Home Work Assignment 3

Thursday, October 3, 2019 11:41 AM

Name : VIKAS VEERABATHINI
NU ID : 001302155

Problem 1 :

Key Value	Probe Sequence
43	0
23	0
1	0
0	0
15	0
31	0
4	0
7	0
11	0,1,2,**3**
3	0,1,2,**3**
5	0,1,2,3,**4**
9	0,1,2,3,4,5,6,7,8,9,10 ***no insert (discard , error hash table overflow)***

	Final Hash Table Contents
0	43
1	0
2	31
3	1
4	5
5	7
6	23
7	15
8	11
9	4
10	3

Problem 2 :

Please find the output logs of solution as follows :

Size of Hash Table is : 64
Mean of Hash Table is : 15
Minimum of Hash Table is : 7

Maximum of Hash Table is : 34
Variance of Hash Table is : 46.6562

Size of Hash Table is : 66
Mean of Hash Table is : 15
Minimum of Hash Table is : 2
Maximum of Hash Table is : 29
Variance of Hash Table is : 31.0909

Size of Hash Table is : 67
Mean of Hash Table is : 14
Minimum of Hash Table is : 5
Maximum of Hash Table is : 28
Variance of Hash Table is : 27.5821

Size of Hash Table is : 61
Mean of Hash Table is : 16
Minimum of Hash Table is : 8
Maximum of Hash Table is : 26
Variance of Hash Table is : 15.1475

Size of Hash Table is : 59
Mean of Hash Table is : 16
Minimum of Hash Table is : 10
Maximum of Hash Table is : 25
Variance of Hash Table is : 14.1356

Size of Hash Table is : 54
Mean of Hash Table is : 18
Minimum of Hash Table is : 9
Maximum of Hash Table is : 35
Variance of Hash Table is : 32.2222

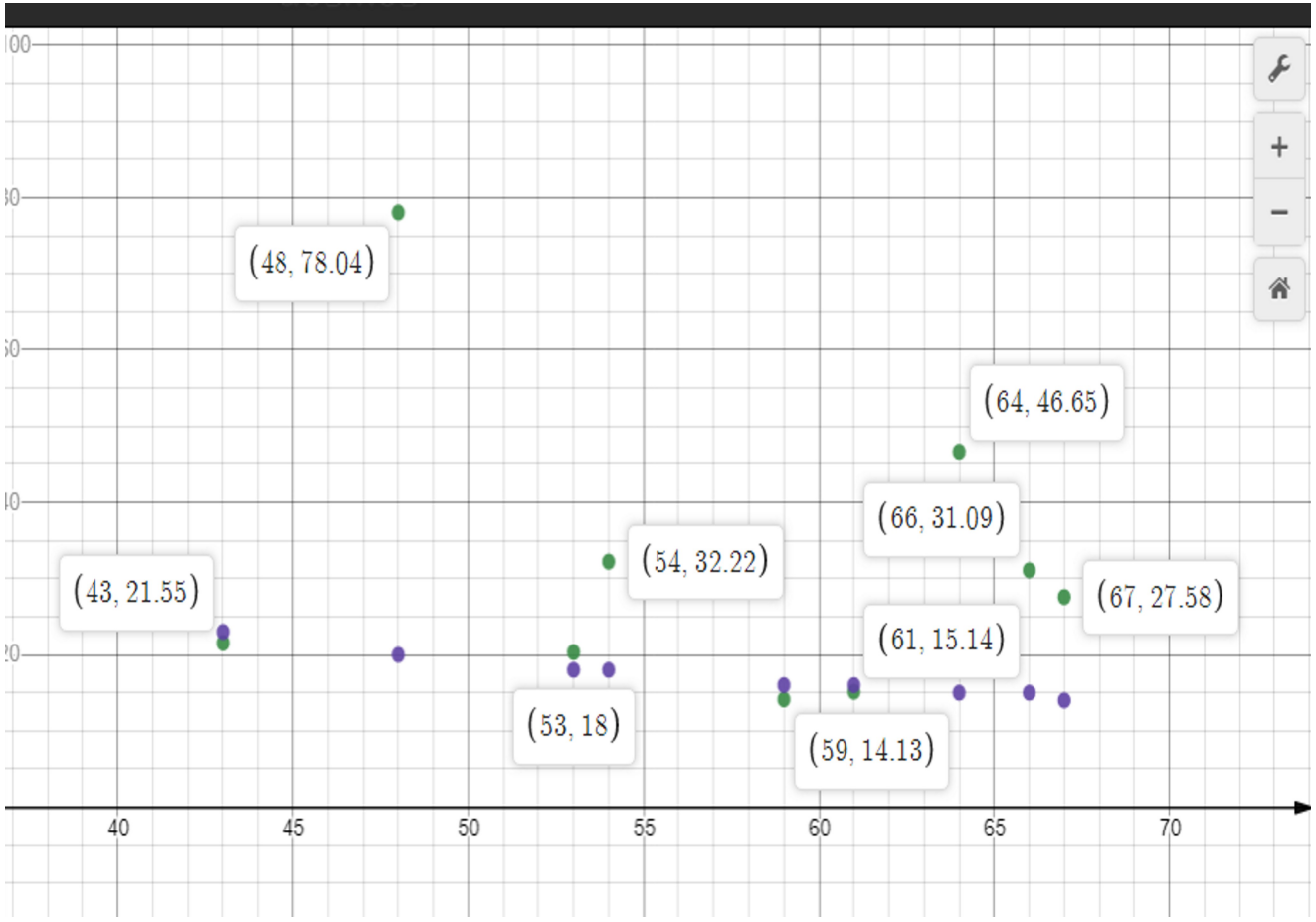
Size of Hash Table is : 53
Mean of Hash Table is : 18
Minimum of Hash Table is : 9
Maximum of Hash Table is : 29
Variance of Hash Table is : 20.3396

Size of Hash Table is : 48
Mean of Hash Table is : 20
Minimum of Hash Table is : 11
Maximum of Hash Table is : 45
Variance of Hash Table is : 78.0417

Size of Hash Table is : 43
Mean of Hash Table is : 23
Minimum of Hash Table is : 14
Maximum of Hash Table is : 33

Variance of Hash Table is : 21.5581 .

Below is a point graph , green dots indicate variance of the particular size on Y-axis .
Blue dots indicate mean on Y-axis .



Graph plot is to convey that there is a high separation between variance and mean for composite numbers . While it is less for prime numbers and minimum when they are away from numbers that are power of 2's . This indicates that having a prime number as a hash table size leads to uniform distribution of numbers across hash table . On the contrary with composite numbers , numbers don't have an uniform distribution across hash table slots .

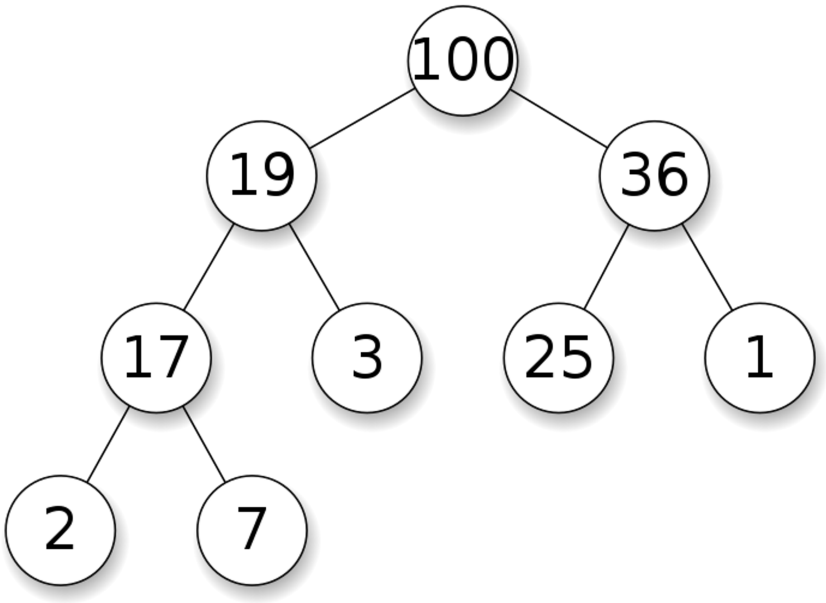
One more interesting point is that there is a highly non uniform data in case of 64 which is a power of 2 . As we know that , 64 = 0b1000000 having a modulus function with 64 as moduli value all the numbers seventh bit would taken

into account , thereby having the hash table left with very less options .

Having an uniformly distributed hash table , would lead to search algorithm to run in an average time rather than in worst running time in some cases , where hash table slot is more dense .

Problem 3 :

Lets consider below heap tree :



A corresponding hash-table if we represent the same using arrays , it would be something like below in a space efficient way :

Value	100	19	36	17	3	25	1	2	7
Index	1	2	3	4	5	6	7	8	9

From the same , the first (or last) element will contain the root. The next two elements of the array contain its children. The next four contain the four children of the two child nodes, etc.

Thus , if we represent them mathematically ,
For any parent with index ***i*** ,
Child 1 would be = ***2i ….. (1)***
Child 2 would be = ***2i+1 ….. (2)***

For a node at index ***j*** , let index of it's parent be ***z .***
Thus , we would have .
j = 2z or 2z + 1 …. (3)

Thereby ,
Z = [j/2] ….. (4) .. [Dividing by 2 on both sides of equation 1] .

Equations 1 , 2 and 4 help to affirm the heap indices calculations as asked in question