COSC 1336: Fall 2022

Assignment for Chapter 12

DUE: December 5, 2022, by 1 AM

20% PENALTY PER EACH DAY

[30 points] Question 1

This question is a version of a question in the Assignment on Files, except that you need to return a dictionary and not write to a file.

Two example files are provided for testing: **`JackAndJill.txt`** and **`TheHillWeClimb.txt`.**

Your task is to write a function **`match_words_dict(inputFilename, Wordlist)`** that takes **`inputFilename`**, the name of a test file, and **`Wordlist`**, a list of words, as parameters and return a dictionary containing the number of times each word in the wordlist appears in the input text file.

Example:

```
wlist1 = ["Jack", "Jill", "up", "river"]
worddic = match_words_dict("JackandJill.txt", wlist1)
print(worddic)

wlist1 = ["up", "rise", "we", "nation"]
worddic = match_words_dict("TheHillWeClimb.txt", wlist1)
print(worddic)
```

Output:

```
{'Jack': 2, 'Jill': 2, 'up': 1, 'river': 0}
{'up': 1, 'rise': 4, 'we': 38, 'nation': 3}
```

[40 points] Question 2

Write a function called **`list_letters(text)`** that whose parameter **`text`** is a string of letters and returns a dictionary with the key as the letter and the value as the number of times each letter appears in a given string.

Case*:*

If the parameter string is **`'Banana'`**, then the dictionary returned should be:

```
{'B': 1, 'a': 3, 'n': 2}
```

If the parameter string is **`"How much wood could a woodchuck chuck if a woodchuck could chuck wood?"`** then the dictionary returned should be:

```
{'H': 1, 'o': 11, 'w': 5, ' ': 12, 'm': 1, 'u': 7, 'c': 11, 'h': 5, 'd': 6,
'l': 2, 'a': 2, 'k': 4, 'i': 1, 'f': 1, '?': 1}
```

Now write a function called **`list_words(text)`** whose **`text`** is a string of letters that consist of words separated by blanks. The function returns a dictionary with the keys as the unique words and the values as the number of times that word appears in the input string.

If the parameter string is **`'Banana'`**, then the dictionary returned should be:

```
{'Banana': 1}
```

If the parameter string is **`"How much wood could a woodchuck chuck if a woodchuck could chuck wood?"`** then the dictionary returned should be:

```
{'How': 1, 'much': 1, 'wood': 1, 'could': 2, 'a': 2, 'woodchuck': 2, 'chuck':
2, 'if': 1, 'wood?': 1}
```

Example:

```
text = "How much wood could a woodchuck chuck if a woodchuck could chuck wood?"
print(list_letters('Banana'))
print(list_letters(text))
print(list_words('Banana'))
print(list_words(text))
```

Output:

```
{'B': 1, 'a': 3, 'n': 2}
{'H': 1, 'o': 11, 'w': 5, ' ': 12, 'm': 1, 'u': 7, 'c': 11, 'h': 5, 'd': 6,
'l': 2, 'a': 2, 'k': 4, 'i': 1, 'f': 1, '?': 1}
{'Banana': 1}
{'How': 1, 'much': 1, 'wood': 1, 'could': 2, 'a': 2, 'woodchuck': 2, 'chuck':
2, 'if': 1, 'wood?': 1}
```

[30 points] Question 3

Write a function called **`check_inventory(inventory, low)`** that has as arguments a dictionary **`inventory`** containing the inventory of a store, and **`low`**, an integer. The function returns a list of items that are below an inventory level that is given by the **`low`** integer parameter.

Case:

`inventory = {'ibuprofen':800,'tylenol':366,'advil':75}`

`check_inventory(inventory, 100)` will return the list `['advil']`.

`check_inventory(inventory, 400)` will return the list `['tylenol', 'advil']`.

Example:

```
inventory = {'ibuprofen':800,'tylenol':366,'advil':75}
print(check_inventory(inventory, 100))
print(check_inventory(inventory, 400))
```

Output:

```
['advil']
['tylenol','advil']
```

[10 point] Question 4 – Extra Credit

Write a function called **`CI2(inventory, low)`** that is similar to **`check_inventory(inventory, low)`** from Question 3. It returns the same list as **`check_inventory`**, but it is sorted – with the lowest inventory item appearing first.

Case:

`inventory = {'ibuprofen':800,'tylenol':366,'advil':75}`
`check_inventory(inventory, 100)` will return the list `['advil']`
`check_inventory(inventory, 400)` will return the list `['advil', 'tylenol']`

Example:

```
inventory = {'ibuprofen':800,'tylenol':366,'advil':75}
print(CI2(inventory, 100))
print(CI2(inventory, 400))
```

Output:

```
['advil']
['advil', 'tylenol']
```