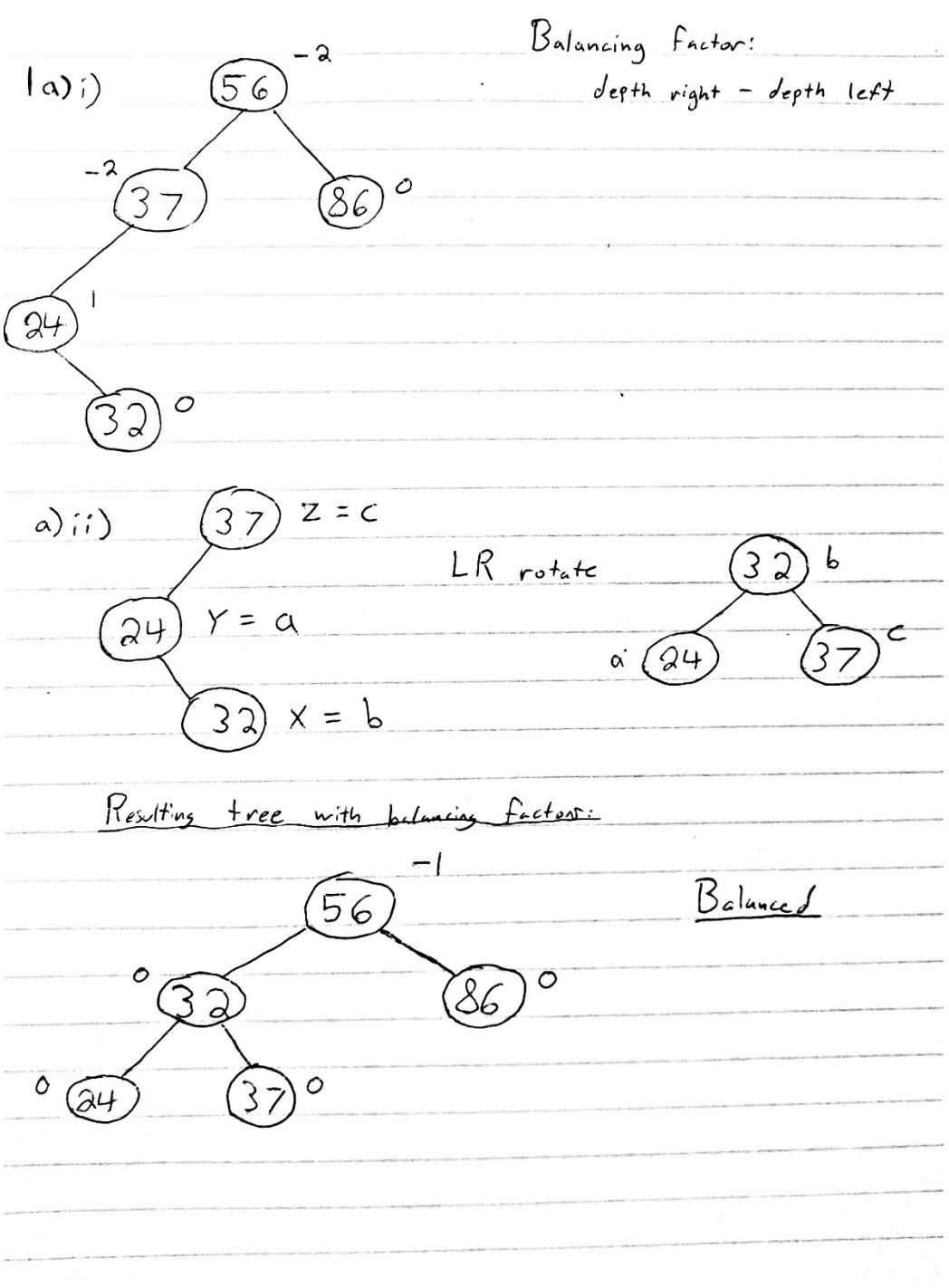
Oliver Benning
7798804
CSI 2110 Fall 2017
Dr. Lucia Moura
Submitted Dec 11, 2017

Assignment 5

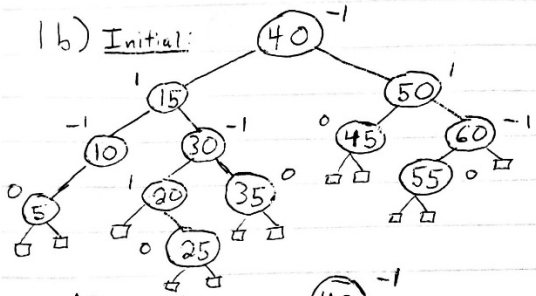
1. a)



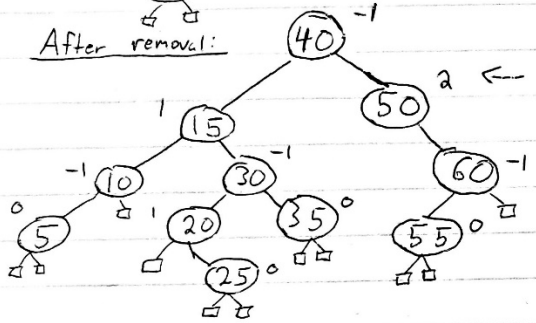
1 b)

Balancing factor: depth right − depth left

1b) Initial:

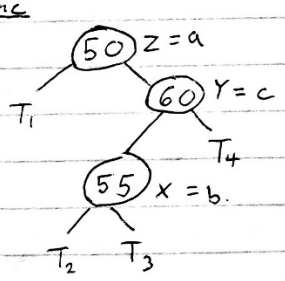


After removal:

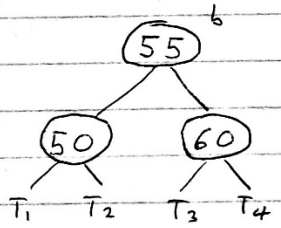


2 ← deepest unbalanced node

Rebalance

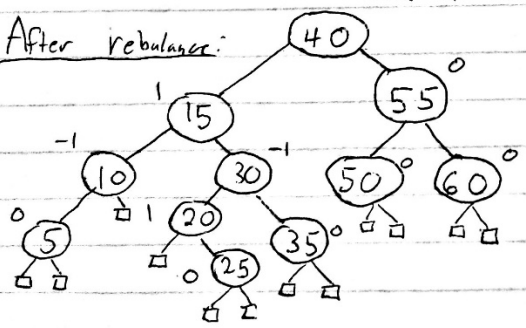


RL rotate:

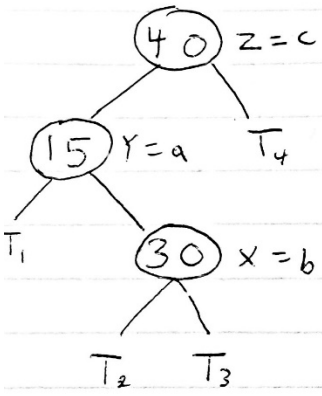


-2 ← deepest unbalanced node

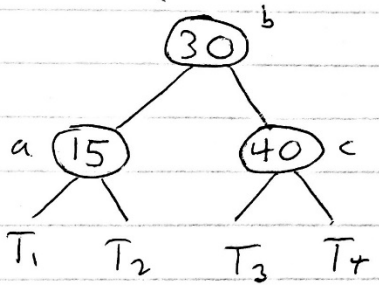
After rebalance:



1b) cont... Rebalance

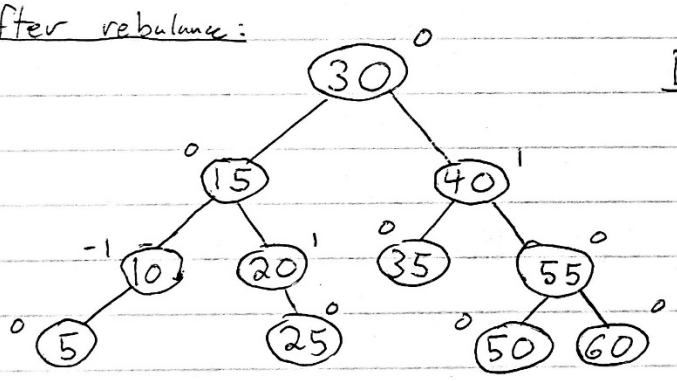


LR rotate:

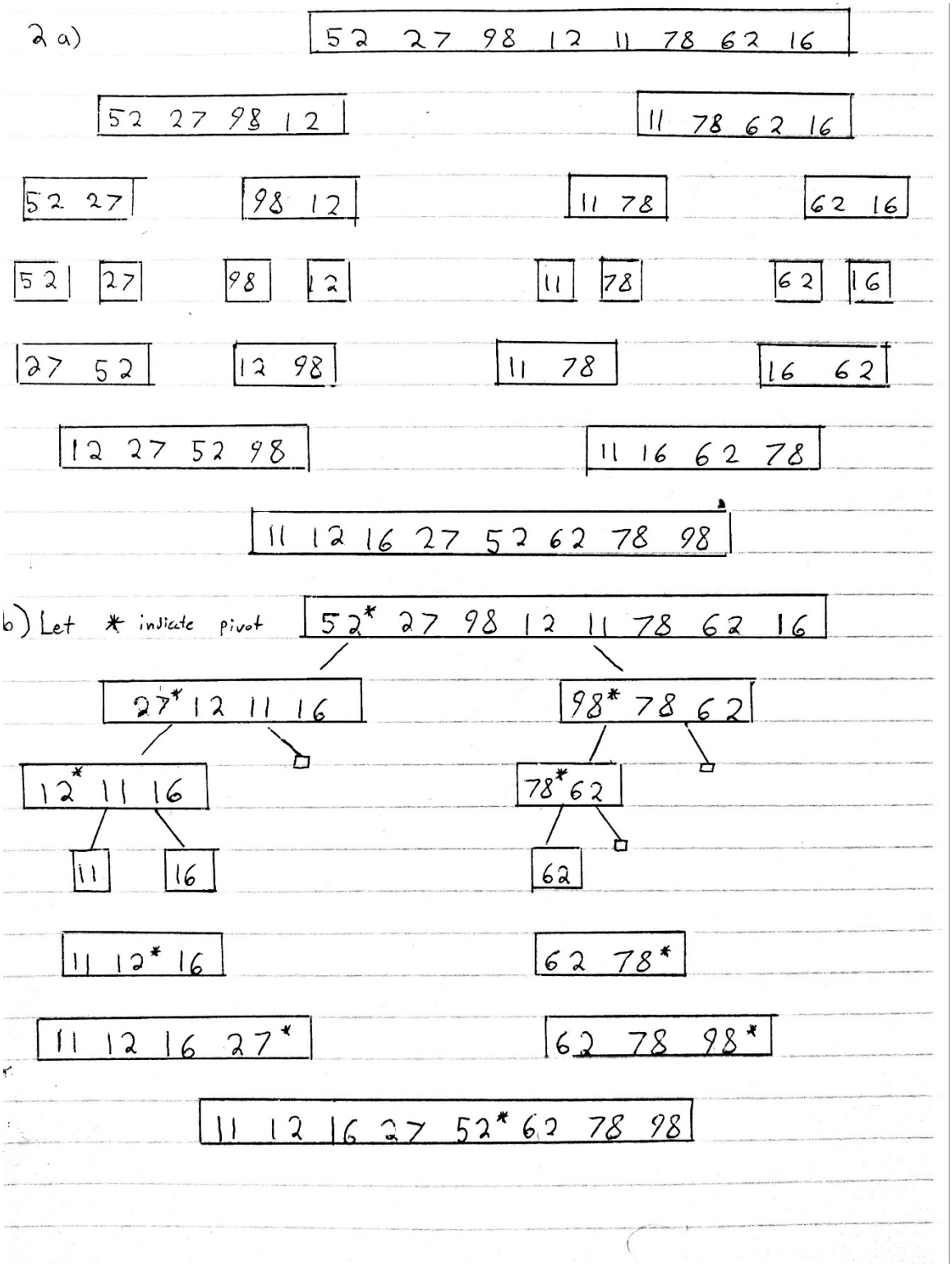


After rebalance:

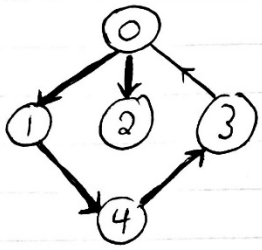
Balanced



2)

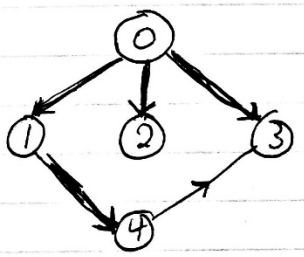

2 a)
52 27 98 12 11 78 62 16
52 27 98 12
11 78 62 16
52 27
98 12
11 78
62 16
52
27
98
12
11
78
62
16
27 52
12 98
11 78
16 62
12 27 52 98
11 16 62 78
11 12 16 27 52 62 78 98
b) Let * indicate pivot
52* 27 98 12 11 78 62 16
27* 12 11 16
98* 78 62
12* 11 16
78* 62
11
16
62
11 12* 16
62 78*
11 12 16 27*
62 78 98*
11 12 16 27 52* 62 78 98

3 a)



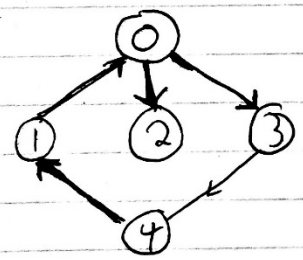
Node	Discovery edge	Back edge
0	–	–
1	(0,1)	–
4	(1,4)	–
3	(4,3)	(3,0)
2	(0,2)	–

b)



Node	Discovery edge	Cross edge
0	–	–
1	(0,1)	–
2	(0,2)	–
3	(0,3)	–
4	(1,4)	(4,3)

c)



Node	Discovery Edge	Back edge
4	–	–
1	(4,1)	–
0	(1,0)	–
2	(0,2)	–
3	(0,3)	(3,4)

4. I will execute Dijkstra's algorithm in the form of a table. Each row will represent the new analysis of a vertex with minimums listed, star represents the next vertex to analyze and dash means infinity.

Vertex	Jean	Paul	Sam	Kim	Emma	Joe	Jack	Jane	Ali	Bob	Luce	Wen	**Edge**
Jean	**0**	1*	-	-	-	1	-	2	-	2	2	-	
Paul	**0**	**1**	2	-	-	1*	-	2	-	2	2	-	(Jean, Paul)
Joe	**0**	**1**	2*	-	-	**1**	3	2	-	2	2	-	(Jean, Joe)
Sam	**0**	**1**	**2**	5	-	**1**	3	2*	-	2	2	-	(Paul, Sam)
Jane	**0**	**1**	**2**	5	-	**1**	3	**2**	5	2*	2	-	(Jean, Jane)
Bob	**0**	**1**	**2**	5	-	**1**	3	**2**	5	**2**	2*	-	(Jean, Bob)
Luce	**0**	**1**	**2**	5	-	**1**	3*	**2**	4	**2**	**2**	-	(Jean, Luce)
Jack	**0**	**1**	**2**	5	-	**1**	**3**	**2**	4*	**2**	**2**	-	(Joe, Jack)
Ali	**0**	**1**	**2**	5*	5	**1**	**3**	**2**	**4**	**2**	**2**	6	(Luce, Ali)
Kim	**0**	**1**	**2**	**5**	5*	**1**	**3**	**2**	**4**	**2**	**2**	6	(Sam, Kim)
Emma	**0**	**1**	**2**	**5**	**5**	**1**	**3**	**2**	**4**	**2**	**2**	6*	(Ali, Emma)
Wen	**0**	**1**	**2**	**5**	**5**	**1**	**3**	**2**	**4**	**2**	**2**	**6**	(Ali, Wen)

From this table we see immediately the total degree of separation separating Emma from Jean is 5. We follow the discovery edges in reverse which forming the path back to Jean. We then deduce the shortest path from Jean to Emma as [Jean, Luce, Ali, Emma] with a total degree of separation of 5.

5.

First we calculate h(k) for each value and insert it into the array. In the event of a collision we increment the value within the mod 17 field until an empty cell is reached.

Calculation and insertion:

k	h(k)	Inc	(h(k)+inc) mod 17
66	15	0	15
20	3	0	3
50	16	0	16
75	7	0	7
104	2	0	2
100	15	2	0
17	0	1	1
24	7	1	8
70	2	2	4
25	8	1	9

Compression map domain [0, 16]

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
100	17	104	20	70			75	24	25						66	50