**Report**

1. Overview of the implementation

The class MySubmission implements the interface submission, and implements the customed equals method used for check whether two objects are equal.

The class Assignment has two primary data structures:

**private** HashMap<String, PriorityQueue<MySubmission>> gradeMap;

//map for unikey to submission time

**private** HashMap<String, TreeMap<Date, MySubmission>> timeMap;

the gradeMap is used for storing each user’s submission history, with PriorityQueue, we can get the max grade in O(1).

And the timeMap is used for stoing each user’s submission history with nested TreeMap, TreeMap is implemented based on red-black tree, so we can make the search and add operations in O(lgn).

The Main.java is used for testing functionality.

1. Time Complexity Analysis

⑴ Integer getBestGrade(String unikey)

Get user from hashMap with unikey is O(1), and peek the largest value from PriorityQueue is also O(1), so time complexity of this method is O(1)

(2) Submission getSubmissionFinal(String unikey)

The time complexity of searching in TreeMap is height of tree, and the height of the tree with n nodes is lgn, So time complexity is O(lgn)

(3) Submission getSubmissionBefore(String unikey , Date deadline)

Time complexity is also O(lgn), because search time complexity in AVL tree is O(lgn)

(4) Submission add(String unikey , Date timestamp , Integer grade)

The priorityQueue’s insertion is O(lgn), because after inserting the element, we should heapify the queue.

The TreeMap’s insertion is O(lgn), because when insert an element, we first search it in the tree, and insert it in appropriate position.

So the time complexity is O(lgn)

(5) void remove(Submission submission)

Search in priorityQueue is O(n), so the time complexity is O(n)

(6) List<String> listTopStudents()

The time complexity is O(n), because we first need to find the max grade of all the students, then check whether each student’s grade is equal to the max grade.

(7) List<String > listRegressions()

Assume number of students is n, and each student’s submission number is m, search the max value in TreeMap is O(lgm), we need to traverse all the students, so time complexity is O(nlgm)