**CSC122F2020BinarySearchTreeAndHashingLab**

**Author:** Filip Segota

**Problem summary:**

Program that studies times that different searches take. Searches used are hashing, binary search tree, and just sort with a binary search.

**List of requirements:**

* Each student has a unique id.
* Using merge sort for a sort
* Using recursion for a binary search
* Count number of compares as building the structures and as looking up for students
* Varying the percentage of population that is searched
* Varying the order of data (in order, random order, reverse order)

**Lessons learned:**

* Hashing and binary tree
* Using interfaces

|  |
| --- |
| Person |
| -name: String  -email: String  -id: String  -dob: String |
| +Person()  +Person(String, String, String, String)  +getName(): String  +getEmail(): String  +getID(): String  +getDob(): String  +setName(String): void  +setEmail(String): void  +setId(String): void  +setDob(String): void  +isEqual(Person): boolean  +compareTo(Person): int  +toString(): String |

* Merge Sort

|  |
| --- |
| Student |
| -gpa: float  -hours: int  -major: String |
| +Student()  +Student(String, String, String, String, float, int, String)  +getGpa(): float  +getHours(): int  +getMajor(): String  +setGpa(float): void  +setHours(int): void  +setMajor(String): void  +modifyGpa(int, char): void  +compareTo(Student): int  +toString(): String |

**Time:** 6hr (It took me more time than expected (4hr), mainly because of merge sort). I would make separate class for testing to make it look nicer if I had more time.

|  |
| --- |
| HashStudentNode |
| -key: int  -value: Student  -next: HashStudentNode |
| +HashStudentNode()  +HashStudentNode(Student)  +getKey(): int  +getValue(): Student  +getNext(): HashStudentNode |

|  |
| --- |
| BTStudentNode |
| -student: Student  -left: BTStudentNode  -right: BTStudentNode |
| +BTStudentNode()  +BTStudentNode(Student)  +getStudent(): Student  +getLeft(): BTStudentNode  +getRight(): BTStudentNode |

|  |
| --- |
| BTStudents |
| -root: BTStudentNode  -compares: int  -size: int |
| +BTStudents()  +preOrder(): String  +inOrder(): String  +postOrder(): String  +preOrder(BTStudentNode): String  +inOrder(BTStudentNode): String  +postOrder(BTStudentNode): String  +add(BTStudentNode, Student): BTStudentNode  +find(String, BTStudentNode) |

|  |
| --- |
| HashTableStudents |
| -bucketArray: ArrayList<HashStudentNode>  -capacity: int  -compares: int  -size: int |
| +HashTableStudents()  +HashTableStudents(int)  +getCapacity(): int  +hashIndex(int): void  +isEmpty(): boolea  +toString(): String |

|  |
| --- |
| DataStructureInterface |
| +add(Student): void  +find(String): Student  +getCompares(): int  +resetCompares(): void  +getSize(): int |

|  |
| --- |
| ArrayStudents |
| -students: Student[]  -capacity: int  -compares: int  -size: int |
| +ArrayStudents()  +ArrayStudents(int)  +getCapacity(): int  +sort(): void  +sort(int, int): void  +merge(int, int, int): void  +find(int, int, String): Student  +isEmpty(): boolean  +isFull(): boolean  +toString(): String |