Announcements:
Ch 6 up soon. Grades for programs 1,2,3 now official.
Honors section 11:15 Friday in CS BLDG 303 (note change)
Captive Lab sessions: this afternoon 230-5, fri 10-1230

Program 4 collection instructions: on the website this afternoon. MAKE SURE YOU FOLLOW THE INSTRUCTIONS CAREFULLY!

Midterm: mon 10/17, 615-715, Thompson 104
I’ll post solution to old midterm

What to make of “this”

Project review – how to proceed. Review a 2–class solution to this problem: flip 1000 coins in main, report ratio of heads to tails. And: (using multiFlip – flip 10 coins 1000 times – which comes up more often – 4 heads or 6 heads?

Method writing –

Write a method that’s passed a string as a parameter and returns the very first character in the string, if there is one. Return the blank character otherwise.

Write a method that’s passed a string as a parameter and returns the string with all white space removed and all letters in lower case.

Write a method that’s passed two double values and prints the greater of the two to the console.

Write a method that’s passed three ints, and returns true if they are in order, e.g. 3 4 7 are in order, so are 4 4 4; but 3 7 5 is bad.

Write a method that’s passed two points and returns the point closer to the origin (in case of a tie, either will do)

Methods that return objects – a Point example that returns the calling point reflected through the y axis. (x,y) -> (-x,y)

```java
public class SimpleCoins{
    static final int HEADS = 1;
    static final int TAILS = 0;

    public int flip(){
        if (Math.random() < 0.5)
            return TAILS;
    }
}
```
else
    return HEADS;
}
public int multiFlip(int flips)
{
    int total = 0;
    for(int j = 0; j < flips; j++) { total += flip();}
    return total;
}

class SimplePt{

private int x;   private int y;

public SimplePt(int xx, int yy){
    x= xx;  y = yy;
}
public int getX(){
    return x;}
public int getY(){
    return y;}

public double dist(SimplePt other){
    double deltaX = (this.getX() - other.getX());
    double deltaY = (this.getY() - other.getY());
    return Math.sqrt(deltaX*deltaX + deltaY*deltaY);}

public SimplePt xProject(){
    SimplePt q = new SimplePt(this.getX(), 0);
    return q;}
}