

COS 488 - Homework 7 - Question 2

Matt Tyler

5/5

I have copied my code below:

```
/* *****  
 * Author:   Matt Tyler  
 *  
 * Description: A class implementing the Ehrenfest model. Takes two command-  
 * line arguments indicating the number of steps and the number of simulations  
 * to run, respectively, and simulates the Ehrenfest model starting with N  
 * balls in chamber A and no balls in chamber B. Plots a distribution of the  
 * number of balls in urn A after each simulation is done.  
 *  
 * Usage example: Run these three commands to conduct the required simulations:  
 *   java-algs4 Ehrenfest 1000 1000  
 *   java-algs4 Ehrenfest 10000 1000  
 *   java-algs4 Ehrenfest 100000 1000  
 * *****/  
  
import edu.princeton.cs.algs4.StdRandom;  
import edu.princeton.cs.algs4.StdDraw;  
import edu.princeton.cs.algs4.StdStats;  
  
public class Ehrenfest  
{  
    // chamberA[i] is true iff ball i is in chamber A  
    private boolean[] chamberA;  
  
    // Initialize every ball to be in chamber A  
    public Ehrenfest(int N)  
    {  
        chamberA = new boolean[N];  
        for (int i = 0; i < N; i++)  
            { chamberA[i] = true; }  
    }  
}
```

```

// Execute one step of the simulation
private void step()
{
    int i = StdRandom.uniform(chamberA.length);
    chamberA[i] = !chamberA[i];
}

// Return the number of balls in chamber A
private int inA()
{
    int inA = 0;
    for (int i = 0; i < chamberA.length; i++)
        { if (chamberA[i]) inA++; }
    return inA;
}

// Plot a histogram of the given distribution
static void plot(double[] freq)
{
    StdDraw.clear();
    StdDraw.setPenColor(StdDraw.BLACK);
    StdDraw.setXscale(-1, freq.length);
    StdDraw.setYscale(-1, StdStats.max(freq) + 1);
    StdStats.plotBars(freq);
}

public static void main(String[] args)
{
    final int N = 1000;

    int steps      = Integer.parseInt(args[0]);
    int simulations = Integer.parseInt(args[1]);

    double[] freq = new double[N + 1];

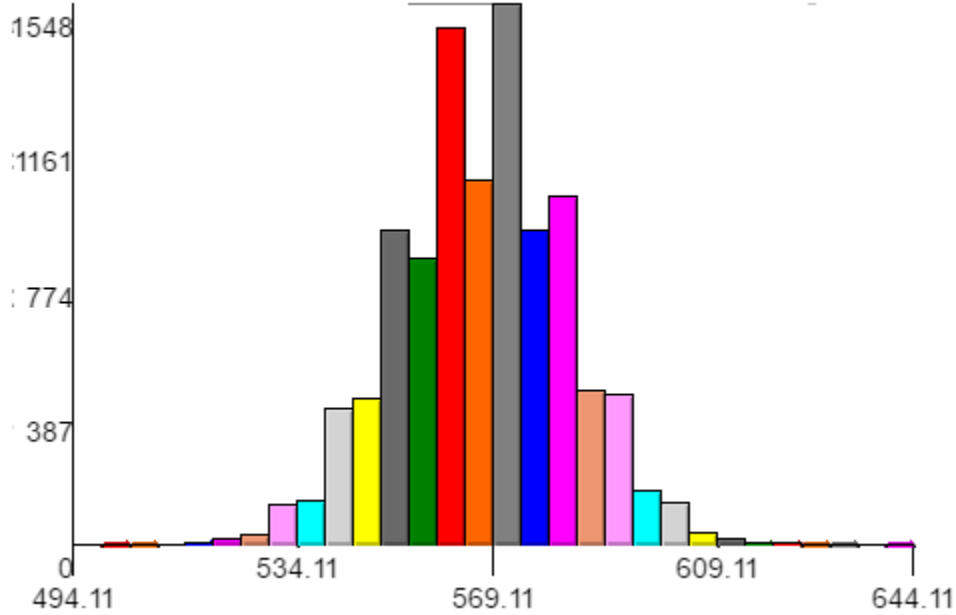
    for (int i = 0; i < simulations; i++)
    {
        Ehrenfest ehrenfest = new Ehrenfest(N);
        for (int j = 0; j < steps; j++)
            { ehrenfest.step(); }
        freq[ehrenfest.inA()]++;
    }

    plot(freq);
}
}

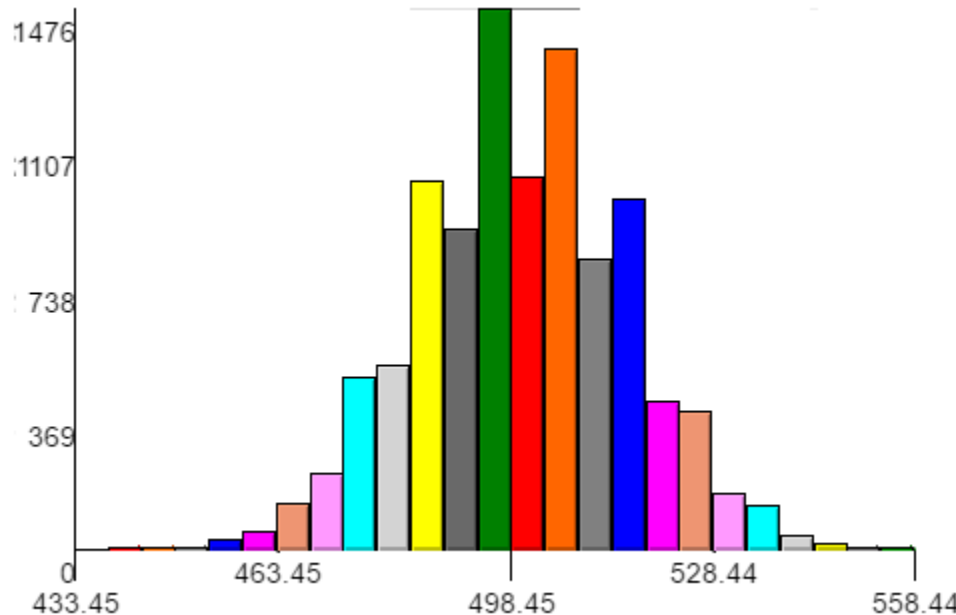
```

Since these are images and are therefore not executable, I have also copied the histograms below. (These are not the histograms directly output by this code, but rather ones made to look nicer with axis values and stuff. The data did, of course, come from this code.) The X-axis has been reduced so that only data with positive frequency is listed. Each simulation was conducted 10,000 times. The X-axis is the number of balls remaining in chamber A, and the Y-axis is the frequency.

This is the data with 10^3 steps:



This is the data with 10^4 steps:



This is the data with 10^5 steps:

