-number 0))**

**(instance-vars (book #f))**

**(method (digest bk) (set! book bk) 1 point for each method: digest, word-num, define**

**(set! word-number (+ word-number (ask book word-num))))**

**(method (word-num) word-number)**

**(method (define wd) (ask book ‘define ‘wd)))**