

Lab Assignment 5

[Start Assignment](#)

Due Tuesday by 11:59pm **Points** 20 **Submitting** a file upload
Available Sep 29 at 8am - Oct 7 at 11:59pm 9 days

Python Collections

Application: Frequency Distribution

Make sure you have read and understood

- **Unit Module 7**, and
- **Text Ch. 6 Dictionaries and Sets**

before submitting this assignment. Hand in only one submission.

Lab Assignment Objectives

1. Use Python lists as a means of storing data.
2. Use Python dictionaries to store associative data.
3. Implement algorithms to compute elementary statistics.

The Problem Description

What is Data? Every year, thousands of earthquakes occur around the world. The majority are so mild that they are only detected by scientists with monitoring equipment. Considering only those earthquakes that people sense, then the measure might well be around 35 on a given day. Some days, there may be only 10; on other days, there may be 40 or 50.

The description just provided is a way that earthquake occurrence might be reported through data - a collection of numeric items that have been observed, measured, or pooled by some means. These data items pertain to some event of interest.

Data can contain information in various forms and information can be used in a variety of ways. This information allows for generalizations to be construed about the aggregate collection of data items. Predictions may also be made about future events based on the data. This kind of data analysis is based on the mathematical science of statistics.

This lab task centers on data, information, and statistics. Focusing on ways that a programming language like Python can help perform some of the basic data processing tasks that are commonly

carried out with collections of data you will use lists and dictionaries to compute a frequency distribution of a supplied data set.

Understand the Application

Working with large amounts of data, some means of organizing the data's storage is needed so that processing the data can be managed in an orderly and efficient manner. Python's built-in collections provide a means for organizing and storing data values. In this lab you will explore using Python collections: lists and dictionaries to organize, process and analyze data.

The Program Specification

value	frequency
3	2
4	3
5	6
8	5

The mode of a data set is essentially a solution to the basic statistics problem of finding the **frequency distribution** - a representation of the number of times each value occurs in the data.

One way to display a frequency distribution is to show a two-column table. The first column gives the data time, and the second column gives the associated count. The table does need to show the items in order.

In this lab you will use a dictionary to compute and display a frequency table. Recall that a dictionary maintains items in the order they were inserted. Since our frequency distribution needs to show the items in order of occurrence, your challenge will be to sort the day in key order for accurate presentation in your table display.

The Testing Specification

Validate data sequence values are in a specified range.

user input is in the range [0 - 20].

Validate user input is in the range [0 - 20].

Write a test driver that will generate the frequency distribution for 2 validated data sequences:

data sequence 1: valid data range [0-9].

```
3,1,1,5,3,1,2,2,3,5,3,5,4,4,6,7,6,7,5,7,8,3,8,2,3,4,1,5,6,7
```

Example test run:

ITEM	FREQUENCY
1	4
2	3
3	6
4	3
5	5
6	3
7	4
8	2

data sequence 2: Programmer supplied data values that pertain to **YOUR** big data topic of interest. Specify and validate the data range prior to processing data collection.

Run both test cases as one test suite (i.e. use an **iterator** to generate 2 program test runs in one execution).

Provide the commented out copy of your test run validation in the **test driver file**.

Example Run:

Documentation Specification

Document your functions with **docstrings**.

What to Turn In

- Hand in two files: Solution source and the test driver
- File 1 : Application source solution
 - Use this title format for your source **.py** file : yournameLab5.py (*Note:* Use **YOUR** name).
 - Be sure to include these 2 parts in your source file:
 - Program Header Description
 - Source Code
- File 2 : Test Driver
 - You will need to import your application source module.
 - Use this title format for your source **.py** file : yournameLab5test.py (*Note:* Use **YOUR** name).
 - Be sure to include these 2 parts in your test driver file:

- Header Description
- Source Code
- Commented out copy of your test run validation

Submitting multiple files to an assignment

Your lab 5 assignment requires uploading more than one file; you should upload these **2** files as one submission.

In this assignment you need to upload: **2 .py** files (algorithm source module and test driver) - both part of one assignment submission. To add these files, the **Add Another File button** is clicked to **upload the two files one by one**. **Check to make sure that both files uploaded okay**. When finished click **Submit Assignment**.

Submission Resources

For more information on how to submit your assignment, please visit:

- [How do I submit an online assignment? Canvas Student Guide \(https://community.canvaslms.com/docs/DOC-9539\)](https://community.canvaslms.com/docs/DOC-9539)
- [Assignments Overview Canvas Video Guide \(https://community.canvaslms.com/videos/1122-assignments-overview-students\)](https://community.canvaslms.com/videos/1122-assignments-overview-students)
- [Assignments Submissions Canvas Video Guide \(https://community.canvaslms.com/videos/1121-assignment-submissions-students\)](https://community.canvaslms.com/videos/1121-assignment-submissions-students)

Questions?

Feel free to [ask](#) in the forum!