COMP201: INTRODUCTION TO COMPUTING

PROGRAMMING ASSIGNMENT #3

Announcement: OCT 31, 2017

Submission deadline: NOV 22, 2017

Description

In this assignment you will perform operations on sound clips in order to better understand multimedia processing and advanced skills in computer programming.

1. Create a function that loads in two sound files (must be WAV) and merges them together. To do this take the shortest (samples) of the two sound files and use that as your base length. Create a new sound where every odd sample belongs to one of the two sounds and every even sample belongs to the other sound.

2. Create a function that simulates a piano. To do this request user string such that all the middle keys (asd..jkl) of your keyboard correspond a specific note. So pressing 'a' would play a note and so would pressing 'h'. It is normal to have to press enter each time you press a key (to close the JES dialogue box). The number of characters in the string correspond to the time the note must be played. The first character decides the note to be played.

For example: 'aaaaaa' would play the same but longer note than 'aaa' or 'a', 'aaaa' is the same as 'aeth' seeing as there are 4 characters and the first one ('a') decides the note. All other note characters (first ones in the string) such as 'qw..op' are ignored and the program must continue. 'q' would be ignored', 'qaaaa' would be ignored. The single key 'x' must stop the program.

For example: 'aaaaaa' is inputed, user presses enter, a note relative to length of 6chars is played. 'adc' is inputted, user presses enter, a note relative to length of 3chars is played. 't' is inputted, user presses enter, nothing happens. 'x' is inputted, user presses enter, program stops.

3. Create your own sound:

A. Create a function that calculates the factorial of a number. If number is 4 then factorial of 4 is 4x3x2x1. [4!=4x3x2x1=24]

B. Create a function that calculates the power of a number to another number. This would take two inputs. If the first input is 4 and the second one is 2 then the output would be 4*4 = 16. If the first input is 10 and the second one is 3 then the output would be 10*10*10=1000

C. Create a function that calculates the absolute value of a number.

D. Create a function that calculates the SIN of a value. This must be done using Maclaurin Expansion. Basically

sin(x) = x - x^3/3! + x^5/5! - x^7/7! + x^9/9! - .....

Although ideally this expansion goes to infinity you can stop it where you want, so that it is accurate but also not too slow.

It must also have the same result as the default buit-in function sin().

E. Create a function that creates a note using the sin wave. A sin wave is A*sin(2*pi*f*t) where A is the amplitude of the wave (highest sample possible), pi is 3.14…., f is the frequency and t is the timestep. Create a list of samples that represent 1 second. This means that in this list you will step through each time step at frequency f. Note that your sin function will be slower that the default sin() function so you can use that one instead. Use sampling rates of higher than 1000.

Submission (electronic submission through Moodle only)

Please create a zip file containing your Python code, a readme text file (.txt). In the readme file document the features and functionality of the application, and anything else you want the grader to know.

Evaluation Procedure

You MUST demonstrate your solution program to the lab instructor during lab hours. You must run your submitted code, demonstrate its full functionality and answer questions about the Python programming aspects of your solution. Major marking is done on the spot during demonstration. Your code will be further checked for structure, non-plagiarism, etc. However, ONLY demonstrated submissions will receive marks. Other submissions will not be marked.

Credits

The assignment and solution was prepared by Timothy Forbes.