

Swift Assignment #2

Dictionaries

CS275 Fall 2021
15 points
due Tuesday, Sept. 21st, 11:59 pm

1 State-Towns Dictionary

You'll write functions to maintain a dictionary in which the keys are US state names, and the entries are arrays of cities/towns in each state.

1.1 The dictionary

Declare a dictionary of this form:

```
var stateInfo = [ String : [String] ] ()
```

1.2 Functions

Write these four functions:

```
func printDict(_ dict: [ String : [String] ])
```

This will print each key and the contents of each array for that key, with the array elements separated by spaces. If the dictionary is empty, print “(empty)”.

```
func addEntry(to key: String, town: String, in dict: inout [ String : [String] ])
```

If the key exists in the dictionary, this function will add an entry to the array of states for that key (duplicates are OK); otherwise, it will put a new key in the dictionary with an array containing the specified town.

```
func removeEntry(from key: String, town: String, in dict: inout [ String : [String] ])
```

If the key exists in the dictionary and the specified town appears in the array for that key, then this function will remove the specified town from the array for that key. If this results in an empty array for that key, then the function will remove the key from the dictionary.

Note: since we are not explicitly checking for duplicates in `addEntry(to:town:in:)`, there could be duplicates entries for a town in the array for a state. It's fine for `removeEntry(from:town:in:)` to remove only the first matching town in the array of towns for the specified state.

```
countEntries(for key: String, in dict: [ String : [String]]) -> Int
```

Return the number of entries in the array for the specified key. If the key is not in the dictionary, then return zero.

Do not do any forced unwrapping: there should not be a single `?` or `!` in your code.

The only dictionary function you will need is: `updateValue(_:forKey:)`. The array functions you will need are: `append(_:)`, `remove(at:)`, and `firstIndex(of:)`. (You'll have to do other operations on arrays and on the dictionary, but these are the only functions you'll need.)

1.3 Note on parameter passing

By way of explanation: dictionaries in Swift are *value types*: this means that as a function argument, a dictionary is a pass-by-value parameter, which means that it cannot be changed in the body of the function. This is why we must declare the dictionary as an `inout` parameter and use an ampersand in front of it in the function call itself—this declares the parameter as a pass-by-reference parameter.

2 Testing your functions

Here is a sequence of calls to make:

```
var stateInfo = [ String:[String] ] ()

printDict(stateInfo)
addEntry(to: "Vermont", town: "Burlington", in: &stateInfo)
addEntry(to: "Vermont", town: "Montpelier", in: &stateInfo)
addEntry(to: "Vermont", town: "Williston", in: &stateInfo)
addEntry(to: "Vermont", town: "South Burlington", in: &stateInfo)
addEntry(to: "Vermont", town: "Winooski", in: &stateInfo)
addEntry(to: "Massachusetts", town: "Boston", in: &stateInfo)
addEntry(to: "Massachusetts", town: "Beverly", in: &stateInfo)
addEntry(to: "Massachusetts", town: "Springfield", in: &stateInfo)
printDict(stateInfo)
removeEntry(from: "Vermont", town: "Montpelier", in: &stateInfo)
removeEntry(from: "Vermont", town: "Colchester", in: &stateInfo)
removeEntry(from: "Maine", town: "Portland", in: &stateInfo)
printDict(stateInfo)
print("# entries for Vermont = \(countEntries(for: "Vermont", in: stateInfo))")
print("# entries for Massachusetts = \(countEntries(for: "Massachusetts", in: stateInfo))")
print("# entries for Maine = \(countEntries(for: "Maine", in: stateInfo))")
addEntry(to: "Maine", town: "Orono", in: &stateInfo)
printDict(stateInfo)
removeEntry(from: "Maine", town: "Orono", in: &stateInfo)
printDict(stateInfo)
```

And here's the output you should get:

```
(empty)
Massachusetts: Boston Beverly Springfield
```

```
Vermont: Burlington Montpelier Williston South Burlington Winooski
Massachusetts: Boston Beverly Springfield
Vermont: Burlington Williston South Burlington Winooski
# entries for Vermont = 4
# entries for Massachusetts = 3
# entries for Maine = 0
Massachusetts: Boston Beverly Springfield
Maine: Orono
Vermont: Burlington Williston South Burlington Winooski
Massachusetts: Boston Beverly Springfield
Vermont: Burlington Williston South Burlington Winooski
```

3 Graduate Students

Modify the `addEntry(to:city:in:)` function so that it keeps the array for each key sorted alphabetically and does not insert a duplicate value. In other words, if I already have Burlington in the array for the key Vermont, then I should not be able to insert Burlington again for the key Vermont.

Also, modify the `printDict(_:)` function so that it prints a comma between each town in the list of towns for each state, like this:

```
Vermont: Burlington, Montpelier, South Burlington, Williston, Winooski
```

Do not print a comma after the last town for a state.

4 What to Submit

Do all of these in a single Xcode Playground. Submit just your Swift file. You can see your `.swift` file in your Playground directory, with the name `Contents.swift`. Rename it `assignment-two.netid.swift`, using your netid.