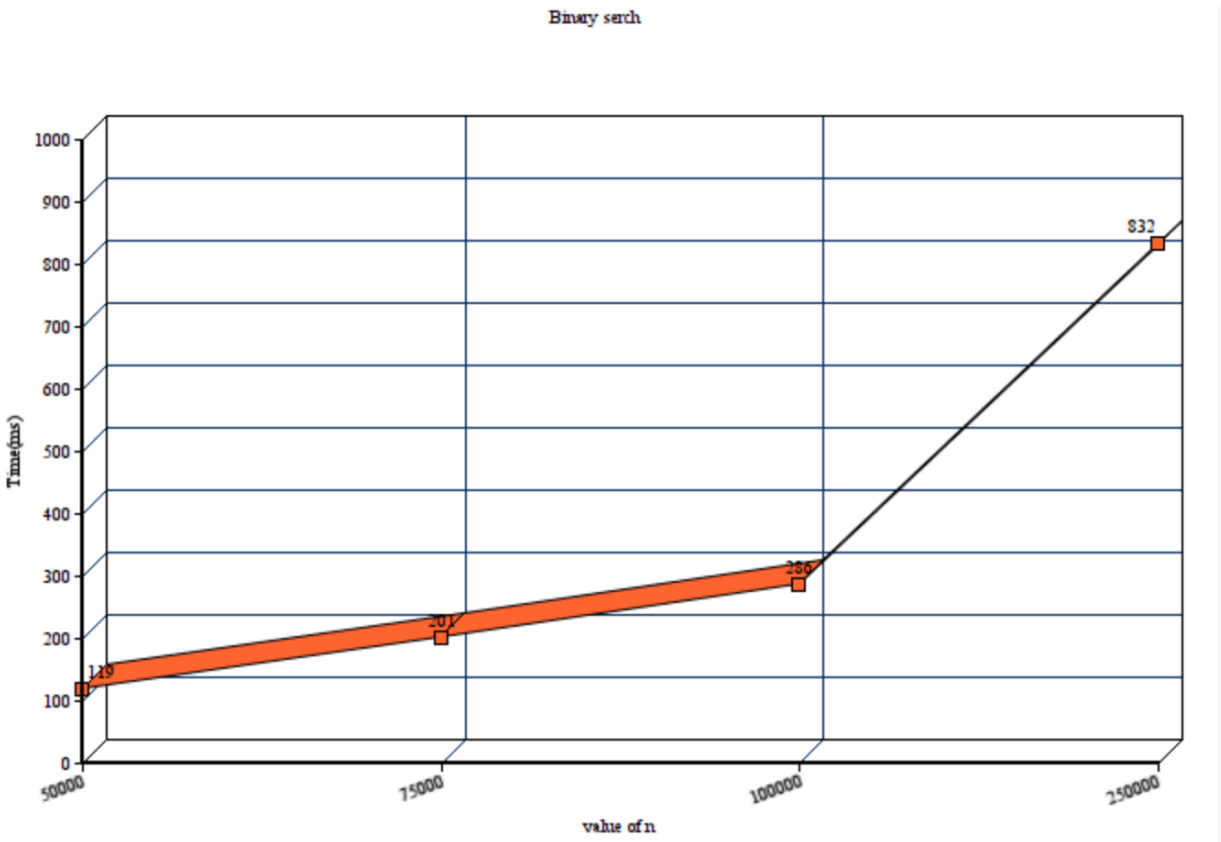
Homework :- 3

Prithiv Dev Devendran

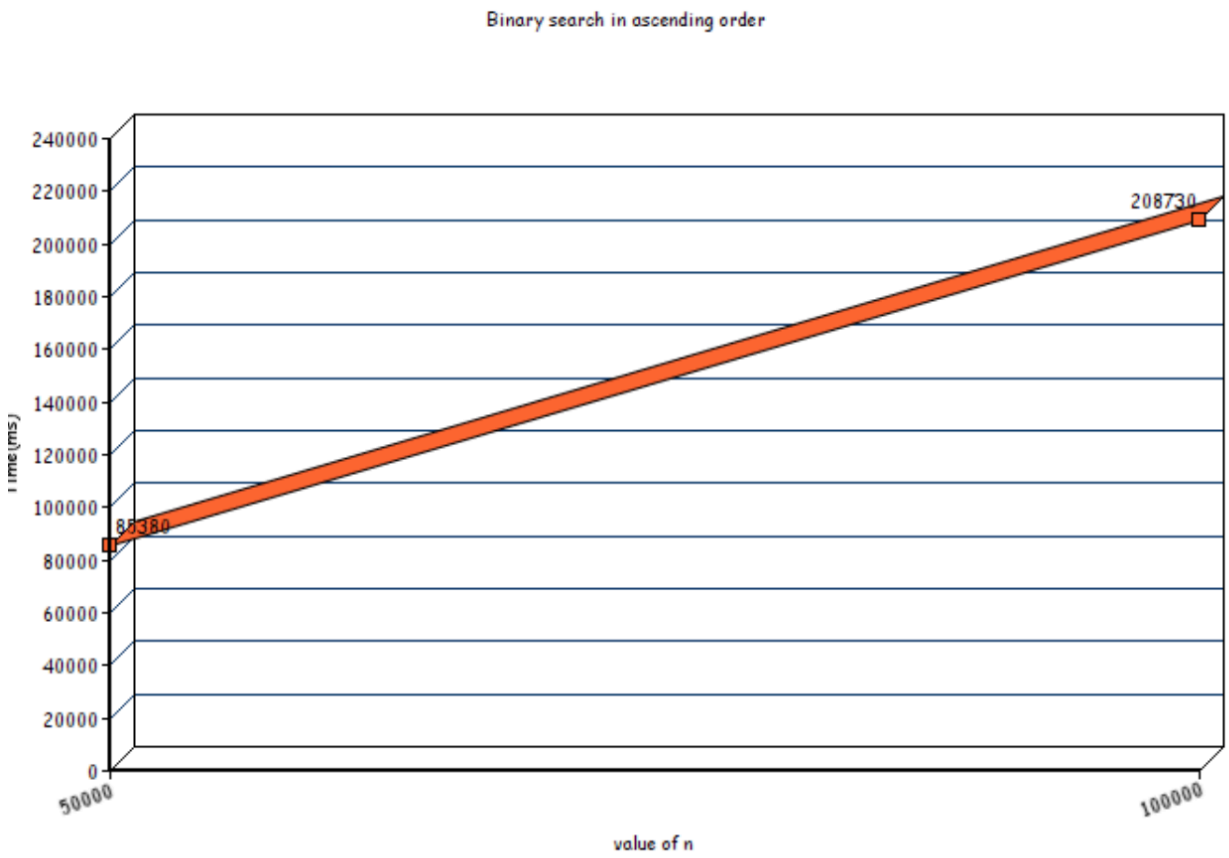
Readings of Binary Search Tree: -

Binary search Tree Input	Average Real	Average User	Average Sys	Average Height of Tree
50000 0 0	119ms	70ms	40ms	41
75000 0 0	201ms	140ms	50ms	37
100000 0 0	286ms	170ms	110ms	38
250000 0 0	832ms	530ms	300ms	44
50000 1 0	85380ms	84970ms	280ms	49999
50000 -1 0	76989ms	76590ms	280ms	49999
100000 1 0	208730ms	207872ms	620ms	99999
100000 -1 0	206117ms	205100ms	740ms	99999

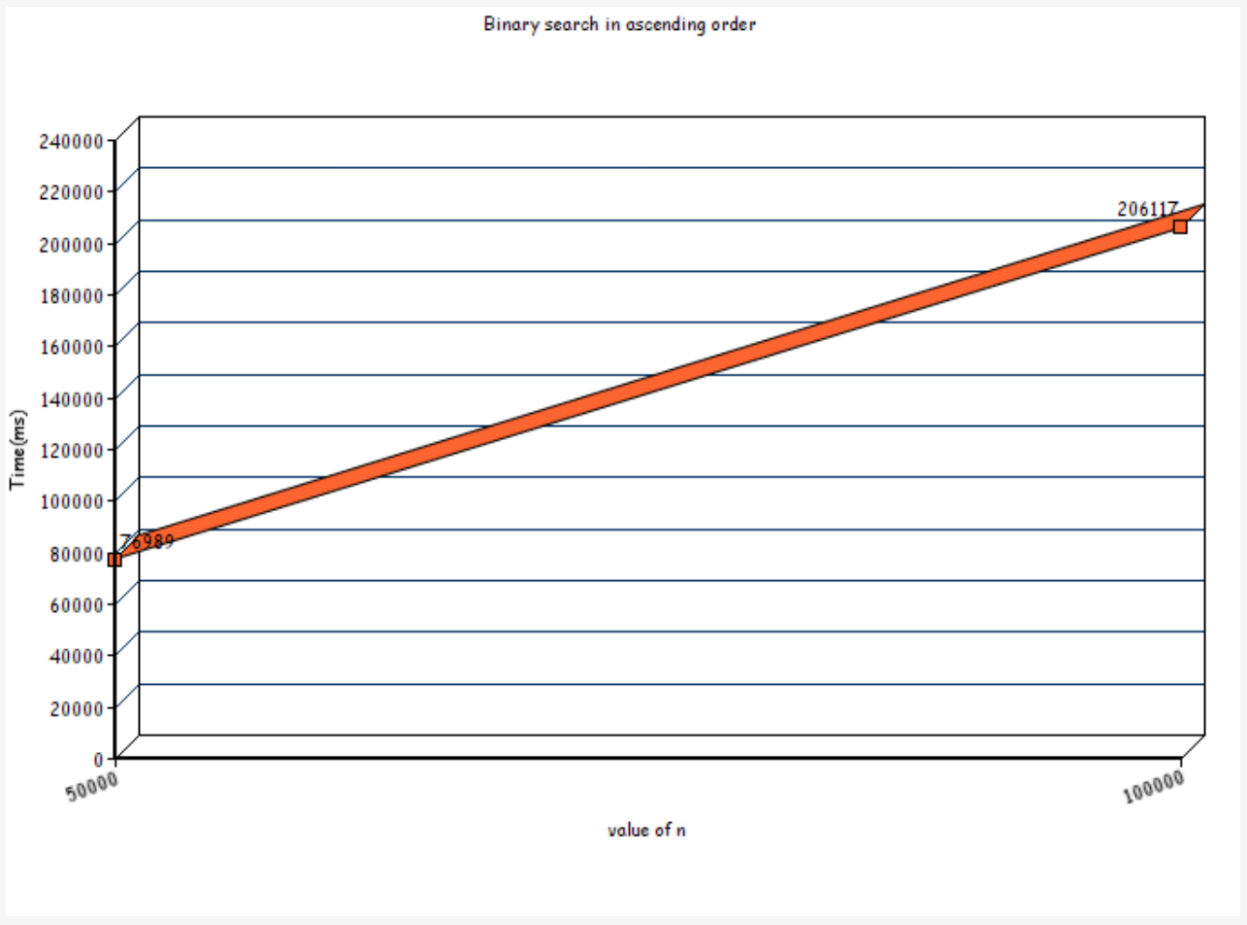
Homework :- 3


Binary serch
1000
900
800
700
600
500
400
300
200
100
0
Time(ms)
832
250
207
119
50000
75000
100000
250000
value of n

Homework :- 3


Binary search in ascending order
240000
220000
200000
180000
160000
140000
120000
100000
80000
60000
40000
20000
0
85380
208730
50000
100000
value of n

Homework :- 3

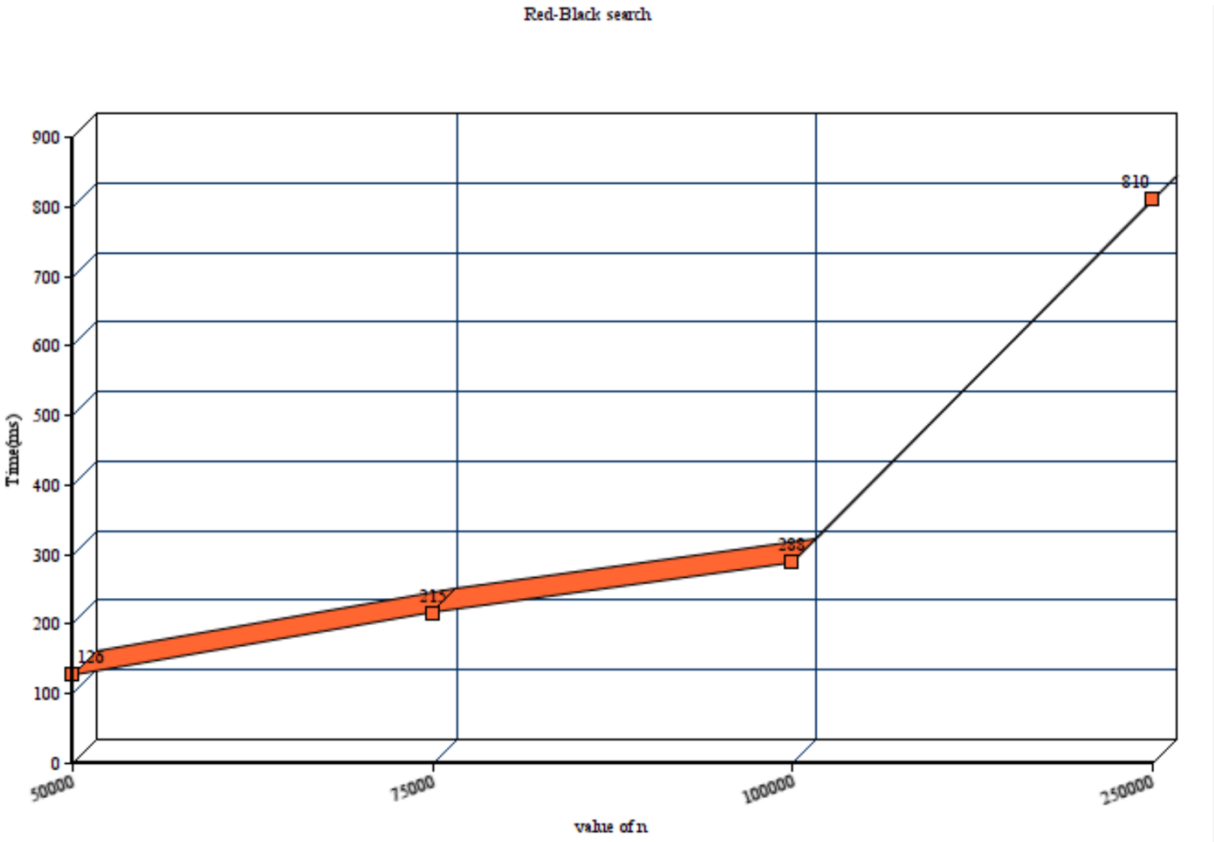


Readings of Red-Black tree: -

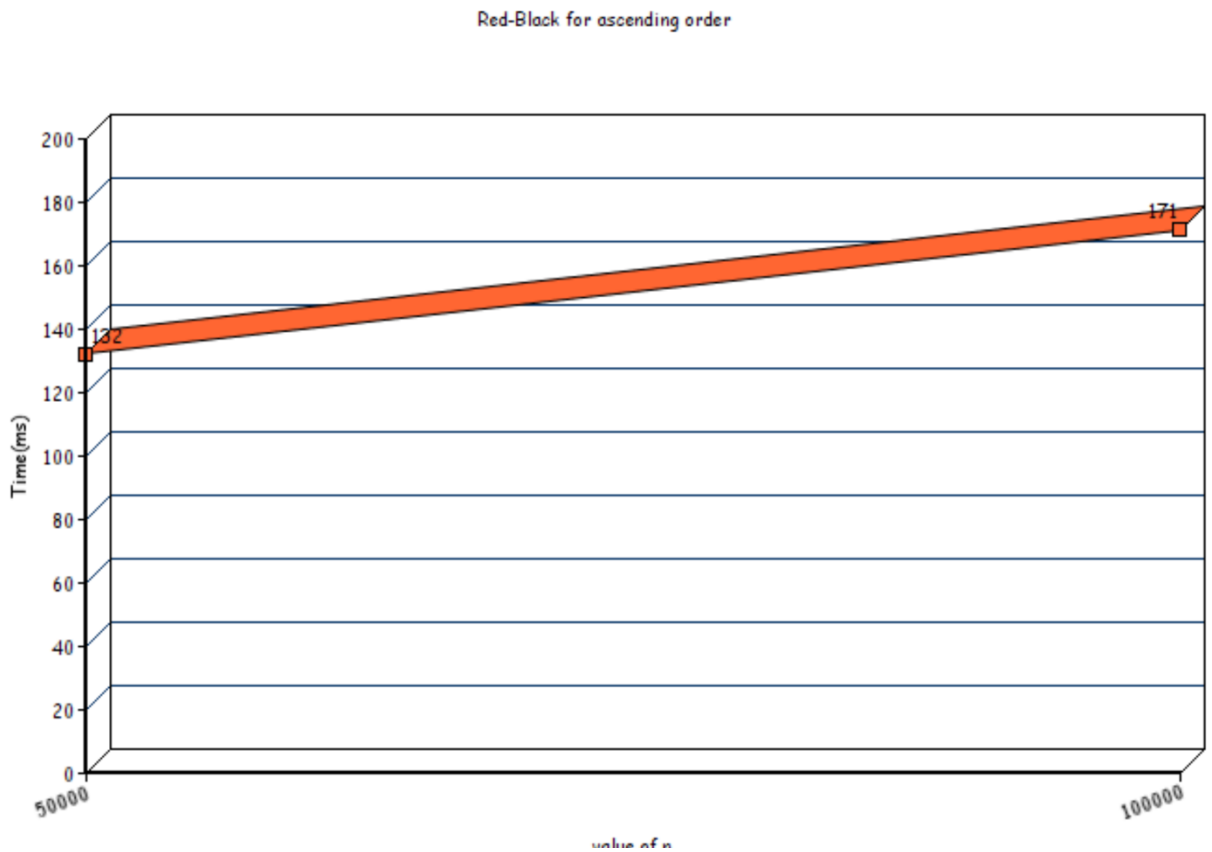
Homework :- 3

Red-Black Tree	Average	Average	Average	Average	Average	Average	Average	Average	Average	Average
Input	Real	User	Sys	Height of Tree	case 1	case2	case 3	Right-rotate	left-rotate	Black height
50000 0 1	126ms	60ms	50ms	18	25583	9716	9716		14609	10
75000 0 1	215ms	140ms	60ms	19	38642	14518	14518	21856	21900	10
100000 0 1	288ms	200ms	80ms	19	51241	19194	19194	28970	28917	11
250000 0 1	810ms	470ms	330ms	21	128338	48533	48533	72980	72662	11
50000 1 0	132ms	70ms	50ms	28	49666	0	0	49971	0	15
50000 -1 0	81ms	20ms	50ms	18	25661	9724	9724	14604	14557	10
100000 1 1	171ms	60ms	100ms	30	99964	0	0	0	99969	16
100000 -1 1	173ms	60ms	100ms	30	99964	0	0	99969	0	16

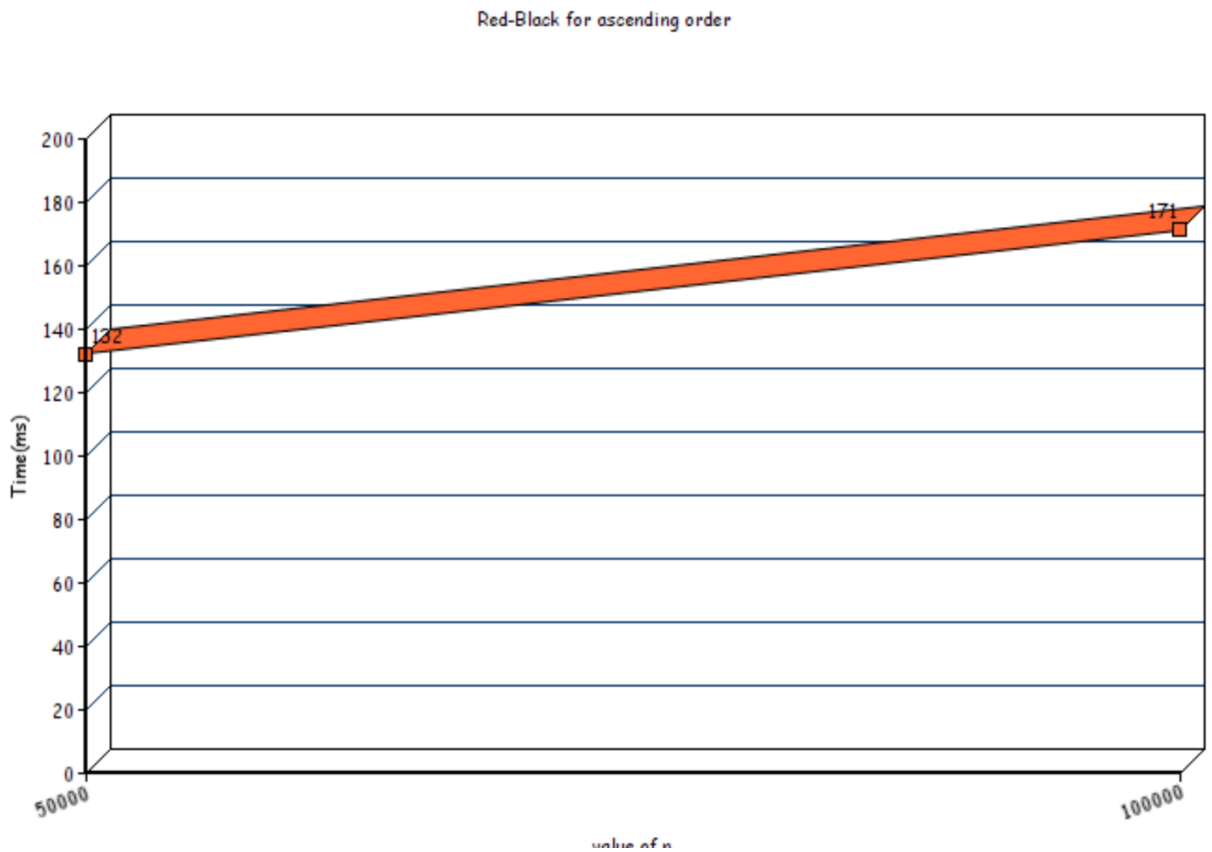
Homework :- 3


Red-Black search
900
800
700
600
500
400
300
200
100
0
Time(ms)
126
215
288
810
50000
75000
100000
250000
value of n

Homework :- 3


Red-Black for ascending order
200
180
160
140
120
100
80
60
40
20
0
Time(ms)
132
171
50000
100000

Homework :- 3



Conclusion :-

From the above graph it is clear that a red-black tree is a self-balancing tree, while a binary search tree is not. binary search tree is able to form long chains of nodes that can cause searches to take linear time whereas red black tree takes logarithmic time.

Red Black Tree: best case O(logN), worst case O(logN)
Binary Search Tree: best case O(logN), worst case O(N)