CSE 8A Lecture 14

• Exam 3: Don’t stress too much about it! (pick it up tomorrow!)
  – It was not as horrible as you were “clicking” ;-)  
  – Lot of grades to average (and one exam will be dropped from the average)  
  – Exam 3 average is 6.5 (so around 5 is still pretty good!)  
  – We know it was harder. It requires more practice and READING all book examples.

• Reading for next class: Chapter 9 and video.
  – Check website for new reading-question about the video.

• Finish PSA 7 tonight (if you need lab door code: tritonlink web)

• BRING HEADPHONES TO LAB TOMORROW!!

• Today’s topics: Sounds!
What are we doing next?

• Chapters 8, 9, 10:
  – Sound!

• Continue with computational basics from Picture work:
  – Iteration/looping, if statements, arrays
  – Emphasis on deeper understanding
  – Emphasis on Java terminology, features, mental “model” of how code is represented in the execution on the machine

• When working with Sound/SoundSample, compare and contrast to Picture/ Pixel examples.

• REMEMBER Turtle, Picture, “Sound” … are classes from this book, not from “standard” Java
Sounds

• Sound is a quasiperiodic pattern of waves of air pressure
  – Increase in air pressure is a compression
  – Decrease in air pressure is a rarefaction

• The strength of the compression and rarefaction is the amplitude of the sound

• The number of compression/rarefaction cycles per second is the frequency of the sound
Digitizing Sounds

• To process sounds in a computer, sound must be digitized

• A microphone converts instantaneous sound pressure level into voltage (+ voltage for compression, - voltage for rarefaction)

• Then the amplitude of the voltage can be converted to bits (digital integers) with an Analog-to-Digital converter (ADC)
Digital Sample Rate versus Sample Size

- Size determines max (and min) amplitude
  - CD audio: 16 bits per sample (per stereo channel)
  - Min: -32,768; Max: 32,767

- Rate is “how often we record an amplitude”
  - CD audio: 44,100 samples per second
  - A.k.a. 44.1 KHz sample rate
If the following sound were modified to be louder it would

A. Have lower frequency and stronger compressions/rarefactions
B. Have smaller amplitude and stronger compressions/rarefactions
C. Have higher frequency and stronger compressions/rarefactions
D. Have larger amplitude and stronger compressions/rarefactions
E. None of the above
1) Solo: (30 sec)
2) Discuss: (2 min)
3) Group: (20 sec)

If higher pitch?

A

B

C

D
Making bad music: What’s wrong with these decisions?

- I’ve decided that I don’t like the sampling rate and sample size provided by the book authors. Comment on my decision to use...

<table>
<thead>
<tr>
<th>Sampling Rate</th>
<th>Sample Size</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>5,000Hz</td>
<td>4 bits</td>
<td></td>
</tr>
<tr>
<td>100,000Hz</td>
<td>32 bits</td>
<td></td>
</tr>
</tbody>
</table>
Our Representation of Sound

String fileName = FileChooser.pickAFile();
Sound noise = new Sound(fileName);
SoundSample[] noiseArray = noise.getSamples();
noiseArray[3].setValue(0);
int foo = noiseArray[2].getValue();
Our Representation of Sound (Shorthand)

String fileName = FileChooser.pickAFile();
Sound noise = new Sound(fileName);
SoundSample[] noiseArray = noise.getSamples();
noiseArray[3].setValue(0);
int foo = noiseArray[2].getValue();
How would we fill in this SampleSound[]?

1) Solo: (60 sec)
2) Discuss: (2 min)
3) Group: (20 sec)

Sample Rate 3 Hz

<p>| | | | | | | | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>B</td>
<td>0</td>
<td>10k</td>
<td>-10k</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>C</td>
<td>0</td>
<td>10k</td>
<td>0</td>
<td>0</td>
<td>5k</td>
<td>0</td>
<td>-5k</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>D</td>
<td>0</td>
<td>5k</td>
<td>10k</td>
<td>5k</td>
<td>0</td>
<td>-5k</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
1) Solo: (60 sec)
2) Discuss: (2 min)
3) Group: (20 sec)

How would we fill in this SampleSound[]

Sample Rate 6 HZ
According to Nyquist’s Theorem what is the minimum sampling rate?

A. 1.5Hz
B. 3Hz
C. 6Hz
D. 10,000Hz
E. 20,000Hz
Write code that reduces the value of the second half of the SoundsSample array by half

```java
String fileName = FileChooser.pickAFile();
Sound noise = new Sound(fileName);
SoundSample[] noiseArray = noise.getSamples();

for (SoundSample sample: noiseArray) {
    int val = sample.getValue();
    sample.setValue(val/2);
}

for (int i = noiseArray.length/2; i < noiseArray.length; i++)
{
    SoundSample sample = noiseArray[i];
    int val = sample.getValue();
    sample.setValue(val/2);
}
```

1) Solo: 45 sec)
2) Discuss: (2 min)
3) Group: (20 sec)
What does that code do

A. Makes a lower pitched sound during first half of play

B. Makes a quieter sound during first half of play

C. Makes a lower pitched sound during second half of play

D. Makes a quieter sound during second half of play

E. For each SoundSample element in the second half of the array it gets the Value and stores that in an int and then sets the Value with something that is half that
For you to practice

• Write code that reduces the volume of every other SoundSample.
  – What does that really sound like?
Summary of Concepts

• Digital representations of sounds
• Manipulating sounds using loops
TODO

• Reading for next class: Chapter 9 + Video
  – More about sounds

• Lab tomorrow!
  – (HEADPHONES + remember to take your exam)
Exam 3. Problem 1

Part of a method in the Picture class

Careful with *corner* pixels! They should be as in the figure.

```java
for ( int x = 0; x < this.getWidth(); x++ )
{
    for ( int y = 0; y < this.getHeight(); y++ )
    {
        if ( y < 2 || y >= this.getHeight()-2 )
        {
            //set this pixel to WHITE
            this.getPixel(x, y).setColor( new Color(255, 255, 255) );
        }
        else if ( (x < 2 || x >= this.getWidth()-2) && (y>=2 && y<this.getHeight()-2) )
        {
            //set this pixel to BLACK
            this.getPixel(x, y).setColor( new Color(0, 0, 0) );
        }
    } // end of y loop
} // end of x loop
```
Exam 3. Problem 2

Picture source = new Picture("tritons.jpg"); //source image
Picture target; //new image
target = source.zoomIn_TopRight(); //target gets assigned the returned Picture

source: ![Source Image]

In the source picture, highlight (shade) the Pixel that should be copied to the location (2, 3) in the target in order to create the stretched picture correctly and write the coordinates of that source pixel in the blank below.

Pixel with coordinates (2, 3) in target
Coordinates of pixel from source that will be copied to position (2, 3) in target: ____(4,1)___________
public Picture zoomIn_TopRight(){
    Picture zoomedPic = new Picture(this.getWidth(), this.getHeight()); // or this
    Pixel sourcePix, targetPix;
    int xTarget, yTarget;
    double xSource, ySource;

    for ( xSource = (this.getWidth()/2), xTarget = 0;
        xTarget<this.getWidth(); xSource = xSource+0.5, xTarget = xTarget+1 ) {
        for ( ySource = 0, yTarget = 0;
            yTarget<this.getHeight(); ySource = ySource+0.5, yTarget = yTarget+1 ){
            // Access the "source" and "target" pixels:
            sourcePix = this.getPixel( (int) xSource, (int) ySource);
            // OTHER OPTION
            // sourcePix = this.getPixel(this.getWidth()/2+xTarget/2, yTarget/2);
            targetPix = zoomedPic.getPixel(xTarget,yTarget);

            // Assign source pixel color to target pixel
            targetPix.setColor(sourcePix.getColor());
        }
    }

    return zoomedPic;
}