CSE 8A Lecture 7

- Reading for next class: 5.2-5.3
- If you don’t see your RQ and P grades, please make sure your clicker is registered.
- PSA1 and 2 grading: by the end of the week, we hope!
- PSA3 – interview: due Saturday 5pm.
  - Remember your (complete) comments including partner history
  - Style in the assignments (we’ll talk today about it, start counting more credit in PSA4)
- PSA 4: discussion sections to get started
- Exam 2: Coming up next Thursday
- 1 week to ask for grading “corrections” (check piazza post and syllabus for details).
1. Can you have a for-loop in the body of another for-loop?

A. Yes you can
B. Yes, but you need special code at the top of the file to be able to do it.
C. No, that will give you a compiler error.
2. Assume I already have a Picture object name pictObj. What does pictObj.getPixels() and pictObj.getPixel(0,0) return respectively?

A. A single Pixel Object; An array of Pixel Objects
B. An array of Pixel Objects; A single Pixel Object
C. They both return a single Pixel object
D. They both return an array of Pixel objects
3. What line of code would you use to get a Pixel located at (50,50) from an already defined Picture object named pictObj?

A. `pictObj.getPixel(int 50, int 50);`
B. `pictObj.getPixel(50,50);`
C. `pictObj.getPixels();`
D. `pictObj.getPixels[50][50];`

READING QUIZ - NO TALKING
4. Which line completes the code so that it changes the color of every Pixel in the first column of Picture pictObj to blue?

```java
for (int index = 0; index < pictObj.getHeight(); index++)
{
    Pixel pixelObj = ____________________;
    pixelObj.setColor(Color.blue);
}
```

A. pictObj.getPixel(index, 0)
B. pictObj.getPixel(0,0)
C. pictObj.getPixel(0, index)
D. pictObj.getPixels(index,index)

**READING QUIZ - NO TALKING**
What picture most accurately describes what this code does?

```java
Pixel[] pixelArray = this.getPixels();
Pixel p;
Pixel q;
for(int index = 0; index < pixelArray.length-1; index++) {
    p = pixelArray[index];
    q = pixelArray[index+1];
    p.setRed(q.getRed());
    p.setBlue(q.getBlue());
    p.setGreen(q.getGreen());
}
```

A. 

B. 

C. 

D. None of these
Why does this code have an error?

A. It tries to access pixelArray[-1]
B. It tries to access pixelArray[0]
C. It tries to access pixelArray[pixelArray.length]
D. It tries to access pixelArray[pixelArray.length+1]
E. None of the above
Fill in the for(…….) to loop over pixels bottom right to top left

• Like this:

```java
Pixel[] pixArr = this.getPixels();

for (int i = pixArr.length - 1; i >= 0; i--) {
    //Some code doing set on pixArr[i]
}
```
True or False: The following code could be written using a for-loop.

```java
Pixel[] pixelArray = this.getPixels();
int index = 0;
while ( index < pixelArray.length )
{
    Pixel pix = pixelArray[index];
    pix.setGreen(255);
    index = index + 1;
}
```

A. True
B. False
while

```java
Pixel[] pixelArray = this.getPixels();
int index = 0;
while ( index < pixelArray.length )
{
    Pixel pix = pixelArray[index];
pix.setGreen(255);
    index = index + 1;
}
```

for

```java
Pixel[] pixelArray = this.getPixels();
for ( int index = 0; index < pixelArray.length; index++ )
{
    Pixel pix = pixelArray[index];
pix.setGreen(255);
}
```

for each

```java
Pixel[] pixelArray = this.getPixels();
for ( Pixel pix: pixelArray )
{
    pix.setGreen(255);
}
```

do...while

```java
Pixel[] pixelArray = this.getPixels();
int index = 0;
do {
    Pixel pix = pixelArray[index];
pix.setGreen(255);
    index = index + 1;
} while (index < pixelArray.length);
```

Which do you prefer? Why?
while vs. for vs. for each

- So when to use each? Sometimes it’s a matter of style, sometimes it’s ease of functionality

Choose the best loop to use in each of these situations
A. For
B. For each
C. While

1. You want to loop through a picture until you find a pixel that is all black. Then you want your loop to stop.

2. You want to loop through all the pixels in a picture and set each pixel red value equal to its green value

3. You want loop through the pixels in the first half of the picture and make them all black.
while vs. for vs. for each: Summary

Less chance of error

• for each
  – Use when you know you need to access and modify each pixel directly

• for
  – Use when you need to loop through a known number of pixels and need access to their index value

• while
  – Use when you are not sure how many pixels to loop through

Greater chance of error

Often, though, you can just choose the loop you like best
Nested Loops: How do they work?
What order are pixels changed?

• This is called a “nested loop” because it is a loop inside another loop. WHAT?!! Can you do that?!

```java
Pixel p;
for (int foo = 0; foo < getWidth(); foo++)
{
    for (int bar = 0; bar < getHeight(); bar++)
    {
        System.out.println( foo + " " + bar );
    }
}
```
Nested Loops: How do they work? What order are pixels changed?

- Nested loops unroll the same way!

```java
Pixel p;
for (int foo = 0; foo < getWidth(); foo++)
{
    System.out.println( foo );
}
```

If the width of the Picture is 2, this will unroll to:

```java
System.out.println( 0 );
System.out.println( 1 );
```
Nested Loops: How do they work? What order are pixels changed?

- Consider a simple loop

Pixel p;
for (int foo = 0; foo < getWidth(); foo++)
{
    for (int bar = 0; bar < getHeight(); bar++) {
        System.out.println(foo + " " + bar);
    }
}

If the width of the Picture is 2, this will unroll to:

for (int bar = 0; bar < getHeight(); bar++) {
    System.out.println(0 + " " + bar);
}
for (int bar = 0; bar < getHeight(); bar++) {
    System.out.println(1 + " " + bar);
}

Then these loops unroll too…
Nested Loops: How do they work? What order are pixels changed?

- A method in Picture.java... what does it print if width is 2 and height is 3?

```java
Pixel p;
for (int foo = 0; foo < getWidth(); foo++)
{
    for (int bar = 0; bar < getHeight(); bar++)
    {
        System.out.println( foo + " " + bar );
    }
}
```

Will be converted automatically to a string for printing

A. 0 0
0 1
1 0
1 1
2 0
2 1

B. 0 0
1 0
2 0
0 1
1 1
2 1

C. 0 0
1 0
2 0
0 2
1 1
2 2

D. 0 0
1 1
2 2
1 0
1 1
1 2

1) Solo: (30 sec)
2) Discuss: (2min)
3) Group: (30 sec)
Nested Loops: Tracing code

- A method in Picture.java... what does it print if width is 2 and height is 3?

```java
Pixel p;
for (int foo = 0; foo < getWidth(); foo++)
{
    for (int bar = 0; bar < getHeight(); bar++)
    {
        System.out.println( foo + " " + bar );
    }
}
```

foo

bar

0 0 0
0 1 2
1 0 1
1 1 2

2
3
What do these Picture methods do?
What are their return types?

• `getPixel(int x, int y)`
  `Pixel`

• `getHeight()`
  `int`

• `getWidth()`
  `int`
Nested Loops: How do they work? In what order are pixels changed?

- A method in Picture.java...

Pixel p;
for (int bar = 0; bar < getHeight(); bar++)
{
    for (int foo = 0; foo < getWidth(); foo++)
    {
        p = getPixel(foo, bar);
        p.setColor(Color.BLACK);
    }
}
Why does this have an error?

- In a method in Picture.java… assume height=50, width=100

```java
Pixel p;
for (int bar = 0; bar < getWidth(); bar++) {
    for (int foo = 0; foo < getHeight(); foo++) {
        p = getPixel(foo, bar);
        p.setColor(Color.BLACK);
    }
}
```

1) Solo: (30 sec)
2) Discuss: (2min)
3) Group: (30 sec)

A. It doesn’t, this loops across rows, top to bottom
B. It doesn’t, this loops down columns, left to right
C. It tries to index a pixel off the end of a row (x value too big)
D. It tries to index a pixel off the end of a column (y value too big)
Why did that have an error?

- The method `getPixel` in `Picture.java` with two parameters interprets the first one as an ‘x’ coordinate, and the second one as a ‘y’ coordinate of the Pixel to get.

- When you call that method to get a Pixel from a Picture, it doesn’t matter what the names of the variables are that you pass in!

- `getPixel(foo,bar)` or `getPixel(bar,foo)` or `getPixel(x,y)` or `getPixel(y,x)`…

- The first parameter is always interpreted as the ‘x’ coordinate, and the second one as the ‘y’ coordinate, of the pixel you want.
How to fix that error

• Since `bar` takes values 0 to `getWidth()`, it is acting like an ‘x’ coordinate
• Since `foo` takes values 0 to `getHeight()`, it is acting like a ‘y’ coordinate
• So pass `bar` as first argument, and `foo` as second argument, to `getPixel`:

  \[ p = \text{getPixel}(\text{bar}, \text{foo}); \]

• (Better yet: write `x` instead of `bar` and `y` instead of `foo`; the computer doesn’t care, but it makes the code clearer to a human reader!)
What’s with foo and bar anyway?

The use of *foo* in hacker and eventually in programming context may have begun in MIT's **Tech Model Railroad Club** (TMRC)

*Foobar* may have derived from the military acronym **FUBAR** and gained popularity because it is pronounced the same.

–Wikipedia foobar page

Despite their popularity, *foo* and *bar* are NOT good choices for variable names

As the name of a bar, it’s pretty good, though
Some comments on style

Pixel p; for (int bar = 0; bar < getWidth(); bar++)
    {
        for (int foo = 0; foo < getHeight(); foo++)
            {
                p = getPixel(foo, bar);
                p.setColor(Color.BLACK);
            }
    }

What’s wrong with this code?
Some comments on style

Meaningful variable names (generally more than 1 character)

Pixel pix;
for (int xpos = 0; xpos < getWidth(); xpos++)
{
  for (int ypos = 0; ypos < getHeight(); ypos++)
  {
    pix = getPixel(xpos, ypos);
    pix.setColor(Color.BLACK);
  }
}

Proper indentation (Dr. Java will help with this)

One statement per line

Lines not longer than 80 characters
TODO

- Keep working on PSA4 and show your cool images on Piazza!
- Reading and video for the next class.