Robb Hill

Dr. Gordon

CS3210

19 April 2020

Programming Assignment 3 Paper

Introduction

The goal of this assignment is to calculate algorithms probabilistically as opposed to their traditional methods. We will examine 5 common algorithms from the perspective of calculating their formulaic results using probability.

Experiments

There are 5 probabilistic algorithms we will explore

1. Computing Pi

This experiment will be to compute pi probabilistically. The idea will be a circle inside of a square with a determined length of side S = r*2, where r is the radius of the circle and N points will be randomly generated filling the area inside the square. If we take the number of points generated inside the circle and divide it by the total number of points generated, we should roughly compute pi.

2. Testing Primes

To test for prime numbers, we will randomly generate a number P and then randomly generate numbers less than or equal to the value of P to check if they are divisors of P. This is not to say that all numbers from 0 to P will be tested but instead randomly chosen numbers in that range. After a predetermined number of attempts K, we will determine if P is prime or if P had a random divisor dictating that it is not prime.

3. Searching Arrays

To search arrays in this experiment we will generate an array of random values named ary[N] where N is the number of values in the array. We will always generate an array of 1,000 ints. We will choose a target value from ary[ ] randomly, called X, and then randomly generate indexes i, such that i <= N, to test if they are the value of the target that was chosen, X. We will check if ary[i] == X. We will conduct this search no more than 5,000 times per trial to determine if the value gets found, and in that case, track the number of tries required before the value is found.

5. The 8 Queens Problem

The age-old N-Queens problem where queens must be placed on a chess board of NxN size without intersecting each other's paths by column, row and on diagonals. In this instance the board is fixed at the traditional 8x8 size. K queens are randomly placed on the board obeying the rules of no intersections. Then the remaining queens must be placed by backtracking until 8 queens are on the board.

Methods

1. Computing Pi

Several darts M will be thrown in total. The square will have side length S = 2 and inside of this square a circle will be drawn whose radius will be r = 1. The number of darts accumulating inside the circle will be N. This will simplify our formulas to:

area of circle = pi(r)^2 = pi

area of square = 2^2 = 4

ratio of areas = pi/4

M/N should be approximately pi/4

So…

4M/N should be approximately pi

A program will be written to simulate this scenario and the results will be its outputs.

2. Testing Primes

The code will mimic the situation that P is a prime such that P < 100 and n will be a random number such that n < P. If n divides P evenly then P shall be determined non-prime. The number n will be randomly generated K times. We will check the results of varying K in relation to the accuracy of prime rejection. We will try K = {10, 100, 1000, 10000}. The programs outputs will be our results.

3. Searching Arrays

An array of 1,000 ints will be randomly generated. One of these values will be randomly selected to serve as the target value. Next, random indexes will be generated up to 5,000 times to correctly locate the target value. This will be run 100 trials and 10,000 trials to find an average number of comparisons required to find the target. The results will be the output of the program.

4. Monte Carlo Integrations

A random, but interesting function will be chosen. Then a random point on the function will be chosen as the center of a rectangle that will enclose a certain section of the function. Then the value of the integration will be determined by throwing darts randomly into the rectangle and totaling the number of points above and below the function. This ratio will then be multiplied by the area of the rectangle to find the area under the function. This will be its approximate integration value.

Additionally, the area under the function will be calculated by finding the mean of the function at randomly selected points along the function within the set interval. This value will then be multiplied by the intervals width to approximate the area under the curve.

These two methods will then be compared to the trapezoid method for accuracy and run time of integral approximation.

5. The 8 Queens Problem

The 8 queens problem will be implemented by running the normal method for solving the n-queens problem, however the board will start with 4 queens placed by randomly selected column and row (then checked for collisions) then the remaining queens will be placed using the backtracking method by systemically placing the queens from lowest to highest valued rows and columns without collisions.

Results

1. Computing Pi

Darts thrown	Approximation of Pi	Runtime (seconds)
1,000	3.176	0.004
10,000	3.119	0.040
100,000	3.107	0.479
1,000,000	3.108	3.972
100,000,000	3.108	452.752

2. Testing Primes

Value of K	Detected Primes (trials run)	False Primes
10	100	53
100	100	4
1000	100	0
10000	100	0

Value of K	Detected Primes (trials run)	False Primes
10	10000	5750
100	10000	512
1000	10000	0
10000	10000	0

3. Searching Arrays

The average number of random guesses to find the correct index was 899

Trial #	Target Value	Tries to guess index
1	49946	23
2	42606	154
3	22103	359
4	32484	292
5	47928	362
6	1980	114
7	41372	622
8	65038	1729
9	53426	213
10	15255	142
11	37053	195
12	6343	1415
13	14283	277
14	64735	249
15	31775	1283

16	26726	121
17	10645	718
18	56564	1984
19	23258	241
20	10543	1000
21	41547	527
22	50044	475
23	53427	826
24	38635	2923
25	55505	469
26	38588	83
27	28027	1188
28	47509	1935
29	63632	1371
30	29671	633
31	44459	1764
32	45989	1476
33	45390	547
34	21143	445
35	49102	2040
36	54516	1670
37	49616	650
38	23638	25

39	10751	59
40	14946	10
41	36283	1221
42	48620	1246
43	61051	1075
44	65052	689
45	8635	548
46	18677	1923
47	46529	73
48	44609	1085
49	57912	832
50	47743	1244
51	10134	326
52	25172	474
53	55489	450
54	6904	1648
55	35101	283
56	43322	2502
57	11724	837
58	18515	435
59	22792	675
60	31977	279
61	29395	602

62	33253	956
63	34742	1180
64	870	169
65	45983	4192
66	22601	702
67	53707	1124
68	57903	35
69	32047	821
70	2584	1052
71	20861	762
72	31765	595
73	32903	1217
74	47527	4634
75	13728	2035
76	4744	146
77	64995	354
78	45076	1454
79	8803	125
80	52354	1299
81	26387	569
82	11744	1906
83	7221	619
84	38005	72

85	58111	1866
86	11401	803
87	42839	917
88	61685	87
89	32985	1730
90	14633	299
91	16636	1185
92	56157	367
93	204	1439
94	29450	564
95	639	135
96	54294	415
97	45143	2750
98	51627	383
99	50953	415
100	54699	1520

5. The 8 Queens Problem

Resulting boards were as follows:

– – – – – – – Q

– – – – – – – –

– – – – – – – –

– – – – – – – –

– – – – – – – –

– – – – – – – –

– – – – – – – –

– – – – – – – –

Q – – – – – – Q

– – – – – – Q –

– – – – Q – – –

– – – – – – – Q

– Q – – – – – –

– – – Q – – – –

– – – – – Q – –

– – Q – – – – –

– – – – – – Q –

– – – – Q – – –

– – – – – – – Q

– Q – – – – – –

– – – Q – – – –

– – – – – Q – –

– – Q – – – – –

Q – – – – – – Q

– – – – – – Q –

– – – – Q – – –

– – – – – – – Q

– Q – – – – – –

– – – Q – – – –

– – – – – Q – –

– – Q – – – – –
