

PMR: This program was just a compilation of everything we learned this module. It was an easy program to build, and I also enjoyed building it.

/\*\*

\* The purpose of this program is to calculate the value of pi by simulating throwing darts at a dart board.

\*

\* @author Anika Jallipalli

\* @date 11/19/19

\*/

import java.util.Scanner;

public class EstimatePi

{

public static double x = (2 \* Math.random()) - 1, y = (2 \* Math.random()) -1, radius = 1.0;

public static double numberOfX = 0, numberOfY = 0, numberOfHits = 0, pi = 0, average = 0;

public static int dartThrows = 0, numberOfTrials = 0;

public static int hits (double x, double y, int numTrials)

{

numberOfHits = 0;

for(int n = 1; n < numTrials; n++)

{

x = (2 \* Math.random()) - 1;

y = (2 \* Math.random()) - 1;

if((Math.pow(x, 2) + Math.pow(y, 2)) <= (Math.pow(radius, 2)))

{

numberOfHits++;

}

}

return (int)numberOfHits;

}

public static double calculatePi (double numberOfHits, double dartThrows)

{

double pi = (4 \* ((double)(numberOfHits)) / dartThrows);

return pi;

}

public static void main (String[] args)

{

Scanner in = new Scanner(System.in);

System.out.println("How many times should the darts be thrown per trial?");

dartThrows = in.nextInt();

System.out.println();

System.out.println("How many trials should there be?");

numberOfTrials = in.nextInt();

double total = 0.0;

for(int n = 0; n < numberOfTrials; n++)

{

int hits = hits(x,y,dartThrows);

double approxPi = calculatePi(hits,dartThrows);

total += approxPi;

System.out.printf("Trial " + (n+1) + ": pi = %12f\n", approxPi);

}

double average = (total / numberOfTrials);

System.out.printf("Average Calculation of Pi: %1.5f", average);

}

}