FUN WITH FUNCTIONS

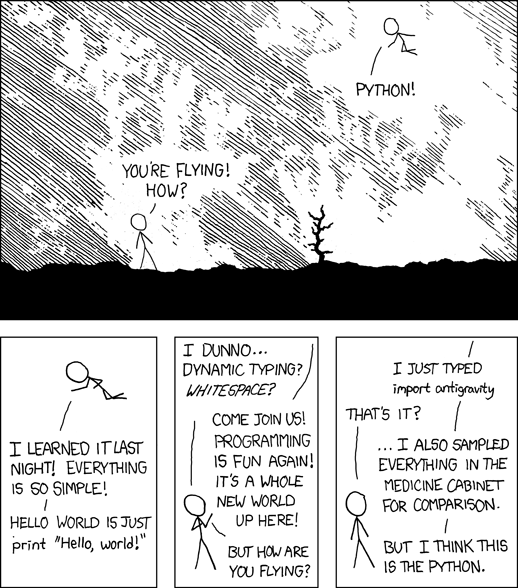
Lab 7

**What You’ll Learn in this Lab:**

* A better facility with functions, parameters, and return values
* A hands-on experience with top-down design
* More practice manipulating sounds in Python

**Getting Ready:**

* Read Sections 9.1 – 9.3 and Chapter 10 in the Python textbook
* Read Sections 6.5 and 7.1 of CSI
* Bring the following to lab:
  + Last lab’s Worksheet, ready to turn in at beginning of lab!!
  + Your class notes!
  + CSIS-110 Lab Manual
  + Both textbooks
  + Something to write with
  + The Lab 7 Worksheet



**Details**

*This section of each lab will involve practicing and experimenting with various aspects of our computer systems – editing, printing, mail, techniques for making your life easier, and saving your projects.*

***None this time!***

**Enhancing your Computer Science Knowledge**

*This section of each lab will involve problems and software that will augment your understanding of concepts that are fundamental to Computer Science.*

You will be developing a program called “Sound Board” that allows the user to choose from a variety of manipulations of 1, 2, or 3 sounds.

This program has already been started so you just need to complete it so it includes the functionality described below. Open the lab7Enhancing.py program provided and study it as you read this page; do not write any new code until you get to the next page of instructions.

The program should first ask the user how many sounds he or she will be manipulating (1, 2, or 3).

**If the user entered “1” for number of sounds**, then a function named processOneSound should be called (Hint: this function has been started but you need to complete it). The processOnceSound function needs to get the name of the file and ask if the user wants to (1) reverse the sound or (2) double the sound’s frequency.

If the user entered “1”, then the program should create a reversed copy of the sound then play both sounds (Hint: the reverse function has been provided for you. Do not change it.)

If the user entered “2” (double the sound’s frequency), then the program should create a doubled-frequency copy of the sound then play both sound (Hint: the double function has been provided for you. Do not change it).

**If the user entered “2” for number of sounds**, then a new function named processTwoSounds should be called (Hint: you need to create this function). The processTwoSounds function needs to get the names of the 2 files then ask if the user wants to (1) mix the sounds or (2) interleave the sounds.

If the user entered “1” (mix the sounds), then the program must ask for the percentage of each sound to use (two numbers adding up to 1), then create and play a new mixed sound (Hint: the makeMix function has been provided for you. Do not change it).

If the user entered “2” (interleave the sounds), then the program will prompt the user to enter an interleaving interval (number of samples to take from each sound before going to the other sound). The program will create a new sound and play that sound. For example, if the user enters 5000, then the program will create and play a new sound consisting of the first 5000 samples of the first sound, then the first 5000 samples of the second sound, then the next 5000 samples of the first sound, etc (Hint: the interleave function has been provided for you but it assumes the interval is always 3000; you need to modify by adding a parameter so that any interval can be used.)

**If the user entered “3” for number of sounds**, then a new function named processThreeSounds should be called (Hint: you need to create this function). The processThreeSounds function needs to get the names of the 3 files then ask for the percentage of each sound to use (three numbers adding up to 1), then create and play the new mixed sound (Hint: you need to create a new makeMix3 function but it will be very similar to the makeMix function provided so you can use that as a guide. makeMix mixes two sounds and your new makeMix3 function will mix three sounds).



***Complete the top-down design diagram on the worksheet.***

**Copy** the file “lab7Enhancing.py” from the proper folder in the common area into your own CSIS110 lab area. **Open** the file in JES. Complete the program, remembering to include comments describing every function and a comment at the top with your names. Some of the functions have been provided for you. Your program should have one function for each box in the top-down design diagram on the worksheet, plus the copy and clip functions provided for you. You may assume that the users only enter valid values. Test your functions using sounds from Media Sources or any other sounds you like. TIPS: (1) To play 2 sounds, one after the other, use “blockingPlay” instead of “play”. (2) Complete “processOneSound” first before moving on to anything else. (3) *Don’t forget, before you finish, to make the required changes to the “interleave” function as described in the comments for that function.*



***Print your copy of “lab7Enhancing.py” and attach it to the worksheet.***

**Extending and Expanding:**

*This section of each lab will involve using Python to answer questions and solve problems.*

Add another option to your program that allows a new action with 3 sound files: it will produce a new sound that starts with ½ second of the first sound, then mixes in the 2nd sound (from its beginning) for ½ second, then mixes in the 3rd sound (from its beginning), at which point all three sounds should continue to play, mixed, until the end of one of them is reached. You can visualize it like this:

sound 1

sound 2

sound 3



***Print your new copy of “lab7Enhancing.py” and attach it to the worksheet.***

**Reflection:**

*The discussion questions in this section of each lab are meant to make you think critically and creatively about some of the things you did earlier in the lab. Your answers to these questions must not be written on the lab worksheet, but on separate sheets of paper attached to the end of your lab worksheet. Your answers must be typed (not handwritten), and you will be graded on all aspects of your answer (correctness, use of proper terminology, readability, use of complete sentences only, etc.). In general you are expected to write at least one or two paragraphs in answer to each question.*

In this lab you developed programs using many functions. What were the advantages to using multiple functions instead of writing just one big function for the whole program? Be specific and thorough, listing as many advantages as you can, considering both the process of writing the program and what might happen with the programs and functions later. Also include any disadvantages you may have noticed.



Your program is not very user-friendly: for example, it only works properly when the user enters valid data; it does not give the user enough help or feedback; etc. If you were going to “release” this program for the general public to use, describe the changes you would make to make it more complete and professional. Try to think of all aspects of a user’s interaction with the program in your answer, and as always, be detailed and thorough.

