Name: _____

CS 151 Fall 2010 at ISU, Exam 2 SAMPLE

Prepared by assistant professor Jeff Kinne on October 24, 2010. You have until 2:55pm to take the exam. I will give you a copy of the latest version of my study sheet that you can use, but this and blank paper I provide are all you will have to use (no computer, textbook, notes, cellphone, calculator, etc.). I have put point values on the questions so it adds up to 28. There are 2 true/false (1 point each), 2 multiple choice (1 point each), 2 fill in the code (2 points each), 2 say what the output is (2 points each), 2 short answer (3 points each), 1 explain how to write a function (5 points), and 1 give code for a function (5 points).

True/False

(1 point each, 2 points total). For each, circle True or False.

Problem 1 Suppose you run the following code:

```
sound = makeSound(pickAFile())
x = getSamplingRate(sound)
```

True or False: x is equal to the number of samples in the sound. **Answer: False.**

Problem 2 True or False: the sample right in the middle of a sound has index getLength(sound)/2. **Answer: True.**

Multiple Choice

(1 point each, 2 points total). For each of the following, circle the correct answer.

Problem 3 What is the total number of bytes needed to store the samples of a sound in JES, with 16-bits per sample, that is 4 minutes long, and has a sampling rate of 22050?

 a. 16*4*60*22050

 b. 2*4*60*22050

 c. 16*60*22050

 d. 16*4*60

Answer: b.

Problem 4 Suppose you have two sounds that you want to **splice** into one, the sounds are in variables called sound1 and sound2, and both are the default 22050 samples per second. Which of the following correctly creates a new sound that is the right length to have sound1 followed by sound2 in the new sound?

 a. `newSound = makeEmptySound(getLength(sound1) + getLength(sound2))`

 b. `newSound = makeEmptySoundBySeconds(getLength(sound1)+getLength(sound2))`

 c. `newSound = makeEmptySound(getLength(sound1)*getLength(sound2))`

 d. `newSound = makeEmptySound(getLength(sound1))`

Answer: a.

Fill in the code

(2 points each problem, 4 points total). For each of the following, fill in what the missing code should be to complete the function correctly.

Problem 5   Fill in the blanks of the following function that is supposed to compute the smallest sample value in the sound.

```
def minSampleValue(sound):
  smallest = 0
  for i in range(0, getLength(sound)):
    value = getSampleValueAt(sound, i)
    if _____:
      smallest = value
  return _____
```

First blank: **Answer: (value < smallest)**

Second blank: **Answer: smallest**

Problem 6   Suppose you have a sound, and you want to create a new sound that has a 1 second pause and then has a copy of the sound. The following is a partially completed function to do this. Fill in the blanks.

```
def addPause(sound):
  newSound = makeEmptySound(getLength(sound)+22050)
  for i in range(22050, getLength(newSound)):
    value = getSampleValueAt(_____,_____)
    setSampleValueAt(newSound, i, value)
  return newSound
```

First blank: **Answer: sound**

Second blank: **Answer: i-22050**

What is the output

(2 points each problem, 4 points total) For each of the following, give what the output of the funciton will be.

Problem 7 Suppose you have the following function loaded.

```
def myFunction(n):
  t = 0
  for i in range(0, n):
    t = t + 3
  print t
  print i
```

What values are printed by the function when you call >>> `myFunction(7)`?

First output: **Answer: 21**

Second output: **Answer: 6**

Problem 8 Suppose you have the following function loaded.

```
def huh(sound1, sound2):
  x = getLength(sound1)/float(getSamplingRate(sound1))
  y = getLength(sound2)/float(getSamplingRate(sound2))
  if (x > y):
    print x
    print "first one"
  if (y > x):
    print y
    print "second one"
  if (y == x):
    print y
    print "both"
```

If sounda and soundb are sounds that have sampling rate 22050, sounda has 44100 samples, and soundb has 66150 samples, what values are printed by the function when you call >>> `huh(soundb, sounda)`?

First output: **Answer: 3.0**

Second output: **Answer: first one**

Short Answer

(3 points each, 6 points total) Give a brief response for each of the following.

Problem 9   Exlain briefly how the computer (JES in particular) stores a sound.

   Answer: The sound can be thought of as a curve, where each point on the curve says what the air pressure is (sound is difference in pressure). In JES, we have the values of the curve at periodic intervals - these are called the "sample values". We choose some number of bits to store a sample value; using 16 bits, the sample value can be any integer between -32768 and 32767.

Problem 10   Suppose you have a sound object in JES that is someone saying something. Explain how you could look at the sound wave and figure out how many words were spoken.

   Answer: People normally pause, at least slightly, in between words. If you look at a sound wave, we should be able to see these pauses in between the words. The pauses will be parts in the sound where the curve is close to 0 for a little bit. We then count how many of these pauses there are in the sound wave.

Explain How To

(5 points) Explain how you would write a JES/Python function to accomplish the task.

Problem 11 Suppose you have a sound object in JES that is a musical note. Explain how you would write a function that creates a new sound that is the same note but one octave higher.

   Answer: A note is one octave higher if it is twice the frequency (not the sampling rate of the sound, but the frequency of the sound curve itself). Here is one way to make a new sound that is twice the frequency of the old sound. I would create a new sound that has twice the sampling rate as the original sound but the same duration in seconds, so it will have twice as many sample values. I will copy all the sample values from the original sound into the first half of the new sound, and copy them again into the second half of the sound. With all of the samples from the original sound happening within the first half of the new sound, the frequency of the sound wave will be twice that of the original (the cycle length will be half).

   [If I wanted to keep the same sampling rate, I could do the same thing but just copy every other sample value from the original sound into the first half of the new sound, and the same for the second half of the new sound.]

Give JES/Python Code

(5 points) Give JES/Python code to solve the problem.

Problem 12 Give the code for a function called makePartQuieter. It takes three inputs: a sound object, a beginning index, and an end index. It should make the sound 1/2 as loud for all samples between the beginning and end indices.

```
def makePartQuieter(sound, begin, end):
```

Answer:

```
def makePartQuieter(sound, begin, end):
  for i in range(sound, begin, end):
    value = getSampleValueAt(sound, i)
    setSampleValueAt(sound, value/2)
```