CSE 8A Lecture 10

• Reading for next class: 6.7-6.8
• Lab tomorrow! (check email for seating)
• PSA5 tonight + PSA6 tomorrow (discussion sections!)
• Use of “missed interview” is automatic
• NEW policy to earn “slip days” with partnerships
  – +0.5 every partnered PSA from now (psa5) on
  – -0.5 slip day from each partner when you need to turn in up to 24 h late (slip day is “split” among partners)
  – 24h late MAX (so 1 slip day only)
• Today:
  – NEW: If-statements
  – REVIEW: Variable scope, return value, comments
What’s your PSA5 status?

• A. done and interviewed
• B. done but no interview
• C. almost done
• D. still got a ways to go
• E. just barely started/haven’t started
Interm Exam 2

• Nice job!
  – Median: 9
  – High score: 10 (105 students got it)

• Exams provide a minimum bar sanity check

• If you scored lower than a 7, it’s time to REALLY take action
  – Come to the lab/office hours and get help
  – Re-do all of the lab quizzes, clicker questions, reading questions, etc. and re-read all chapters so far.

CLICKERS OUT!
1. If I wanted the following lines to print “if statement was true”
   What could go in the blank to do this?

   ```java
   int x = 30;
   if(_______)
       System.out.println("if statement was true");
   ```

   A. x < 30
   B. x > 30
   C. x == 30
   D. x < 29
2. Consider the following code.

```c
int foo = 2;
int bar = 9;
```

Now, what is the result of evaluating the following expression, using the values `foo` and `bar` assigned above?

```c
((bar / 3) == foo)
```

A. True
B. False
C. 1
D. 0
3. What gets printed in the following:

```java
int x = 10, r = 10;
if(x < 10) //false
{
    x = 100;
}
else
{
    r = r + 20;
}
System.out.println(x+r);
```

A. 20
B. 40
C. 120
D. 140
4. What does the code below print?

```java
int x = 6;
if (x>7) {
    System.out.println("Dogs can fly");
}
if(x>5) {
    System.out.println("Cats can drive");
}
else {
    System.out.println("Non-Sensical nonsense");
}
```

A. Dogs can fly
B. Cats can drive
C. Non-Sensical nonsense
D. Nothing
// while loop version
public void takeSteps( int nSteps )
{
    int stepsTaken = 0;
    while ( stepsTaken < 5 ) {
        this.takeRandomStep( this.getRandomStep() ) ;
        stepsTaken = stepsTaken + 1;
    }
}

// POSSIBLE RESULT: for loop version
public void takeSteps( int nSteps )
{
    for (int stepsTaken = 0; stepsTaken < 5; stepsTaken++)
    {
        this.takeRandomStep( this.getRandomStep() ) ;
    }
}

public static void main(String[] args) {
    World w = new World();
    Turtle t = new Turtle( w );
    t.takeSteps( 10 );
}

Turtle t is NOT doing 10 iterations.
We need to use nSteps as loop condition
(stepsTaken<nSteps)
// What does it print if width is 3 and height 3?

Pixel p;
for (int x = 0; x < getWidth(); x++) {
    for (int y = 1; y < getHeight(); y++) {
        System.out.println(y + "#" + x);
    }
}

1 # 0
2 # 0
1 # 1
2 # 1
1 # 2
2 # 2
Exam 2, Problem 3. *Traversing a pixelArray*

// get the method to do what it says

```java
public void reducePartByHalfColor() {
    Pixel[] pixArray = this.getPixels();
    Pixel p;
    int red, green, blue;
    for (int index=pixArray.length/4; index<pixArray.length/2; index++)
    {
        p = pixArray[index];
        red = p.getRed()/2;
        green = p.getGreen()/2;
        blue = p.getBlue()/2;
        p.setRed(red);
        p.setGreen(green);
        p.setBlue(blue);
    }
}
```

White: (255, 255, 255)
Black: (0, 0, 0)

How would this part of the image appear:
A. Lighter
B. Darker
C. Greyer
D. Something else
Chapter 6: Conditionally modifying pixels

All pixels change if COORDINATES meet criteria

All pixels change if COLOR meets criteria

All pixels change if meet both a COLOR and COORDINATE criteria
public void fillBottom(Color newColor) {
    Pixel pix;
    for (int y = 0; y < this.getHeight(); y++) {
        for (int x = 0; x < this.getWidth(); x++) {
            {<<<SELECT LINE OF CODE>>>}
                pix = this.getPixel(x, y);
                pix.setColor(newColor);
            }
        }
    }
}

A) if (y < this.getHeight() / 2)
B) if (y > this.getHeight() / 2)
C) if (this.getPixel(x, y) < this.getHeight() / 2)
D) if (this.getPixel(x, y) > this.getHeight() / 2)
Does the order of the for loops matter?

public void fillBottom(Color newColor) {
    Pixel pix;
    for (int y = 0; y < this.getHeight(); y++) {
        for (int x = 0; x < this.getWidth(); x++) {
            pix = this.getPixel(x, y);
            pix.setColor(newColor);
        }
    }
}

A. Yes, since we are changing the bottom half, we have to “fill in” across the rows in the inner loop

B. Yes, because we need to make sure the if statement is checking y not x

C. No, the if statement controls the assignment
How many times is the variable pix assigned a value?

```
public void fillBottom(Color newColor) {
    Pixel pix;
    for (int y = 0; y < this.getHeight(); y++)
    {  
        for (int x = 0; x < this.getWidth(); x++)
        {
            pix = this.getPixel(x,y);
            pix.setColor(newColor);
        }
    }
}
```

A. 1  
B. this.getWidth() times  
C. this.getHeight() times  
D. this.getHeight() * this.getWidth() times  
E. this.getHeight() / 2 * this.getWidth() times
A stripy filter

```java
public void everyOtherColumn(Color newColor)
{
    Pixel pix;
    for (int y = 0; y < this.getHeight(); y++)
    {
        for (int x = 0; x < this.getWidth(); x = x + 2)
        {
            pix = this.getPixel(x,y);
            pix.setColor(newColor);
        }
    }
}
```
How many iterations of the loop body are executed?

A. \((\text{getHeight}()-1) \times (\text{getWidth}()-1)\)
B. \((\text{getHeight}()-1) \times (\text{getWidth}()-1)/2\)
C. \(\text{getHeight}() \times \text{getWidth}()\)
D. \(\text{getHeight}() \times \text{getWidth}()/2\)
E. None of the above are always true

```java
public void everyOtherColumn(Color newColor)
{
    Pixel pix;
    for (int y = 0; y < this.getHeight(); y++)
    {
        for (int x = 0; x < this.getWidth(); x = x + 2)
        {
            pix = this.getPixel(x, y);
            pix.setColor(newColor);
        }
    }
}
```
public void everyOtherColumn(Color newColor)
{
    Pixel pix;
    for (int y = 0; y < this.getHeight(); y++)
    {
        for (int x = 0; x < this.getWidth(); x++)
        {
            pix = this.getPixel(x, y);
            pix.setColor(newColor);
        }
    }
}

A) if(x<this.getWidth()/2)  
B) if(x<this.getHeight()/2) 
C) if ((x % 2) == 0)            
D) if ( (this.getPixel(x, y) % 2) == 0)
If you can do it both ways, which is "better"?

• Efficiency
  – How much time (computer instructions) does it take?
  – How much space (computer memory) does it take?

• Software Design
  – How readable is it for a human?
  – How easy is it to modify the code if needed, without introducing new bugs?
Which is more efficient?

• We just looked at these two approaches for setting pixel color in even-indexed columns:
  – (A) Looping over only some (x,y) pixels
  – (B) Looping over all pixels, with if statement inside

• Which is more efficient?
  A - computer time

• Why? B - possibly easier to write
More complex control for color change

• What’s the most efficient way to reduce red eye?
  – A) Restricted loop bounds, only loop over x,y in range
  – B) Unrestricted loop bounds, use if statement to check for x,y in range

Pixels change if meet both a COLOR and COORDINATE criteria
public void removeRedEye (Color newColor,  
        int startx, int endx, int starty, int endy ) {

    Pixel pix;
    for (int x = startx ; x < endx ; x++)
    {
        for (int y = starty ; y < endy ; y++)
        {
            pix = this.getPixel(x, y);
            if (pix.colorDistance(Color.red) < 167) {
                pix.setColor(newColor);
            }
        }
    }
}
Let’s do it the “inefficient” way

• Assume we loop over ALL pixels in picture…

• What if statement would we use to check for pixels in range?
public void removeRedEye(Color newColor,
    int startx, int endx, int starty, int endy)
{
    Pixel pix;
    for (int x = 0; x < getWidth(); x++)
    {
        for (int y = 0; y < getHeight(); y++)
        {
            if (startx <= x && x <= endx && starty <= y && y <= endy)
            {
                pix = this.getPixel(x, y);
                if (pix.colorDistance(Color.red) < 167)
                    pix.setColor(newColor);
            }
        }
    }
}
Real-world example of if-else rules (among other things)

Eamonn Keogh, UCR

if (temp > 104.0)
ALARM!
Temp too high!
Real-world example of if-else rules (among other things)

Eamonn Keogh, UCR

How long would it take to loop through this data? How could you find patterns?

TRILLIONS of data points (or more!)
Real-world example of if-else rules (among other things)

Usually, the sensors only display the last few minutes of data and figures such as the minimum and maximum temperature for that day. In most cases, the rest of the data is discarded. This is in part due to legal and privacy issues, which the researchers believe can be solved. It’s also because computer scientists didn’t have the tools to mine the vast amounts of data produced in pediatric intensive care units. That changed after Keogh and a group of researchers recently developed a new technique, which allows for searching of datasets with more than one trillion objects. That’s a larger set than the combined size of all datasets in all data mining papers ever published.

http://ucrtoday.ucr.edu/9667
Real-world example of if-else rules (among other things)

Eamonn Keogh, UCR

Keogh plans to use the archived data to develop algorithms that incorporate what he calls “if then rules” that can assist doctors. For example, if a heart beat looks like this, then a child may have difficulty breathing in five seconds.

http://ucrtoday.ucr.edu/9667
TODO

• Reading for next class: 6.7-6.8

• Finish PSA5