

CSC241 – Lab 7

Instructions: Submit a single file lab7.py containing solutions to the following problems.

Make sure you run the doctest to check your solutions before you submit. Copy and paste the results of the doctest in a multiline comment at the top of your submission.

- 1) Write function `reverse()` that takes as input a dictionary and returns another (new) dictionary that is the “reverse” of the original, that is, the pairs are the same but with the keys and values switched. Remember:
 - a) that dictionaries are not ordered so you don’t expect the reversed dictionary to have the same order indicated here,
 - b) To create an empty dictionary say: `d = {}`,
 - c) To retrieve a value say: `d[key]`,
 - d) To make a new entry say: `d[key]=value`.

```
>>> pb = {'eric':'123-4567','sue':'999-9999','sally':'3333'}
>>> pb['sue']
'999-9999'
>>> reverse( pb )
{'999-9999': 'sue', '3333': 'sally', '123-4567': 'eric'}
>>> reverse( pb )['3333']
'sally'
```

- 2) Write a function `letter2number()` that takes a string representing a letter grade as a parameter and returns the grade point associated with that grade. The grade letter will be one of A, B, C, D, or F (upper- or lowercase). It may include a plus or minus after the letter. An A corresponds to a grade point of 4, a B to 3, a C to 2, a D to 1, and an F to 0. A plus increases the base grade point by 0.3 and a minus decreases it by 0.3. There is no such thing as an A+ or an F-. If the user provides a string as an argument that doesn't correspond to a valid grade, the function returns the string 'unknown grade'. You MUST use a dictionary for this problem. Any solution that does not involve a dictionary will not earn full credit. The following shows the `letter2number()` function as used on several different arguments:

```
>>> letter2number('A-')
3.7
>>> letter2number('c+')
2.3
>>> letter2number('f')
0
>>> letter2number('e')
'unknown grade'
```

- 3) Write function `names()` that takes no input and repeatedly asks the user to enter the first name of a student in class. When the user enters the empty string, the function print the number of students with each name that are in the class. For full credit the names are not case sensitive, i.e. `'FRANK'` and `'fRAnk'` are all considered the same as `'Frank'`. And, the list of students MUST be printed in alphabetical order (hints: use `list(dict.keys())` to get the dictionary keys and convert them to a list, and also, lists have a sort method) Sample usage:

```
>>> names()
Enter next name: Frank
Enter next name: frank
Enter next name: Sue
Enter next name: sue
Enter next name: SUE
Enter next name: fraNK
Enter next name:
There are 3 students named FRANK
There are 3 students named SUE
```