Report of Database Lab4

Name: Nayun Xu
Student ID: 515030910635

1.Design decisions

1.1 IntHistogram

I use a ***Binary Index Tree*** to maintain the buckets, since the operation I will do to the buckets is single position modification and range sum query. I find there would be some situations that bucket number is larger than the range of attributes. I deal with such situations in the way that I set the width of buckets to 1, and use the first 'range' buckets. Leave the rest buckets empty.

2.Changes on API

I add some support methods to implement ***Binary Index Tree***.

3.Incomplete elements

I only follow the instruction to finish the project. I didn't use better estimation techniques.

4.Experience

I spent about 3-5 hours on this project. I first encounter problems when implementing the IntHistogram. The problem is that the range may be smaller than the number of buckets, and this will cause my initial implementation to divide zero. I use the method as I mentioned in **Section 1** to solve this problem. And when implementing JoinOptimizer, I spent some time understanding the pseudo code of the ordering algorithms, and going through the support methods and classes.