Student:Zili Luo
Student#:20001744

CISC 235
Assignment 2
Data Analyze part

```
when n = 1000 RtResult = 0,0,184,309,7
when n = 1000 RmResult = 0,0,4,43,453
when n = 2000 RtResult = 0,0,145,349,6
when n = 2000 RmResult = 0,0,3,15,482
when n = 4000 RtResult = 0,0,119,379,2
when n = 4000 RmResult = 0,0,2,9,489
when n = 8000 RtResult = 0,0,73,425,2
when n = 8000 RmResult = 0,0,1,3,496
when n = 16000 RtResult = 0,0,75,422,3
when n = 16000 RmResult = 0,0,1,4,495
```

Run the program serval times and get similar result.
The tables of Rt and Rm as following.

Here is the Rt table:

N	Rt<0.5	0.5≤Rt<0.75	0.75≤Rt≤1.25	1.25<Rt≤1.5	Rt>1.5
1000	0%	0%	36.8%	61.8%	1.4%
2000	0%	0%	29%	69.8%	1.2%
4000	0%	0%	23.8%	75.8%	0.4%
8000	0%	0%	14.6%	85.0%	0.4%
16000	0%	0%	15%	84.4%	0.6%

From the Rt table, we can conclude most of time Rt is between 1.25 and 1.5. So the Average Search time of Red-Black tree is better than simple Binary Search Tree in most of time.
And when n increase, the result group more tightly in range of 1.25 and 1.5.

Here is the Rm table:

N	Rm<0.5	0.5≤Rm<0.75	0.75≤Rm≤1.25	1.25<Rm≤1.5	Rm>1.5
1000	0%	0%	0.8%	8.6%	90.6%
2000	0%	0%	0.6%	3%	96.4%
4000	0%	0%	0.4%	1.8%	97.8%
8000	0%	0%	0.2%	0.6%	99.2%
16000	0%	0%	0.2%	0.8%	99%

From the Rm table, we can conclude nearly all the time Rm is greater than 1.5, which means Red-Black Tree's worst cases are always better than simple Binary Search Tree's worst case.
And when when n increase, the result group more tightly in range of 1.5 and higher.

Conclusion: The Red-Black Tree are superior to simple Binary Search Trees in both structure and worst-case-protect, the difference is more significant when n becomes greater.