

**Athlone Institute of Technology**

**Report**

***Medical Store Management System***

**with BST, Hash Table, Hash Map & Linked List**

**Software Design 4.2**

*Chhaya Sharma*

*A00268860*

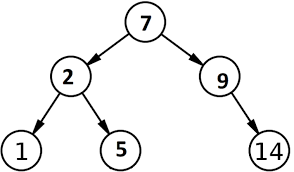
*Bachelor of Engineering (Honours) Software Engineering*

# Abstract

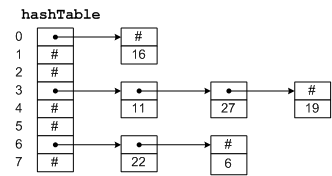
Medical Store Management allows user to *add* new medicines into system, *search* and *purchase* medicine. This system also helps in keeping track of medicine as user can see the updated stock after every purchase.

We have implemented the above functionalities using Binary Search Tree (BST), Hash Table, Hash Map and Linked List.

## Binary Search Tree (implemented)

We have used Binary Search Tree to insert new medicine into the records, searching the medicine and purchasing the medicine and for keeping a count of medicines.

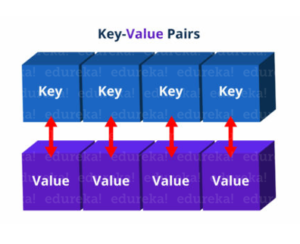
## Hash Table (implemented)

Hash Table is also implemented to add new medicine into database, user can search the medicine using Hash table. User can also see the updated stock after every purchase using Hash Table.

***We have also compared the total no of Iterations between Binary Search Tree and Hash Table for searching an element to show Efficiency of Hash Table and we found that for larger data Hash Table is efficient to use.***

## Hash Map (From Collection Framework:

We have used Hash Map from collection framework to insert, search and update the medicines. Using collection framework makes it easier to implement functionality.



## Linked List (From Collection Framework)

A close up of a clock

Description automatically generatedLinked List is also used to insert the medicines.

# Introduction

Here are system features and how they are implemented:

* Add Medicine Details (Id, Name, Disease, Price, Stock and Expiry date) using BST, Hash Table, Hash Map and Linked List,
* Search Medicines using BST, Hash Table and Hash Map.
* Compared the Performance of Operations of BST and Hash Table,
* Aggregation: Class Date in class Medicine
* Medicines are stored in node ((for BST) and HNode (for Hash Table))
* Iterator pattern is used to iterate through the array

# Class Diagram

## Classes:

* BST: To implement Binary Search Tree to add, search and purchase medicine.
* Node: Store the details of Medicine in BST
* HashTable: To implement Binary Search Tree to add, search and purchase medicine.
* HNode: Store the details of medicine for Hash Table
* Mainframe: For GUI screen
* Medicine: To maintain medicine fields
* Date: With variable d, m and y for day, month and year respectively.

# Implementation and Functionalities:

## Binary Search Tree

#### Add Item as Node:

This function is used to insert the medicine A screenshot of a cell phone

Description automatically generated

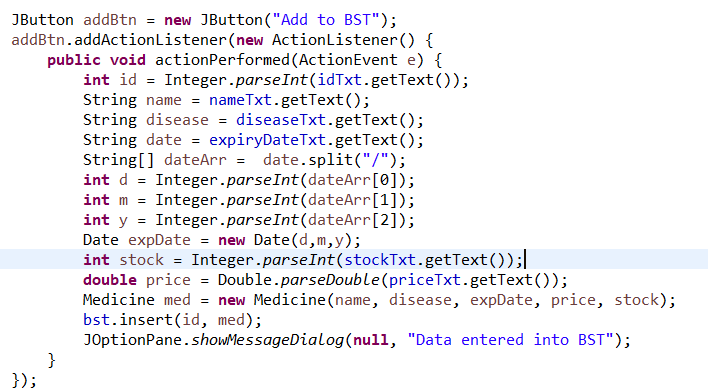
#### A screenshot of a cell phone Description automatically generatedSearch an Item:

#### Update an Item:

A screenshot of a cell phone

Description automatically generated

#### To enter data into BST (Mainframe)



## Hash Table

#### A screenshot of a cell phone Description automatically generatedTo calculate Index, create Hash Table

#### A picture containing people, holding, table, room Description automatically generatedInsert function

#### A screenshot of a cell phone Description automatically generatedSearch Function

#### A screenshot of a cell phone Description automatically generatedUpdate Function

# Screenshots

#### Main Screen

A screenshot of a cell phone

Description automatically generated

#### Adding Medicine (By different Data Structures)

A screenshot of a cell phone

Description automatically generated

#### Search a Medicine:

As screenshot below shows that for searching the medicine with id 20 BST took 6 iterations while Hash Table took only 1 iteration which makes it more efficient

A screenshot of a social media post

Description automatically generated

#### Purchase Medicine:

The screenshot below shows that after purchase the stock is updated for all

A screenshot of a social media post

Description automatically generated

## Conclusions & Learnings:

* Learnt to implement two data structures BST and Hash Table which improved my knowledge of data structure.
* Comparison of performance of BST and Hash Table. For BST Time Complexity is O(LogN) and Hash Table it is O(1) which makes Hash Table faster when we are concerned about Insert, Search and Delete operations.
* Different collection framework and their functions which helped me learning collection framework.