Media Computation Lecture 27 / Lab 8 Sound Playback Rate Name:_____

Today we'll have a hands-on lab to further familiarize you with modifying the sound frequency using a for-loop to modify the "playback rate" of a sound, so **pairs of students** should get a laptop from the cart.

Download to the Desktop and extract the files needed for the lab from:

http://www.cs.uni.edu/~fienup/cs1120s15/sessions/s27/lec27.zip

Part A. In JES run the downloaded `lec27/speedup.py` program and select the `antidisestablishmenterianism.wav` file.

```
def speedup(source):
  """ Returns the 2x-faster sound of the source """
  target = makeEmptySound(getLength(source)/2)

  for targetIndex in range(0,getLength(source)/2):
    sourceValue = getSampleValueAt(source, targetIndex*2)
    setSampleValueAt(target, targetIndex, sourceValue)

  return target
```

a) How did the sound change?

In JES run the downloaded `lec27/slowDown.py` program and select the `antidisestablishmenterianism.wav` file.

```
def slowDown(source):
  """ Returns the half-speed slower sound of the source """
  target = makeEmptySound(getLength(source)*2)

  for targetIndex in range(0,getLength(source)*2):
    sourceValue = getSampleValueAt(source, targetIndex/2)
    setSampleValueAt(target, targetIndex, sourceValue)

  return target
```

b) How did the sound change?

You **don't** need to show me that you did this, but you might want to refer to the above code to complete Part B.

Part B. In JES, open the downloaded `lec27/modifySoundPlaybackRate.py` partial program. Complete the code for the `modifyPlaybackRate(source, factor)` function. It should generalize the `source` sound's playback rate based on the parameter `factor`:

- a `factor` of 1 should be normal speed,
- a `factor` of 2 should be 2x-faster speed (same effect as `speedup.py` program),
- a `factor` of 0.5 should be half-speed (same effect `slowDown.py` program),
- a `factor` of 1.5 should be 1.5x-faster speed, etc.

Warning: the length and sound sample indexes must be integers, but the parameter `factor` is a real value (type `double`). Recall the `int()` cast function that takes a real value parameter and return the corresponding integer.

After you complete Part B, raise your hand and demonstrate your program.