Lab 4

Date Submitted: 11/29/2023

Name: Enrique Hernandez

COSC 3319 Section 01

Meeting Days: MWF

Grading Option: C / B

**Table:**

**Part D analysis:**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | **Burris Hash Linear** | **My Hash Linear** | **Burris Hash Random** | **My Hash Random** |  |  |
| **First 25** | **86%** | **86%** | **86%** | **86%** |  |  |
| **Min Probe** | **1** | **1** | **1** | **1** |  |  |
| **Max** | **11** | **1** | **8** | **1** |  |  |
| **Expected** | **4.28** | **1** | **2.92** | **1** |  |  |
| **Theoretical** | **4.07** | **4.07** | **4.07** | **4.07** |  |  |
|  |  |  |  |  |  |  |
| **Last 25** | **86%** | **86%** | **86%** | **86%** |  |  |
| **Min Probe** | **1** | **1** | **3** | **1** |  |  |
| **Max** | **104** | **22** | **57** | **11** |  |  |
| **Expected** | **61.76** | **8.24** | **23.24** | **4.24** |  |  |
| **Theoretical** | **4.07** | **4.07** | **4.07** | **4.07** |  |  |
|  |  |  |  |  |  |

**Burris Linear |My linear |Burris Random |My Random**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Whole Table** | **86%** |  | **86%** |  | **86%** |  | **86%** |  |
| **Expected** | **34.5** |  | **3.245** |  | **13.345** |  | **2.118** |  |
| **Theoretical** | **4.07** |  | **4.07** |  | **4.07** |  | **4.07** |  |

**Analysis for all parts:**

**Part A:**

The analysis revealed a notable disparity between the anticipated theoretical probe counts of 4.07 and the actual insertion results. While the initial 25 keys closely aligned with the expected values due to the table's initial population skewing toward lower probe counts, the subsequent addition of the last 25 keys starkly deviated, recording an average probe count of 61.96. This glaring divergence underscored the hash function's inefficiency, directly linked to an abundance of collisions and subsequent probe increments. These findings emphasize the pivotal role of an effective hash function in minimizing collisions and optimizing probe counts during key insertions.

**Part B:**

Like Part A, the observed results significantly diverged from the theoretical expectations across the table. The initial 25 keys performed better than expected, benefiting from the empty table scenario, increasing the likelihood of a single-hash insertion. However, the insertion of the last 25 keys yielded an average of 61.96 probes, far exceeding the expected theoretical count of 3. This disparity primarily arose due to the limitations of linear probing combined with an ineffective hash function. In cases where the table approaches 86% capacity, linear probing makes the issue by necessitating traversal through clusters, resulting in a notably higher probe count, as observed.

**Part C:**

When employing random probing with an 86% full hash table, the first 25 keys registered an average of 4.2 probes, mirroring the outcomes of linear probing due to the table's predominant emptiness at this stage. Conversely, the last 25 keys displayed an average of 4.24 probes, a notable improvement compared to the linear approach's 8.24 average. Despite this progress, both methods fell short of the theoretical expectation of 4.07 for this load factor. In the case of random probing with an 86% full table, the initial 25 keys mirrored the previous results due to similar circumstances. However, the last 25 keys experienced a dramatic increase in average probes, recording 4.24 probes. Although this figure remains inferior to the theoretical expectation, it signifies a marked enhancement over linear probing, showcasing nearly half the probe count of the linear method (8.24). While the theoretical benchmark remains lower, this progress marks a positive trajectory for the Burris hash function in its current state.

**Part D:**  
The disparity between theoretical and actual results stemmed from two primary reasons. Firstly, the hash function we used, provided by you, significantly underperformed, greatly impacting the overall results. Secondly, the theoretical probe count is calculated for the entire table, unlike the specific portions I compared—first 25 keys versus last 25 keys—where the table starts empty, causing varied comparisons. To address this, I introduced an additional comparison using the expected probes for the entire table against theoretical values. This clarified the stark differences, highlighting the shortcomings of the initial hash function. My custom-designed function outperformed the provided one comprehensively, surpassing theoretical expectations across the entire dataset. This comparison distinctly showcased the superiority of the function I developed.  
  
**Part E:**   
Dr. Burris your hashing algorithm suffers from inadequate distribution of weight among key components, leading to numerous shortcomings.

The algorithm's divisions by large divisors diminish the influence of Str(1) to Str(5), rendering their contributions almost negligible. Moreover, shorter input words exacerbate this issue by losing potential value from key components.

While Str(5) initially contributes, its division by 2\*\*1=-0 minimizes its impact. Consequently, Str(6), Str(7), and Str(10) play a more significant role in determining the hash address, with Str(7) and Str(10) often leading to collisions due to their smaller divisors.

In essence, the algorithm's uneven weight distribution among key parts limits its ability to create distinct hash addresses, resulting in a high collision rate.

**Code:**  
import java.util.\*;

import java.io.\*;

public class Hash {

private String[] hash = new String[128];

private double loadFactor;

private String probeMethod;

private String hashMethod;

public Hash(double load, String probe, String hash) {

this.loadFactor = load;

this.probeMethod = probe;

this.hashMethod = hash;

}

public void hash() {

if (hashMethod.equals("Burris")) {

if (probeMethod.equals("Linear")) {

try {

finalHash(loadFactor, hashMethod, probeMethod);

} catch (FileNotFoundException e) {

System.out.println("test");

}

} else if (probeMethod.equals("Random")) {

try {

finalHash(loadFactor, hashMethod, probeMethod);

} catch (FileNotFoundException e) {

System.out.println("test");

}

} else {

System.out.println("Error wrong selection");

}

} else if (hashMethod.equals("MyHash")) {

if (probeMethod.equals("Linear")) {

try {

finalHash(loadFactor, hashMethod, probeMethod);

} catch (FileNotFoundException e) {

System.out.println("test");

}

} else if (probeMethod.equals("Random")) {

try {

finalHash(loadFactor, hashMethod, probeMethod);

} catch (FileNotFoundException e) {

System.out.println("test");

}

} else {

System.out.println("Error wrong selection");

}

}

}

public static Integer slice(String str, int start, int end) {

// Performes slicing function

if (start < 0) {

start = 0;

}

if (end > str.length()) {

end = str.length();

}

if (start >= end) {

return 0;

}

return str.charAt(start) + str.charAt(end);

}

public static int BurrisPoorlyMadeHashFunction(String word) {

int key = 0;

if (word.length() >= 10) {

int slice1 = slice(word, 0, 1);

int slice2 = slice(word, 2, 3);

int slice3 = word.charAt(4);

int slice4 = word.charAt(5);

int slice5 = word.charAt(6);

int slice6 = word.charAt(9);

double power = Math.pow(2, 10);

key = (int) (((slice1 / 256.0 + slice2 / 277.0 + slice3) / power +

slice4 / 313.0 + slice5 / 3.0 + slice6) % 128);

}

return key;

}

public int myHash(String word) {

int key = 0;

long prime = 12582917;

for (int i = 0; i < word.length(); i++) {

key = (key \* (int) prime + word.charAt(i)) % 128;

}

return key;

}

public void averageWhole(int[] array) {

double average = 0;

double sum = 0;

int temp = 0;

for (int i = 0; i < 127; i++) {

sum = sum + array[i];

if (array[i] != 0) {

temp++;

}

}

average = sum / temp;

System.out.println(average);

}

public void firstminMaxAvgProbes(int[] array, int end) {

int max = -1, min = 800000000;

int sum = 0;

double average = 0;

for (int i = end - 25; i < end; i++) {

sum = sum + array[i];

if (array[i] > max) {

max = array[i];

}

if (array[i] < min) {

min = array[i];

}

}

average = sum / 25.0;

System.out.println();

System.out.println("Minimum probes is: " + min);

System.out.println("Maximum probes is: " + max);

System.out.println("Average probes is: " + average);

}

public int genRandom(int t) {

int temp = (int) Math.pow(2, 9);

int r = 1;

int q = 0;

int p = 0;

while (p != t) {

r = r \* 5;

r = r % temp;

q = r / 4;

p++;

}

return q;

}

public void display(int[] array) {

String str;

int key = 0;

int probes;

System.out.println("Index " + " Word " + " Probes"

+ " Original Hash");

for (int i = 0; i < 128; i++) {

probes = array[i];

if (hash[i] == null) {

str = "empty ";

key = -1;

} else {

str = hash[i];

if (hashMethod.equals("Burris")) {

key = BurrisPoorlyMadeHashFunction(str);

} else if (hashMethod.equals("MyHash")) {

key = myHash(str);

}

}

System.out.println(String.format("%-15s", String.valueOf(i)) + "| " + String.format("%-15s", str)

+ " | " + String.format("%-15s", String.valueOf(probes))

+ "| " + String.format("%-15s", String.valueOf(key)));

}

}

public void finalHash(double loadFact, String hashMethod, String probeMethod) throws FileNotFoundException {

int[] probes = new int[128];

int[] realprobes = new int[128];

double end = loadFact \* 128;

System.out.println("\nfinalHash" + " " + hashMethod + " " + probeMethod);

File file = new File("c:/Users/Esdeath/Desktop/Hash/Words200D16.txt");

Scanner scanner = new Scanner(file);

while (scanner.hasNext()) {

for (int i = 0; i <= 127 \* loadFact; i++) {

String line = scanner.nextLine();

int maxprobes = 1;

if (hashMethod.equals("Burris")) {

int key = BurrisPoorlyMadeHashFunction(line);

if (probeMethod.equals("Linear")) {

while (hash[key] != null) {

key = key + 1;

maxprobes = maxprobes + 1;

if (key == 128) {

key = 0;

}

}

realprobes[key] = maxprobes;

probes[i] = maxprobes;

hash[key] = line;

} else if (probeMethod.equals("Random")) {

int temp = key;

while (hash[temp] != null) {

temp = (genRandom(maxprobes) + key) % 128;

maxprobes = maxprobes + 1;

}

realprobes[temp] = maxprobes;

probes[i] = maxprobes;

hash[temp] = line;

}

} else if (hashMethod.equals("MyHash")) {

int key = myHash(line);

if (probeMethod.equals("Linear")) {

while (hash[key] != null) {

key = key + 1;

maxprobes = maxprobes + 1;

if (key == 128) {

key = 0;

}

}

realprobes[key] = maxprobes;

probes[i] = maxprobes;

hash[key] = line;

} else if (probeMethod.equals("Random")) {

int temp = key;

while (hash[temp] != null) {

temp = (genRandom(maxprobes) + key) % 128;

maxprobes = maxprobes + 1;

}

realprobes[temp] = maxprobes;

probes[i] = maxprobes;

hash[temp] = line;

}

}

}

display(realprobes);

System.out.print("\n\nFirst 25 keys:");

firstminMaxAvgProbes(probes, 25);

System.out.print("\n\nLast 25 keys:");

firstminMaxAvgProbes(probes, (int) end);

System.out.print("\n\nTotal Average:");

averageWhole(probes);

System.out.println("\n\nTheoretical Probes:");

theoretical();

}

scanner.close();

}

public void theoretical() {

double theo = 1 - (loadFactor / 2);

theo = theo / (1 - loadFactor);

System.out.println("Theoretical Probes: " + theo);

}

public static void main(String[] args) {

//Hash Mylin86 = new Hash(.86, "Linear", "MyHash");

Hash Myrand86 = new Hash(.86, "Random", "MyHash");

//Hash Brand86 = new Hash(.86, "Random", "Burris");

//Hash Blin86 = new Hash(.86, "Linear", "Burris");

//Blin86.hash();

//Brand86.hash();

//Mylin86.hash();

Myrand86.hash();

Myrand86.printOutputToFile("output.txt");

}

public void printOutputToFile(String filename) {

try {

// Create a new file output stream

FileOutputStream fos = new FileOutputStream(filename);

PrintStream ps = new PrintStream(fos);

System.setOut(ps); // Redirect output stream

Hash Myrand86 = new Hash(.86, "Random", "MyHash");

Myrand86.hash();

// Close the PrintStream and the file output stream

ps.close();

fos.close();

} catch (IOException e) {

e.printStackTrace();

}

}

}

**Output:**

finalHash Burris Linear

Index Word Probes Original Hash

0 | scat | 87 | 42

1 | toluene | 64 | 66

2 | wet | 89 | 42

3 | sparse | 89 | 43

4 | quandary | 63 | 70

5 | Necromancer | 1 | 5

6 | dactyl | 92 | 43

7 | nosegay | 64 | 72

8 | Irreversible | 1 | 8

9 | option | 95 | 43

10 | Christopher | 1 | 10

11 | forgetful | 74 | 66

12 | Constriction | 1 | 12

13 | privilege | 76 | 66

14 | orthodontist | 1 | 14

15 | sponsor | 74 | 70

16 | exhilarate | 6 | 11

17 | guard | 104 | 42

18 | empty | 0 | -1

19 | Afterwards | 1 | 19

20 | empty | 0 | -1

21 | empty | 0 | -1

22 | empty | 0 | -1

23 | ratiocination | 1 | 23

24 | Misogamist | 1 | 24

25 | parasympathetic | 2 | 24

26 | prepossess | 2 | 25

27 | rationalize | 2 | 26

28 | empty | 0 | -1

29 | empty | 0 | -1

30 | empty | 0 | -1

31 | empty | 0 | -1

32 | empty | 0 | -1

33 | empty | 0 | -1

34 | empty | 0 | -1

35 | empty | 0 | -1

36 | empty | 0 | -1

37 | empty | 0 | -1

38 | empty | 0 | -1

39 | empty | 0 | -1

40 | empty | 0 | -1

41 | empty | 0 | -1

42 | Cook | 1 | 42

43 | Casper | 1 | 43

44 | Red | 3 | 42

45 | Ajose | 4 | 42

46 | Arauza | 4 | 43

47 | Buck | 6 | 42

48 | Clark | 7 | 42

49 | Crouch | 7 | 43

50 | Davies | 8 | 43

51 | Dugger | 9 | 43

52 | Egbe | 11 | 42

53 | Farral | 11 | 43

54 | Garza | 13 | 42

55 | Gurung | 13 | 43

56 | Joseph | 14 | 43

57 | Kelly | 16 | 42

58 | Corey | 17 | 42

59 | Adam | 18 | 42

60 | Dustin | 18 | 43

61 | Robert | 19 | 43

62 | Kyle | 21 | 42

63 | Scott | 22 | 42

64 | Alcantara | 1 | 64

65 | Carmona | 2 | 64

66 | 1234567890123456 | 1 | 66

67 | Aguirrie | 1 | 67

68 | Paschal | 1 | 68

69 | Zulfiqar | 6 | 64

70 | Bhandari | 1 | 70

71 | Daniels | 2 | 70

72 | Nienberg | 3 | 70

73 | Salkowski | 4 | 70

74 | Qamruddin | 10 | 65

75 | Acevedo | 7 | 69

76 | Ellington | 6 | 71

77 | Clayton | 9 | 69

78 | Octavio | 10 | 69

79 | Judy | 38 | 42

80 | Derek | 39 | 42

81 | Jeffrey | 10 | 72

82 | Jordon | 40 | 43

83 | Vinnela | 20 | 64

84 | Lisa | 43 | 42

85 | Todd | 44 | 42

86 | Veronica | 22 | 65

87 | Matthew | 16 | 72

88 | Michael | 21 | 68

89 | Akhila | 47 | 43

90 | John | 49 | 42

91 | Charles | 22 | 70

92 | James | 51 | 42

93 | Chris | 52 | 42

94 | Wade | 53 | 42

95 | Fernando | 31 | 65

96 | Batbold | 32 | 65

97 | ABCDEFGHIJKLMNOP | 1 | 97

98 | Joel | 57 | 42

99 | Fabulous | 29 | 71

100 | Maiden | 58 | 43

101 | Eye | 60 | 42

102 | Syncopate | 39 | 64

103 | Yolk | 62 | 42

104 | Person | 62 | 43

105 | Northwest | 40 | 66

106 | Fabricate | 43 | 64

107 | Honor | 66 | 42

108 | Staple | 66 | 43

109 | Under | 68 | 42

110 | Jutty | 69 | 42

111 | Finagle | 46 | 66

112 | Cook | 71 | 42

113 | Rush | 72 | 42

114 | Wine | 73 | 42

115 | Screen | 73 | 43

116 | Perfect | 46 | 71

117 | mole | 76 | 42

118 | poison | 76 | 43

119 | brutalize | 53 | 67

120 | cap | 79 | 42

121 | cauldron | 53 | 69

122 | wince | 81 | 42

123 | live | 82 | 42

124 | magnetic | 58 | 67

125 | inlet | 84 | 42

126 | constrain | 63 | 64

127 | marsupial | 61 | 67

First 25 keys:

Minimum probes is: 1

Maximum probes is: 11

Average probes is: 4.28

Last 25 keys:

Minimum probes is: 1

Maximum probes is: 104

Average probes is: 61.76

Total Average:34.5

Theoretical Probes:

Theoretical Probes: 4.071428571428571

finalHash MyHash Linear

Index Word Probes Original Hash

0 | orthodontist | 8 | 121

1 | toluene | 14 | 116

2 | Matthew | 1 | 2

3 | quandary | 7 | 125

4 | empty | 0 | -1

5 | empty | 0 | -1

6 | wince | 1 | 6

7 | Christopher | 1 | 7

8 | Salkowski | 1 | 8

9 | magnetic | 2 | 8

10 | marsupial | 1 | 10

11 | Eye | 1 | 11

12 | sponsor | 5 | 8

13 | Lisa | 1 | 13

14 | Charles | 1 | 14

15 | Necromancer | 3 | 13

16 | Ellington | 1 | 16

17 | Screen | 2 | 16

18 | Corey | 1 | 18

19 | Clayton | 2 | 18

20 | sparse | 3 | 18

21 | empty | 0 | -1

22 | empty | 0 | -1

23 | Alcantara | 1 | 23

24 | James | 1 | 24

25 | Kelly | 1 | 25

26 | brutalize | 1 | 26

27 | Afterwards | 1 | 27

28 | Batbold | 1 | 28

29 | rationalize | 4 | 26

30 | Honor | 1 | 30

31 | John | 1 | 31

32 | parasympathetic | 2 | 31

33 | Qamruddin | 1 | 33

34 | Misogamist | 2 | 33

35 | Todd | 1 | 35

36 | cap | 5 | 32

37 | Scott | 1 | 37

38 | scat | 12 | 27

39 | wet | 12 | 28

40 | Crouch | 1 | 40

41 | nosegay | 8 | 34

42 | forgetful | 17 | 26

43 | Veronica | 1 | 43

44 | privilege | 22 | 23

45 | exhilarate | 19 | 27

46 | Robert | 1 | 46

47 | empty | 0 | -1

48 | live | 1 | 48

49 | empty | 0 | -1

50 | empty | 0 | -1

51 | empty | 0 | -1

52 | empty | 0 | -1

53 | Bhandari | 1 | 53

54 | Zulfiqar | 1 | 54

55 | Michael | 1 | 55

56 | Daniels | 1 | 56

57 | empty | 0 | -1

58 | empty | 0 | -1

59 | empty | 0 | -1

60 | Judy | 1 | 60

61 | Carmona | 1 | 61

62 | Octavio | 2 | 61

63 | Egbe | 1 | 63

64 | Joel | 3 | 62

65 | Kyle | 1 | 65

66 | Northwest | 1 | 66

67 | Adam | 1 | 67

68 | Yolk | 2 | 67

69 | Joseph | 1 | 69

70 | Akhila | 1 | 70

71 | Irreversible | 2 | 70

72 | Fabricate | 12 | 61

73 | Chris | 1 | 73

74 | Finagle | 5 | 70

75 | Person | 1 | 75

76 | Farral | 1 | 76

77 | Jutty | 2 | 76

78 | Dugger | 1 | 78

79 | Syncopate | 2 | 78

80 | poison | 9 | 72

81 | Buck | 1 | 81

82 | cauldron | 15 | 68

83 | inlet | 12 | 72

84 | constrain | 20 | 65

85 | Fabulous | 1 | 85

86 | Ajose | 1 | 86

87 | Jeffrey | 1 | 87

88 | Jordon | 1 | 88

89 | empty | 0 | -1

90 | empty | 0 | -1

91 | empty | 0 | -1

92 | Aguirrie | 1 | 92

93 | Gurung | 2 | 92

94 | Wade | 2 | 93

95 | empty | 0 | -1

96 | empty | 0 | -1

97 | Fernando | 1 | 97

98 | Casper | 1 | 98

99 | Maiden | 2 | 98

100 | Cook | 1 | 100

101 | Vinnela | 1 | 101

102 | Under | 5 | 98

103 | Acevedo | 1 | 103

104 | ABCDEFGHIJKLMNOP | 1 | 104

105 | Staple | 1 | 105

106 | Cook | 7 | 100

107 | Red | 1 | 107

108 | Constriction | 2 | 107

109 | mole | 13 | 97

110 | option | 14 | 97

111 | Derek | 1 | 111

112 | Davies | 1 | 112

113 | dactyl | 1 | 113

114 | empty | 0 | -1

115 | guard | 1 | 115

116 | Arauza | 1 | 116

117 | Garza | 1 | 117

118 | Nienberg | 1 | 118

119 | Dustin | 1 | 119

120 | Paschal | 1 | 120

121 | Clark | 1 | 121

122 | 1234567890123456 | 1 | 122

123 | Wine | 5 | 119

124 | ratiocination | 1 | 124

125 | Perfect | 1 | 125

126 | Rush | 1 | 126

127 | prepossess | 1 | 127

First 25 keys:

Minimum probes is: 1

Maximum probes is: 1

Average probes is: 1.0

Last 25 keys:

Minimum probes is: 1

Maximum probes is: 22

Average probes is: 8.24

Total Average:3.2454545454545456

Theoretical Probes:

Theoretical Probes: 4.071428571428571

finalHash Burris Random

Index Word Probes Original Hash

0 | inlet | 51 | 42

1 | marsupial | 11 | 67

2 | Syncopate | 7 | 64

3 | Kelly | 10 | 42

4 | Dustin | 10 | 43

5 | Necromancer | 1 | 5

6 | nosegay | 11 | 72

7 | empty | 0 | -1

8 | Irreversible | 1 | 8

9 | empty | 0 | -1

10 | Christopher | 1 | 10

11 | Fabricate | 8 | 64

12 | Fernando | 8 | 65

13 | Judy | 16 | 42

14 | Person | 16 | 43

15 | Kyle | 14 | 42

16 | Maiden | 14 | 43

17 | Yolk | 28 | 42

18 | Wade | 25 | 42

19 | Afterwards | 1 | 19

20 | Under | 31 | 42

21 | option | 31 | 43

22 | wet | 53 | 42

23 | Wine | 38 | 42

24 | Misogamist | 1 | 24

25 | parasympathetic | 2 | 24

26 | Derek | 17 | 42

27 | Staple | 17 | 43

28 | magnetic | 10 | 67

29 | ratiocination | 3 | 23

30 | cap | 45 | 42

31 | prepossess | 3 | 25

32 | Todd | 19 | 42

33 | Screen | 19 | 43

34 | Garza | 9 | 42

35 | Joseph | 9 | 43

36 | Scott | 15 | 42

37 | Chris | 24 | 42

38 | Rush | 37 | 42

39 | rationalize | 6 | 26

40 | Constriction | 5 | 12

41 | Cook | 36 | 42

42 | Cook | 1 | 42

43 | Casper | 1 | 43

44 | Arauza | 2 | 43

45 | orthodontist | 4 | 14

46 | Honor | 29 | 42

47 | dactyl | 29 | 43

48 | Red | 3 | 42

49 | Crouch | 3 | 43

50 | empty | 0 | -1

51 | Joel | 26 | 42

52 | sparse | 26 | 43

53 | empty | 0 | -1

54 | James | 21 | 42

55 | Buck | 6 | 42

56 | Dugger | 6 | 43

57 | Batbold | 9 | 65

58 | toluene | 9 | 66

59 | live | 50 | 42

60 | privilege | 15 | 66

61 | Jutty | 32 | 42

62 | Adam | 13 | 42

63 | Akhila | 13 | 43

64 | Alcantara | 1 | 64

65 | Carmona | 2 | 64

66 | 1234567890123456 | 1 | 66

67 | Aguirrie | 1 | 67

68 | Paschal | 1 | 68

69 | Acevedo | 1 | 69

70 | Bhandari | 1 | 70

71 | Daniels | 2 | 70

72 | Ellington | 2 | 71

73 | Ajose | 4 | 42

74 | Davies | 4 | 43

75 | Clayton | 3 | 69

76 | Nienberg | 3 | 70

77 | Fabulous | 3 | 71

78 | Jeffrey | 3 | 72

79 | Finagle | 6 | 66

80 | brutalize | 6 | 67

81 | empty | 0 | -1

82 | cauldron | 6 | 69

83 | Charles | 6 | 70

84 | empty | 0 | -1

85 | mole | 40 | 42

86 | forgetful | 13 | 66

87 | empty | 0 | -1

88 | Eye | 27 | 42

89 | scat | 52 | 42

90 | sponsor | 13 | 70

91 | Lisa | 18 | 42

92 | Vinnela | 5 | 64

93 | Veronica | 5 | 65

94 | Northwest | 5 | 66

95 | Zulfiqar | 4 | 64

96 | Qamruddin | 4 | 65

97 | ABCDEFGHIJKLMNOP | 1 | 97

98 | Jordon | 12 | 43

99 | Michael | 4 | 68

100 | Octavio | 4 | 69

101 | Salkowski | 4 | 70

102 | Perfect | 4 | 71

103 | Matthew | 4 | 72

104 | Corey | 11 | 42

105 | Robert | 11 | 43

106 | empty | 0 | -1

107 | empty | 0 | -1

108 | Clark | 7 | 42

109 | Farral | 7 | 43

110 | empty | 0 | -1

111 | wince | 46 | 42

112 | exhilarate | 14 | 11

113 | empty | 0 | -1

114 | guard | 57 | 42

115 | empty | 0 | -1

116 | empty | 0 | -1

117 | Egbe | 8 | 42

118 | Gurung | 8 | 43

119 | empty | 0 | -1

120 | empty | 0 | -1

121 | John | 20 | 42

122 | poison | 20 | 43

123 | empty | 0 | -1

124 | empty | 0 | -1

125 | quandary | 12 | 70

126 | constrain | 11 | 64

127 | empty | 0 | -1

First 25 keys:

Minimum probes is: 1

Maximum probes is: 8

Average probes is: 2.92

Last 25 keys:

Minimum probes is: 3

Maximum probes is: 57

Average probes is: 23.24

Total Average:13.345454545454546

Theoretical Probes:

Theoretical Probes: 4.071428571428571

finalHash MyHash Random

Index Word Probes Original Hash

0 | mole | 4 | 97

1 | Under | 4 | 98

2 | Matthew | 1 | 2

3 | Perfect | 3 | 125

4 | empty | 0 | -1

5 | empty | 0 | -1

6 | wince | 1 | 6

7 | Christopher | 1 | 7

8 | Salkowski | 1 | 8

9 | magnetic | 2 | 8

10 | inlet | 7 | 72

11 | Eye | 1 | 11

12 | empty | 0 | -1

13 | Lisa | 1 | 13

14 | Charles | 1 | 14

15 | empty | 0 | -1

16 | Ellington | 1 | 16

17 | Screen | 2 | 16

18 | Corey | 1 | 18

19 | Clayton | 2 | 18

20 | empty | 0 | -1

21 | orthodontist | 5 | 121

22 | empty | 0 | -1

23 | Alcantara | 1 | 23

24 | James | 1 | 24

25 | Kelly | 1 | 25

26 | brutalize | 1 | 26

27 | Afterwards | 1 | 27

28 | Batbold | 1 | 28

29 | wet | 2 | 28

30 | Honor | 1 | 30

31 | John | 1 | 31

32 | parasympathetic | 2 | 31

33 | Qamruddin | 1 | 33

34 | Misogamist | 2 | 33

35 | Todd | 1 | 35

36 | sponsor | 5 | 8

37 | Scott | 1 | 37

38 | cap | 3 | 32

39 | forgetful | 6 | 26

40 | Crouch | 1 | 40

41 | marsupial | 4 | 10

42 | empty | 0 | -1

43 | Veronica | 1 | 43

44 | Necromancer | 4 | 13

45 | empty | 0 | -1

46 | Robert | 1 | 46

47 | nosegay | 6 | 34

48 | live | 1 | 48

49 | sparse | 4 | 18

50 | toluene | 11 | 116

51 | privilege | 5 | 23

52 | empty | 0 | -1

53 | Bhandari | 1 | 53

54 | Zulfiqar | 1 | 54

55 | Michael | 1 | 55

56 | Daniels | 1 | 56

57 | rationalize | 4 | 26

58 | scat | 4 | 27

59 | quandary | 11 | 125

60 | Judy | 1 | 60

61 | Carmona | 1 | 61

62 | Octavio | 2 | 61

63 | Egbe | 1 | 63

64 | empty | 0 | -1

65 | Kyle | 1 | 65

66 | Northwest | 1 | 66

67 | Adam | 1 | 67

68 | Joel | 3 | 62

69 | Joseph | 1 | 69

70 | Akhila | 1 | 70

71 | Irreversible | 2 | 70

72 | poison | 1 | 72

73 | Chris | 1 | 73

74 | cauldron | 3 | 68

75 | Person | 1 | 75

76 | Farral | 1 | 76

77 | Jutty | 2 | 76

78 | Dugger | 1 | 78

79 | Syncopate | 2 | 78

80 | empty | 0 | -1

81 | Buck | 1 | 81

82 | empty | 0 | -1

83 | Finagle | 6 | 70

84 | empty | 0 | -1

85 | Fabulous | 1 | 85

86 | Ajose | 1 | 86

87 | Jeffrey | 1 | 87

88 | Jordon | 1 | 88

89 | Fabricate | 5 | 61

90 | empty | 0 | -1

91 | empty | 0 | -1

92 | Aguirrie | 1 | 92

93 | Gurung | 2 | 92

94 | Wade | 2 | 93

95 | Yolk | 5 | 67

96 | constrain | 4 | 65

97 | Fernando | 1 | 97

98 | Casper | 1 | 98

99 | Maiden | 2 | 98

100 | Cook | 1 | 100

101 | Vinnela | 1 | 101

102 | exhilarate | 8 | 27

103 | Acevedo | 1 | 103

104 | ABCDEFGHIJKLMNOP | 1 | 104

105 | Staple | 1 | 105

106 | Cook | 3 | 100

107 | Red | 1 | 107

108 | Constriction | 2 | 107

109 | empty | 0 | -1

110 | option | 6 | 97

111 | Derek | 1 | 111

112 | Davies | 1 | 112

113 | dactyl | 1 | 113

114 | empty | 0 | -1

115 | guard | 1 | 115

116 | Arauza | 1 | 116

117 | Garza | 1 | 117

118 | Nienberg | 1 | 118

119 | Dustin | 1 | 119

120 | Paschal | 1 | 120

121 | Clark | 1 | 121

122 | 1234567890123456 | 1 | 122

123 | empty | 0 | -1

124 | ratiocination | 1 | 124

125 | Wine | 3 | 119

126 | Rush | 1 | 126

127 | prepossess | 1 | 127

First 25 keys:

Minimum probes is: 1

Maximum probes is: 1

Average probes is: 1.0

Last 25 keys:

Minimum probes is: 1

Maximum probes is: 11

Average probes is: 4.24

Total Average:2.118181818181818

Theoretical Probes:

Theoretical Probes: 4.071428571428571