

## 22C:16 Practice Problem Set 10

### Complete before Tuesday, 4-14-2015

---

The focus of this problem set is on evaluating expressions involving Python dictionary operations and on writing short functions involving dictionaries.

- Suppose that D is the dictionary {"what": "why", "are": "why", "you": "what", "why": "you", "next": "are", "hello": "are"}. Given below are a bunch of expressions. Write down what each expression evaluates to.
  - `list(D.keys())`
  - `D[D["next"]]`
  - `list(D.values())`
  - `list(D.items())`
  - `D[D[D["you"]]]`
- Suppose that D is the dictionary {"what":22, "are":11, "you":14, "doing":5, "next":9, "Saturday?":4}. Write down what the value of D is after each of the following Python statements. Evaluate each statement starting with the same value of the dictionary D, mentioned above.
  - `D["what"] = D["are"]`
  - `D.update({"Sunday":25, "what":5})`
  - `del D["you"]`
  - `D["which"] = 19`
  - `D.clear()`
  - `D["you"] = D["doing"] + D["next"]`
  - `D[D["you"]] = "you"`
- Suppose that D is the dictionary {"what": "why", "are": "why", "you": "what", "why": "you", "next": "are", "hello": "are"}. Write down what the value of D is after each of the following Python statements. Evaluate each statement starting with the same value of the dictionary D, mentioned above.
  - `del D["are"]`
  - `D[D["next"]] = D["you"]`
  - `D.update({"you" : "why", "skip": "hello"})`
  - `D.update({"you" : "you"})`
  - `Dclear()`
- Suppose that D is the dictionary {"what":22, "are":11, "you":14, "doing":5, "next":9, "Saturday?":4}. What is a possible correct output for the following code fragment?

```
sum1 = []
sum2 = 0
for x in D.items()
    sum1 = sum1 + [x]
    sum2 = sum2 + D[x[0]]
print(sum1, sum2)
```

5. Suppose that `D` is the dictionary of 5-letter words and their “neighbors” constructed in the `playLaddersGame1` program that we wrote in class. Suppose that we want to write a function called `secondNeighbors` that takes a word `w` that appears as a key in `D` and returns the list of all words that can be obtained by replacing either one or two letters in `w`. Code for this function is provide below with some parts missing. Your task is to complete this function.

```
def secondNeighbors(w, D):
    neighborsList = D[w]
    secondNeighborsList = neighborsList
    for neighbor in neighborsList:
        # Find the list of neighbors of the word neighbor

        L = -----

        # For each word in list L, append it to secondNeighborsList,
        # if it is not already in it. Can be done in 3 lines of code.

        -----
        -----
        -----

    return secondNeighborsList
```

6. Suppose that `D` is the dictionary of 5-letter words and their “neighbors” constructed in the `playGame` program that we wrote in class. For words  $u, v$  appearing as keys in `D`,  $(u, v)$  is called an **isolated pair** if  $u$  has only one neighbor  $v$  and  $v$  has only one neighbor  $u$ . Write a function called `isolatedPairs` that takes this dictionary `D` as a parameter and returns the list of all isolated pairs. Thus the function should return a return a list of elements such that each element is itself a list of size 2. Note that each isolated pair  $(u, v)$  will appear in the returned list twice, once as `[u, v]` and once as `[v, u]`.
7. Write a function `deleteSmallFrequencies` that takes as its single parameter, a dictionary `D`. This keys of this dictionary are strings and the each key has an associated nonnegative integer as its value. You may think of the strings are words and the associated values as frequencies. The function `deleteSmallFrequencies` is required to return a new dictionary obtained by deleting from the given dictionary all words that have frequency less than or equal to 10. For example, if `D` were `{"hello":20, "hi":7, "text":3, "earthquake":1000}` then the function call `deleteSmallFrequencies(D)` should return `{"hello":20, "earthquake":1000}`.