

## 1 Nested Lists

1. Suppose we have a nested list where each inner list contains strings that represent symptoms exhibited by the corresponding patient. We want to write a function that takes this list as a parameter and returns a new list containing integers. For each patient, the new list should contain the number of symptoms they were exhibiting. Here is an example:

```
>>> symptom_count(['fatigue', 'abdominal swelling', 'bruising'], ['loss of appetite', 'fatigue'])
[3, 2]
```

Reminder: you can use `list.append` to add elements to a list.

2. Python, all the elements of a list don't have to be the same type. That means it is possible to have a list like this: `['Milos', 'Jones', 48, 'male', 'smoker', 210]` which represents one person giving personal information. (The last number is total cholesterol in mg/dL.) Then we could have a list of elements like this to represent a list of people.

```
[ ['Milos', 'Jones', 48, 'male', 'smoker', 210],
  ['Delia', 'Chan', 39, 'female', 'non-smoker', 170],
  ['Denise', 'Ross', 62, 'female', 'non-smoker', 150] ]
```

Suppose we have such a list in the variable `patients`. Write code that would loop over this list and create a list of the last names of only the female patients.

3. Suppose we have list that represents repeated heart-rate measurements all for the same patient over a number of tests. Each inner-list is a test/situation and for that test, we monitored the heart rate for a little while taking a few measurements.

```
hr = [ [ 72, 75, 71, 73], # resting
        [ 91, 90, 94, 93], # walking slowly
        [ 130, 135, 139, 142], # running on treadmill
        [ 120, 118, 110, 105]] # after minute recovery
```

Write code to calculate the average for each subject. Then, write code to calculate the average for each *setting* (i.e., resting, walking slowly, etc.)

## 2 Files and While Loops

1. Create a file using Python and then open it using a text editor
2. The file `january06.txt` contains data from the UTM weather station for January 2006. Download it from the C4M website to your machine and put it in a directory of your choice.
  - (a) Open it up in Pyzo to see what it looks like.
  - (b) Write a Python program to open the file and read only the first line
  - (c) Read the second line (this is still a header)
  - (d) Read the third line into a variable `line`.
  - (e) What is the type of `line`?
  - (f) Call the method `split()` on `line` and save the return value. What is the type that is returned by this method?
  - (g) Look up the method `split()` in the Python 3 documentation.
3. Use a for-loop to print the contents of `january06.txt` up until the 5-th line
4. Write a function to print the first line of `january06.txt` where the temperature is above 0. Write two versions: one should use a while loop, and one should use a for-loop and a return statement in order to finish running when it finds a temperature above 0.
5. Write a program `max_rh.py` that asks the user for two days and prints the maximum relative-humidity measurement for the period between those days (including the days themselves.)