Wittenberg UNIVERSITY

Course Title: Sequential and parallel Algorithm
Course Number & Section: COMP 275-01
Activity 10b
(Reb-black, 2-3 BST Tree and Hash Table)

Spring 2024

April 22 -29

Name: Thomas Weis

ID # : 572556

Exercise # 1: x +y, kx+y rules, XOR-bitwise and Java-Object Hash functions

a. This exercise aims to confirm the effectiveness of the kx + y rule over the x+y rule in implementing a modulo hashing algorithm(function) where m= $\overset{34\ 69}{\{\cancel{8},\cancel{16}\}}$, and k=6n+1=prime number. Hence, implement a Java program that will use these rules to generate a hash code for a set of $^{L}P_{r}$ letters, where L= {letters: A, B, C, D} and r = 2. Assuming the letters ASCII code are to be used accordingly, and each letter pair is equivalent to {x, y}, complete the table below.

Sn	$^{L}P_{r}$	*x + y rule* Hash code m=34	*x + y rule* Hash code m=69	*kx + y rule* Hash code m=34			*kx + y rule* Hash code m=69			xor Hash code (x+y)	xor Hash code (x*y)	Object Hash code (x+y)	Object Hash code (x*y)
	xy			k=7	k=19	k=31	k=7	k=19	k=31				
1.	AB	29	62	11	9	7	38	59	11	163	5362	162	4321
2.	AC	30	63	12	10	8	39	60	12	165	5443	163	4336
3.	AD	31	64	13	11	9	40	61	13	164	5397	164	4451
4.	BC	31	64	19	29	5	46	10	43	164	5399	164	4453
5.	BD	32	65	20	30	6	47	11	44	167	5610	165	4519
6.	CD	33	66	27	15	3	54	30	6	166	5567	166	4587
7.	BA	29	62	17	27	3	44	8	41	163	5362	162	4321
8.	CA	30	63	24	12	0	51	27	3	165	5443	163	4386
9.	DA	31	64	31	31	31	58	46	34	164	5397	164	4451
10.	CB	31	64	25	13	1	52	28	4	164	5399	164	4463
11.	DB	32	65	20	30	6	47	11	44	167	5610	165	4519
12.	DC	33	66	33	33	33	60	48	36	166	5567	166	4587

naive XY hash mod

b. Write your observation concerning the different methods in terms of effectiveness.

The x+y rule (34 and 69), XOR x+y hashcode, and Object x+y hash-codes all generate duplicates + an not effective hashes.

kx+y rule (all k and m values), XOR x·y hashcode, and Object x·y hash-codes all do not generate duplicates + are the effective methods.

Exercise # 2:

a. This exercise wants you to leverage the hash code methods mentioned in Q1a to create an index for storing values in an array of n size. Assume the data to be stored are sets of individual names and ages with their names as keys. Write a Java program for this and complete the table below.

Sn	Name	$x + y$ rule Hash code m=34	$x + y$ rule Hash code m=69	$kx + y$ rule Hash code m=34			$kx + y$ rule Hash code m=69		
				k=7	k=19	k=31	k=7	k=19	k=31
1.	John 32	15	47	17	21	25	8	68	59
2.	Matthew 18	4	36	24	30	2	15	42	0
3.	Luke 56	23	55	3	31	25	28	43	58
4.	Jerry 29	5	37	7	11	15	67	58	49
5.	Tom 21	25	57	19	7	29	9	51	24
6.	Kerry 45	6	38	14	30	12	5	8	11
7.	Linus 36	11	43	25	19	13	16	31	46
8.	Caleb 52	28	26	22	10	32	14	59	35
9.	Adam 27	29	27	11	9	7	3	24	45
10.	Philps 30	14	46	18	26	0	43	37	31

from Moodle

b. Write your observations about the hash code in terms of collision.

There were, in fact, no collisions occuring and all values created were unique. This might have to do with the veriaty of characters and string lengths in the origenal table to hash.

c. Using one of the separate chaining or linear probing techniques rewrite your code to improve its capabilities in handling collisions.