

Technical Analysis Stock Screener

# Low Level Design – High Level Design

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Technical Analysis Stock Screener** | | | | |
| **Date** | **Version** | **Author** | **Brief Description of Changes** | **Approver** |
| 28-09-2022 | 1.0 | Group-3 |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

History:

|  |  |
| --- | --- |
| **Index** | |
| 1. Introduction | ------------------------------------------------ 3 |
| 1.1 Intended audience | ------------------------------------------------ 3 |
| 1.2 Project purpose | ------------------------------------------------ 3 |
| 1.3 Key project objective | ------------------------------------------------ 3 |
| 1.4 Project scope | ------------------------------------------------ 4 |
| 2. Design overview | ------------------------------------------------ 4 |
| 2.1 Design objective | ------------------------------------------------ 5 |
| 2.2 Design alternative | ------------------------------------------------ 5 |
| 2.3 User interface paradigms | ------------------------------------------------ 5 |
| 2.4 Validations | ------------------------------------------------5 |
| 2.5 System Requirements | ------------------------------------------------5 |

3. Detailed [System](#_bookmark0) Design -------------------------------------------------- 6

3.1 Flowchart of main application 7

|  |  |
| --- | --- |
| 4. Functional overview | ------------------------------------------------ 9 |
| 4.1 Header files Included | ------------------------------------------------ 9 |
| 4.2 Functions Included | -----------------------------------------------9 |

|  |  |
| --- | --- |
| 5. Coding | ------------------------------------------------ 10-14 |

|  |  |
| --- | --- |
| 6. Tools used | ------------------------------------------------ 15 |
| 6.1 Make file | ------------------------------------------------ 15 |
| 6.2 Valgrind | ------------------------------------------------ 16 |
| 6.3 Splint | ------------------------------------------------ 17 |

|  |  |
| --- | --- |
| 7. Testing | ------------------------------------------------ 18 |
| 7.1 Integration testing | ------------------------------------------------ 18 |

## Introduction

The Technical Analysis Stock Screener is a system that allows the user to enter into the system and check the available listed Stocks and their recommendations. Here the recommendations are done based on Technical Analysis. The user can see the recommendation based on those suggestions the user can do any of these like Buy, Sell and Hold the Stocks.

### Intended Audience: -

The target audience set for this project can be identified as Equity Stock traders / Investors.

### Project Purpose: -

The Technical Analysis Stock Screener is a project that helps us understand the basic concepts of functions like file handling, and data structure. Here this project mainly displays the list of stocks to the user and give recommendations based on the averages calculated between the latest 10 days average stock to 50 days average stock. Based on these averages It provides the recommendations like Buy, Sell and Hold of the stocks.

### Key Project Objectives: -

1. Allow the User to enter the System
2. Can see the list of available stocks
3. updating Functionality upto latest 20 stocks
4. Displays all the records of The stock
5. calculating the Average close Stocks.
6. Suggest the recommendations like Buy, Sell or Hold.

### Project scope : -

This project aims to create the development of an Technical Analysis Stock Screener

,Which takes the Stock information from various Stocks, adds it to the database and processes the Average of latest Stock records, And give recommendations based on the technical Analysis.

## Design Overview: -

### Technical Analysis Stock Screener Comprises of the following modules:

|  |  |
| --- | --- |
| Name of the Module | List of Stocks Module |
| Handled by |  |
| Description | It display list of stocks available |

|  |  |
| --- | --- |
| Name of the Module | Main Menu Module |
| Handled by |  |
| Description | Consist of List of choices and exit option |

|  |  |
| --- | --- |
| Name of the Module | Sub Menu Module |
| Handled by |  |
| Description | It consist the information about selected choice |

|  |  |
| --- | --- |
| Name of the Module | Exit Module |
| Handled by |  |
| Description | It returns to the main menu from sub menu |

* 1. **Design Objectives:**
     + Add No of stocks into the data.
     + User can select which stock data want to see.
     + It will Calculate averages of the stock records.
     + Based on average it show recommendations to the user.
     + Recommendations are buy, sell and hold.
     + Then user can take decision based on technical Analysis of stocks.

### Design Alternative: -

We have used a linked list structure to store data i.e.. Stock data files, to find averages and for recommendations

### User Interface Paradigms: -

This project Directly perform the Technical analysis on stocks and give recommendations to the user based on the analysis.

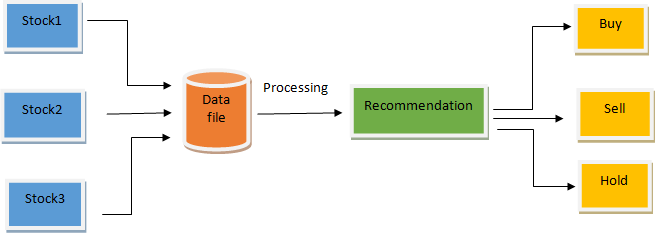
### Validation: -

* User cannot enter more than one choice at a time
* User cannot perform any operations in the stocks data.

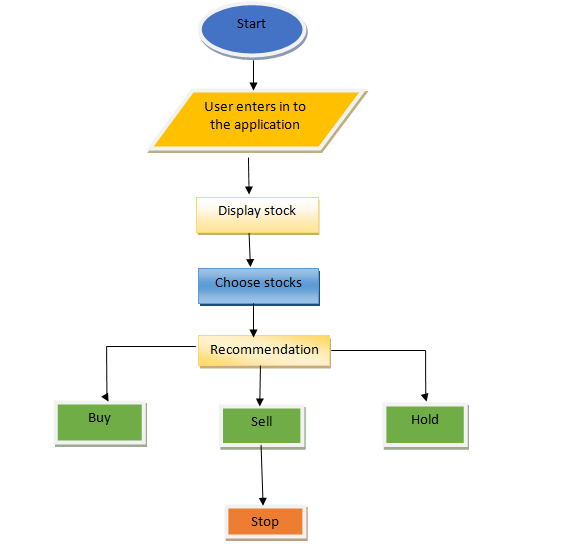
### System Requirements:-

* + - Language Used: C language
    - Tools Used : Make,Valgrind
    - Compiler : Gcc
    - Environment : Linux

1. **DETAILED SYSTEM DESIGN:**



# Flow Chart of the application

****

**4.Functional overview:**

4.1 Following Header files are include:

* #include<stdio.h>
* #include<stdlib.h>
* #include<string.h>
* #include<main.h>
* #include<struct.h>

4.2 Following Functions are included:

1. selectChoice( ) :

This Function mainly used to give the choices to the user so that based on the list of choices the user can select from it.

1. loadStockData( ) :

This function is used to load the CSV file into the linked list. By using the file pointer we read the CSV file into the linked list.

1. showRecommendations( ) :

This Function is used to give recommendations to the stock holders based on the average calculated. It give recommendations like buy, sell or hold.

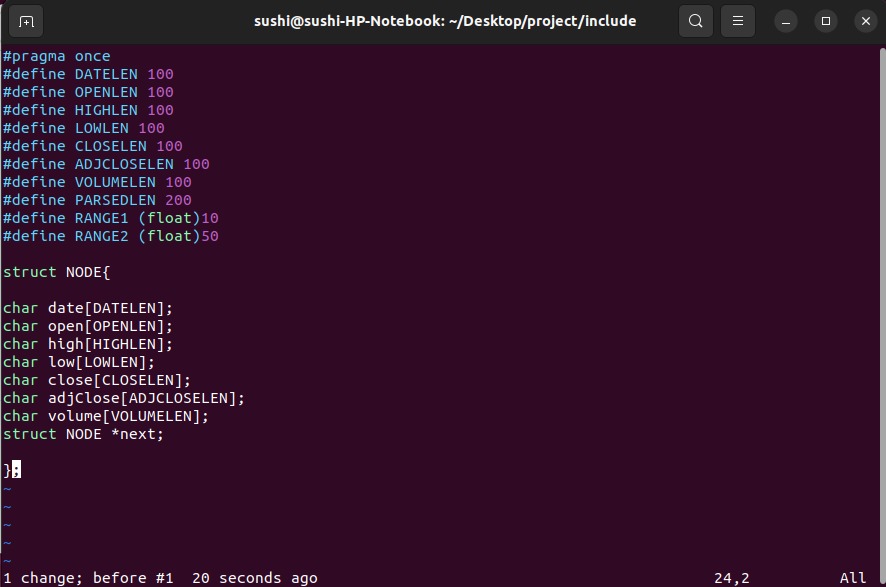
## 5.Coding:

## Main.h:

## 

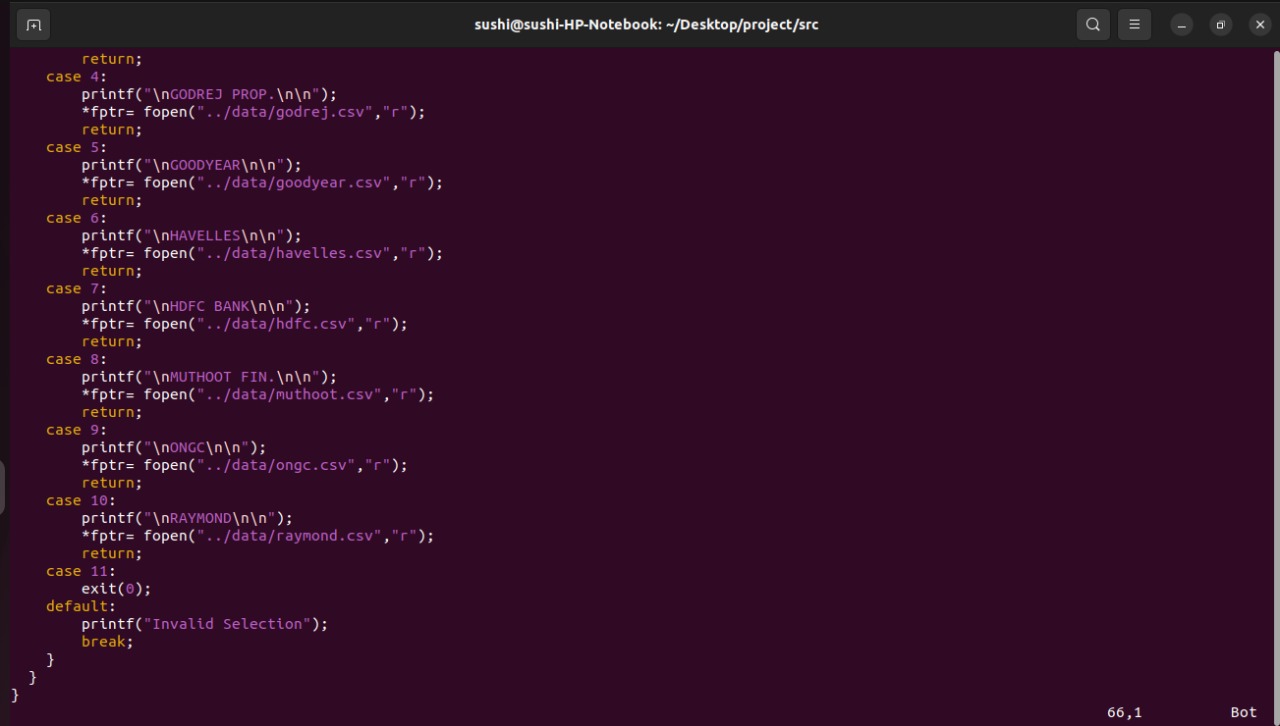
## mainh screenshot.jpeg

**Struct.h:**

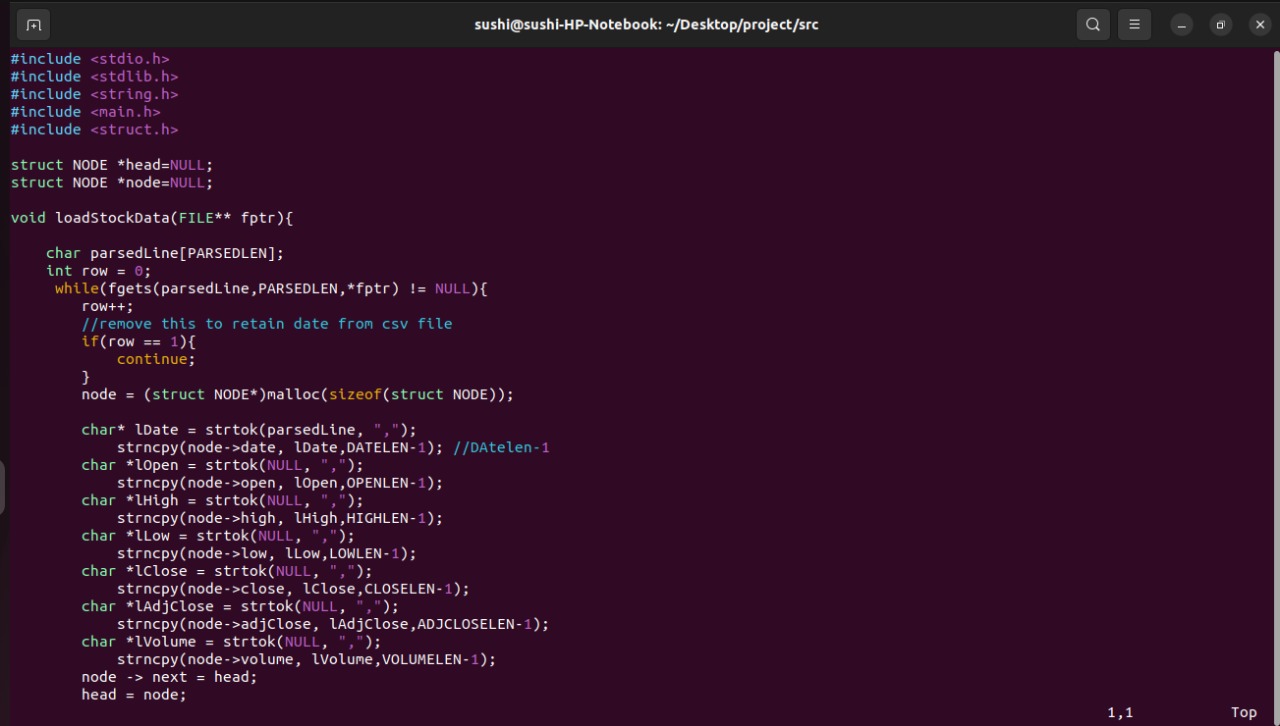
****

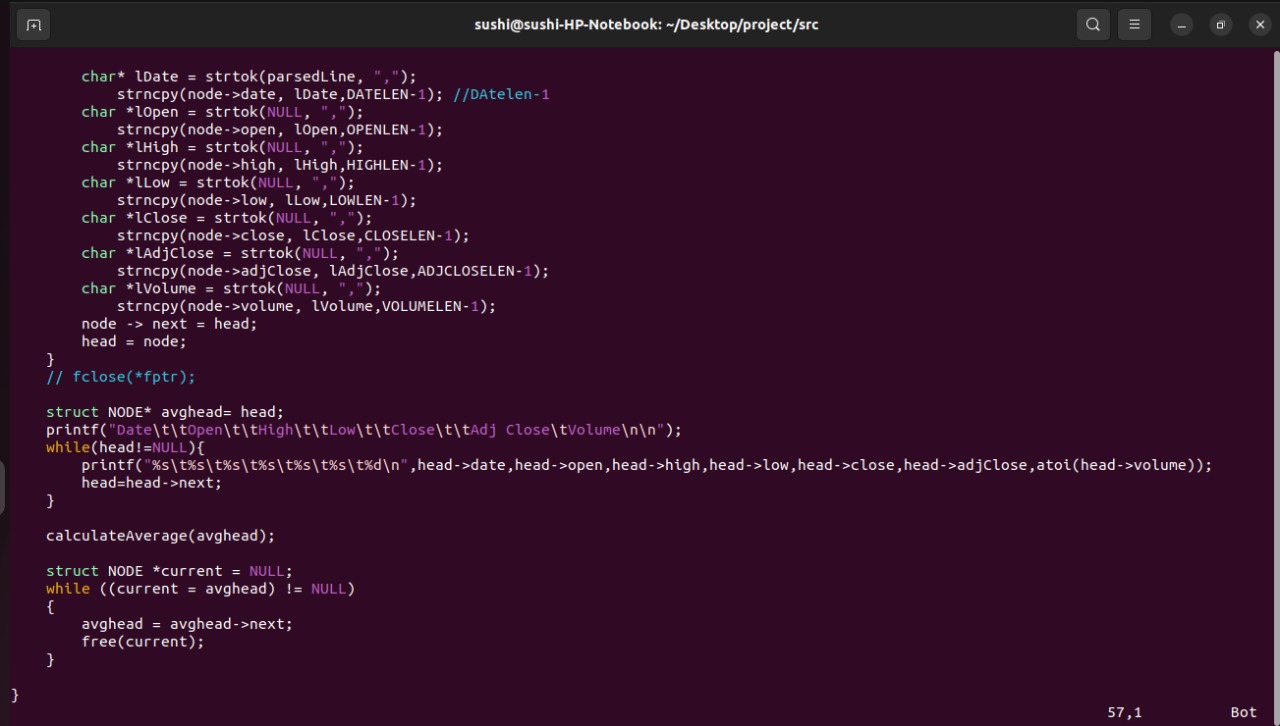
**SelectChoice ( ):**

****

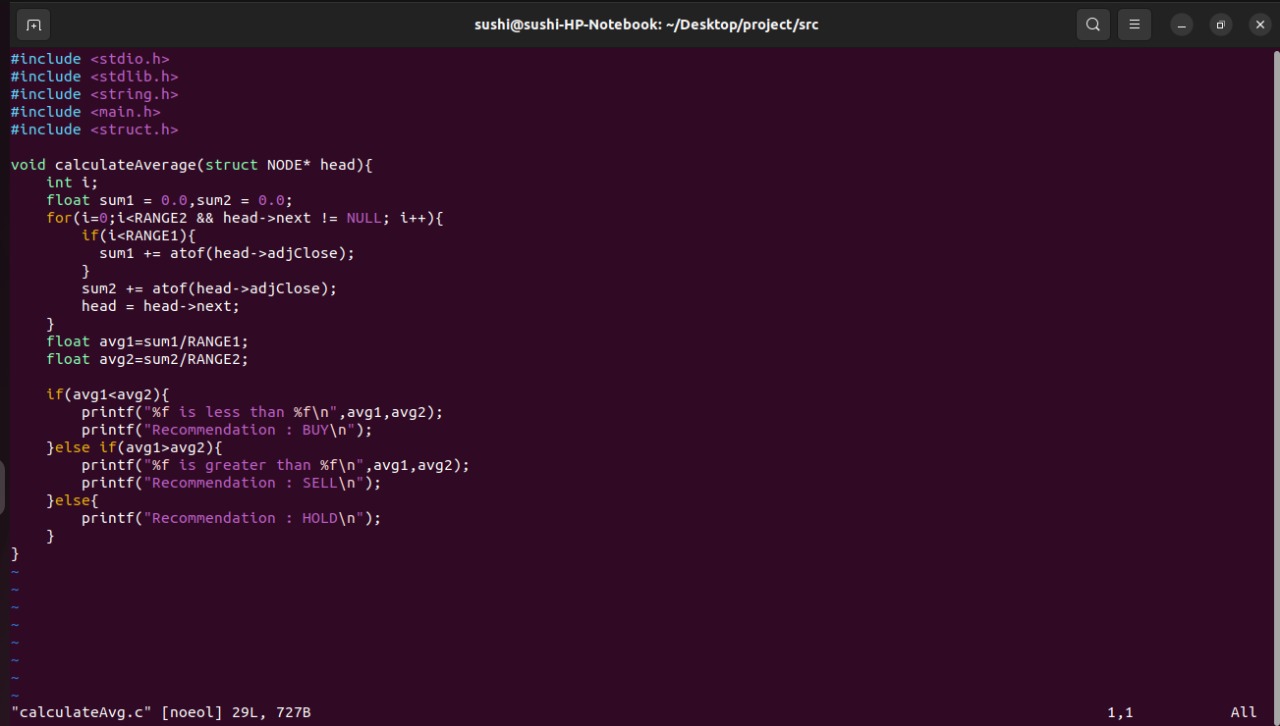
****

**loadStockData ( ):**

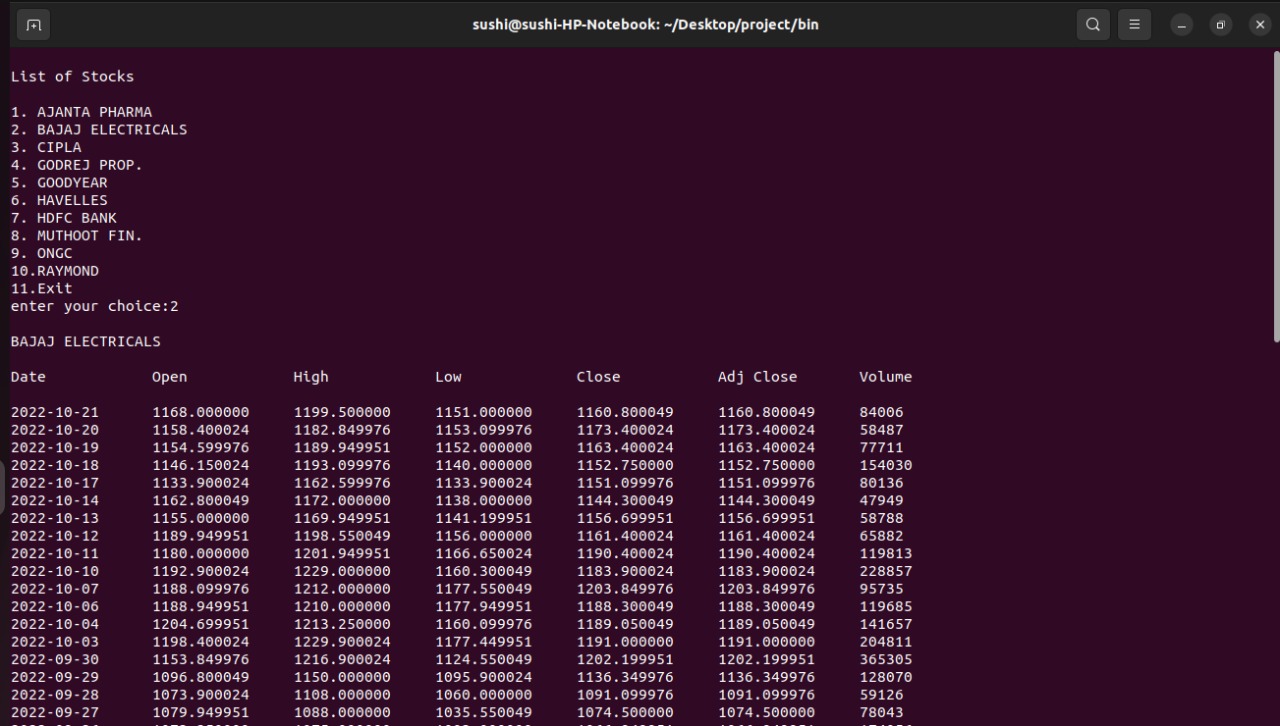
****

****

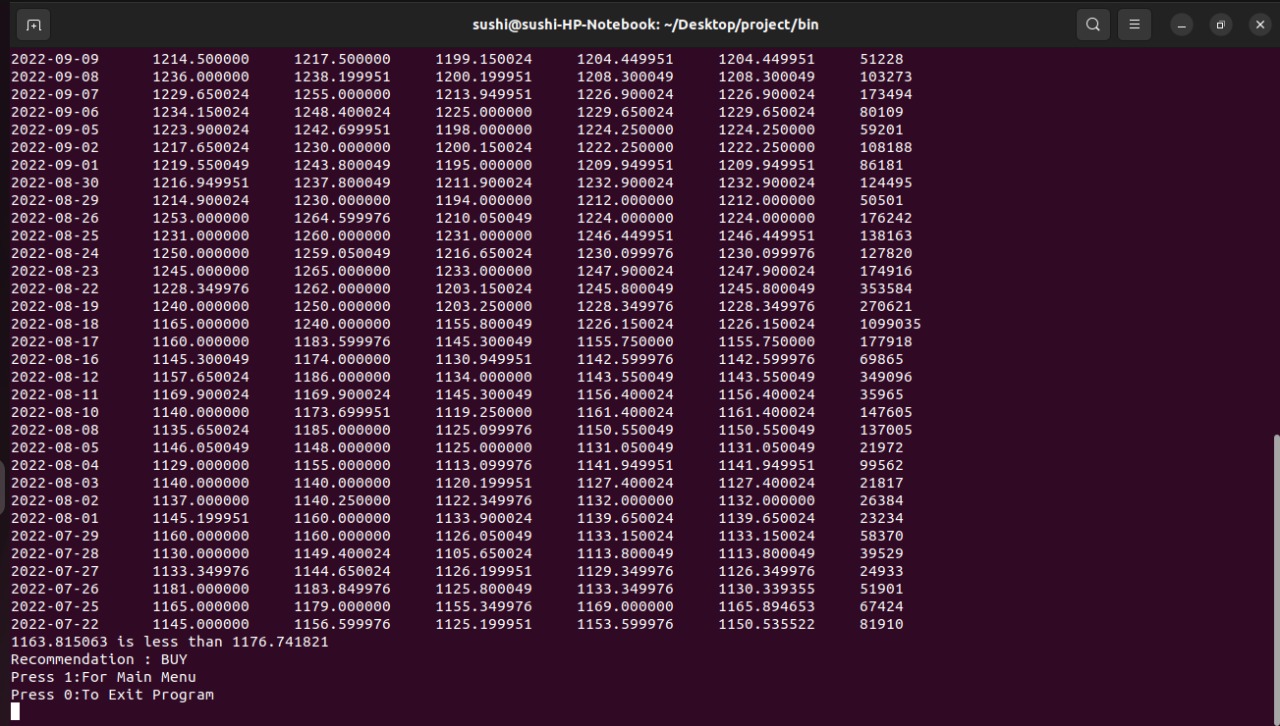
**showRecommendations ( ):**

****

**List of stocks:**

****

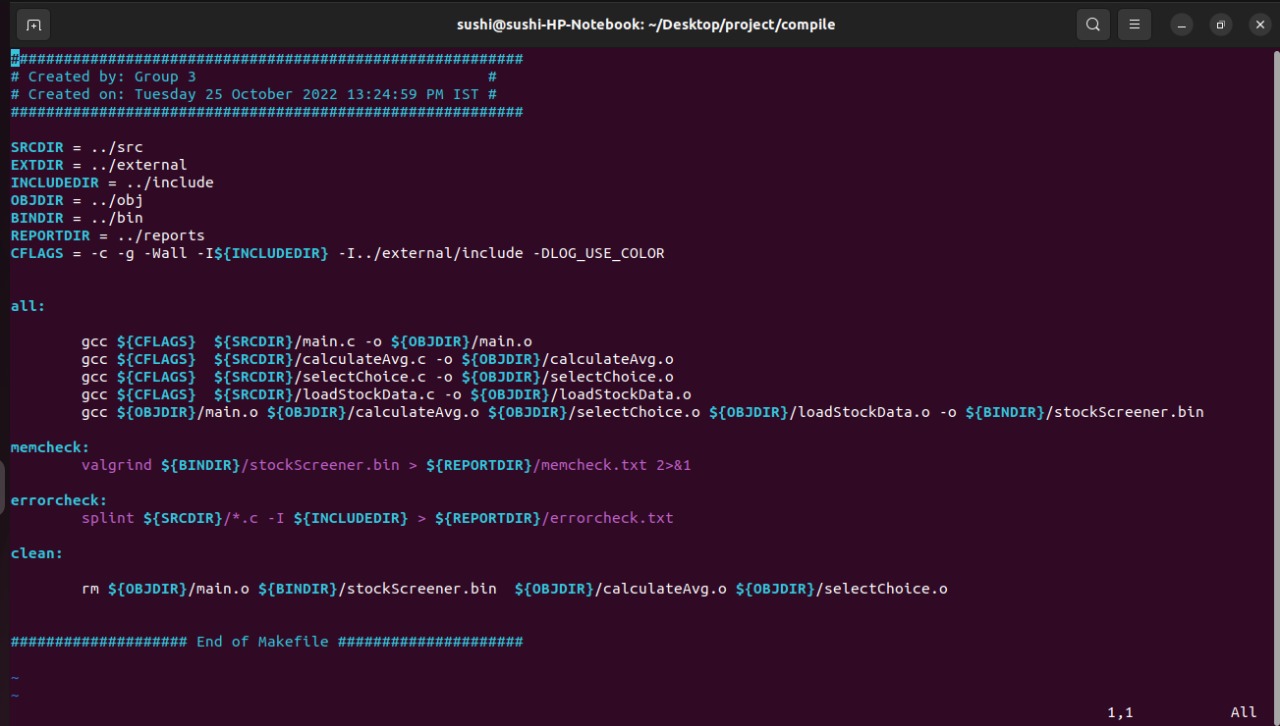
**Output:**

****

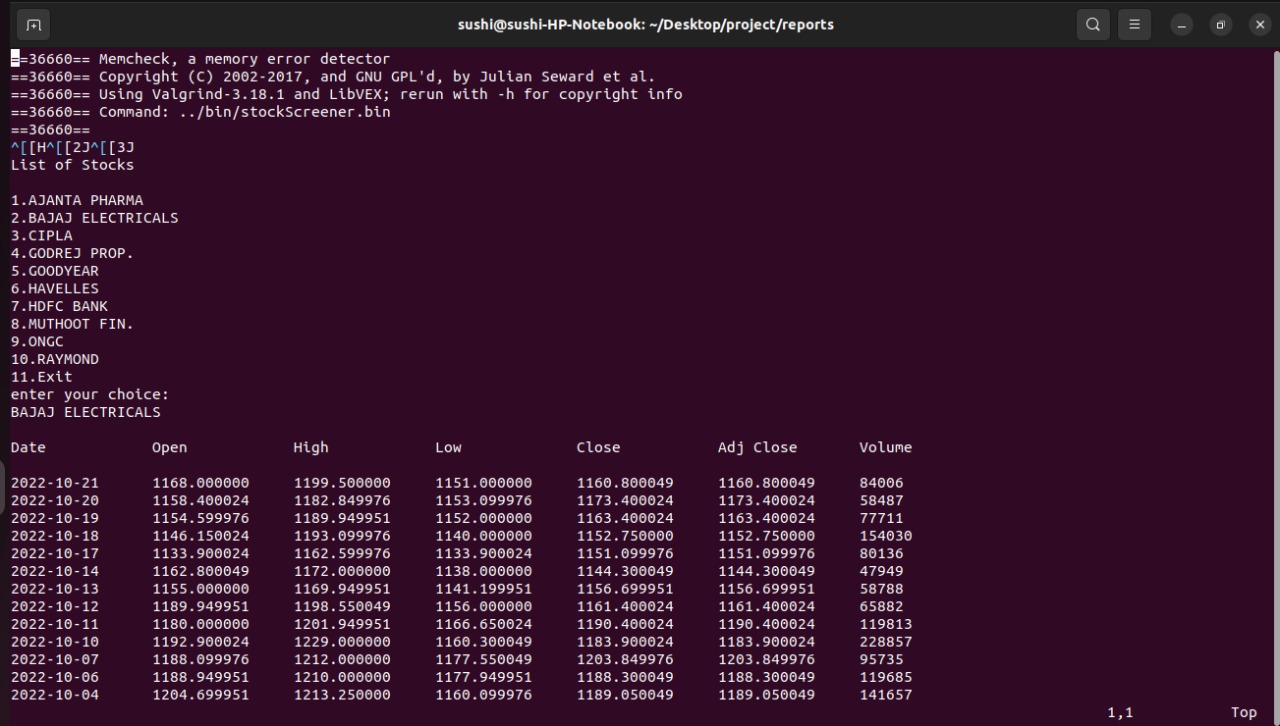
**6.Tools Used**:

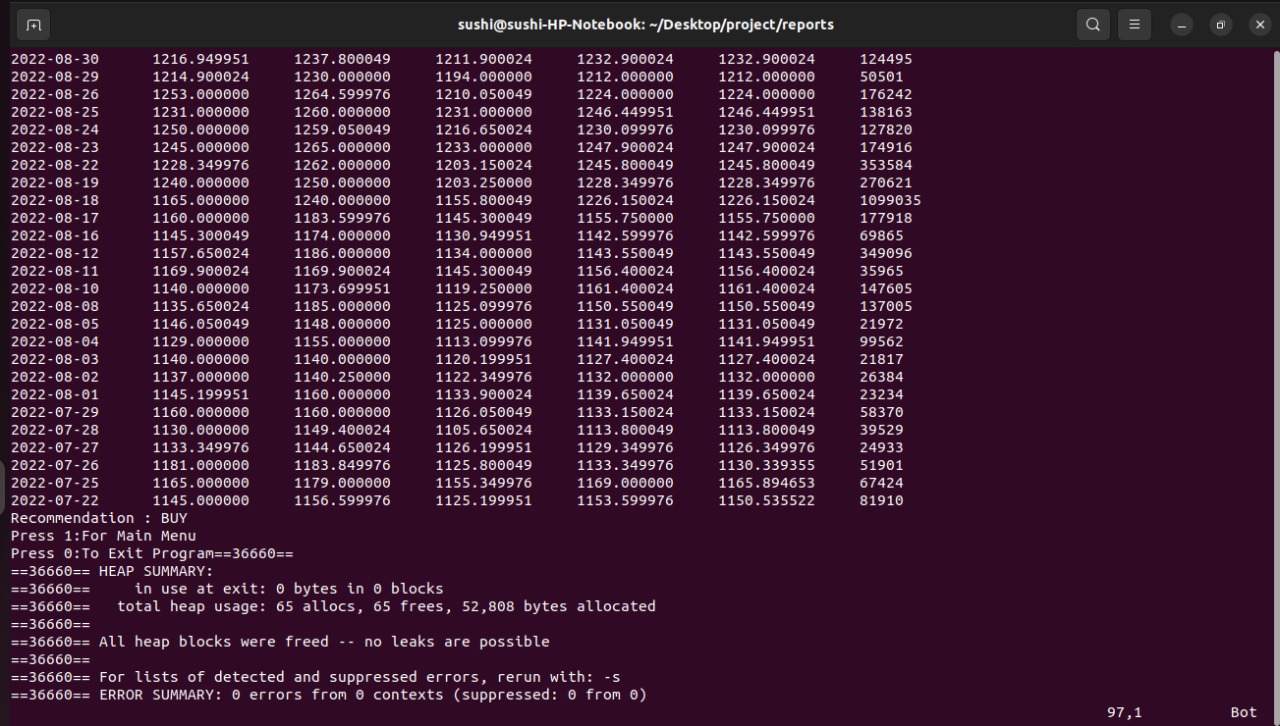
**1.Make file:**

Make file is a set of commands with variable names and targets to create object files and to remove them. In a single makefile we can create multiple targets to compile and to remove object, binary files. We can compile our project any number of times using make file.

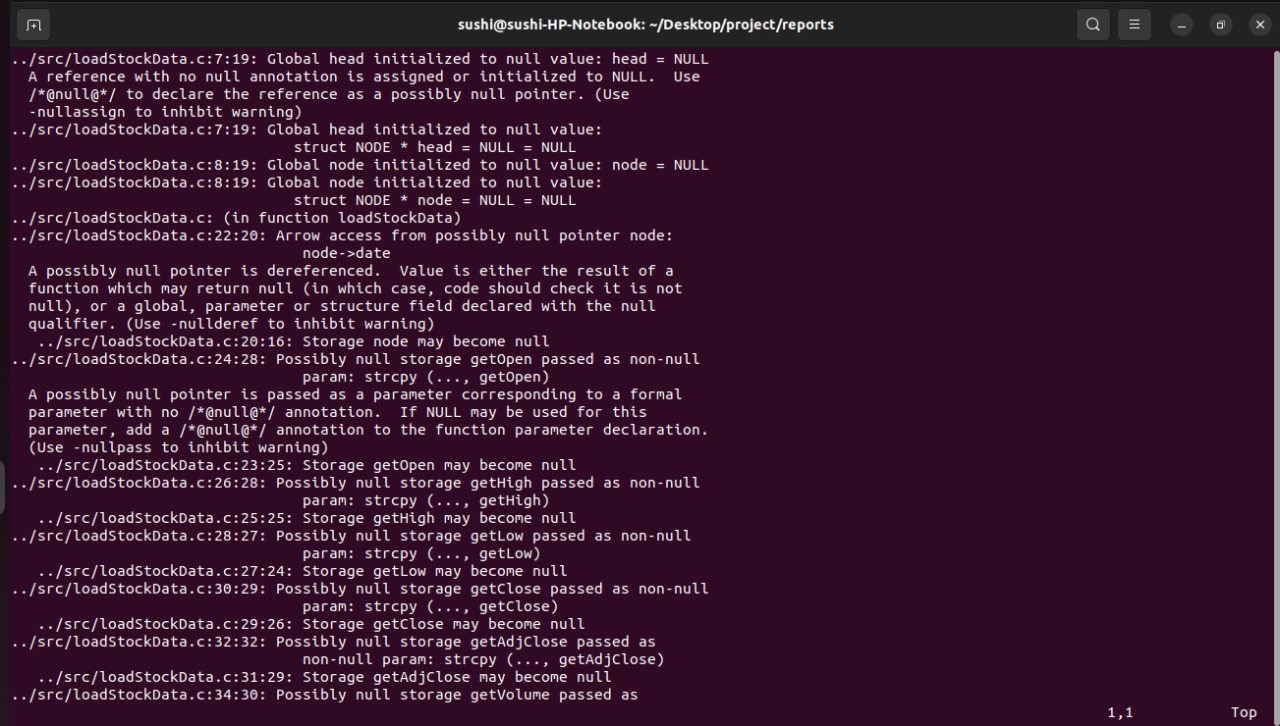
****

**2.Valgrind:**

****

****

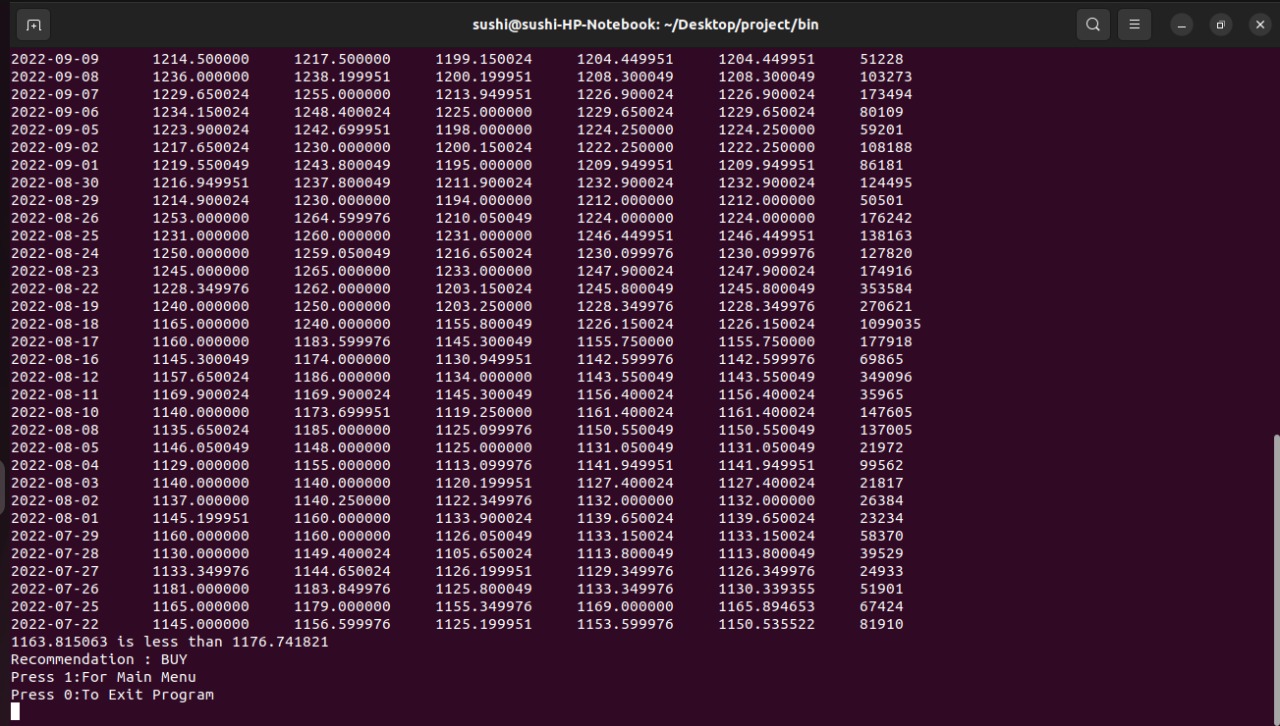
**3**.**Splint**:



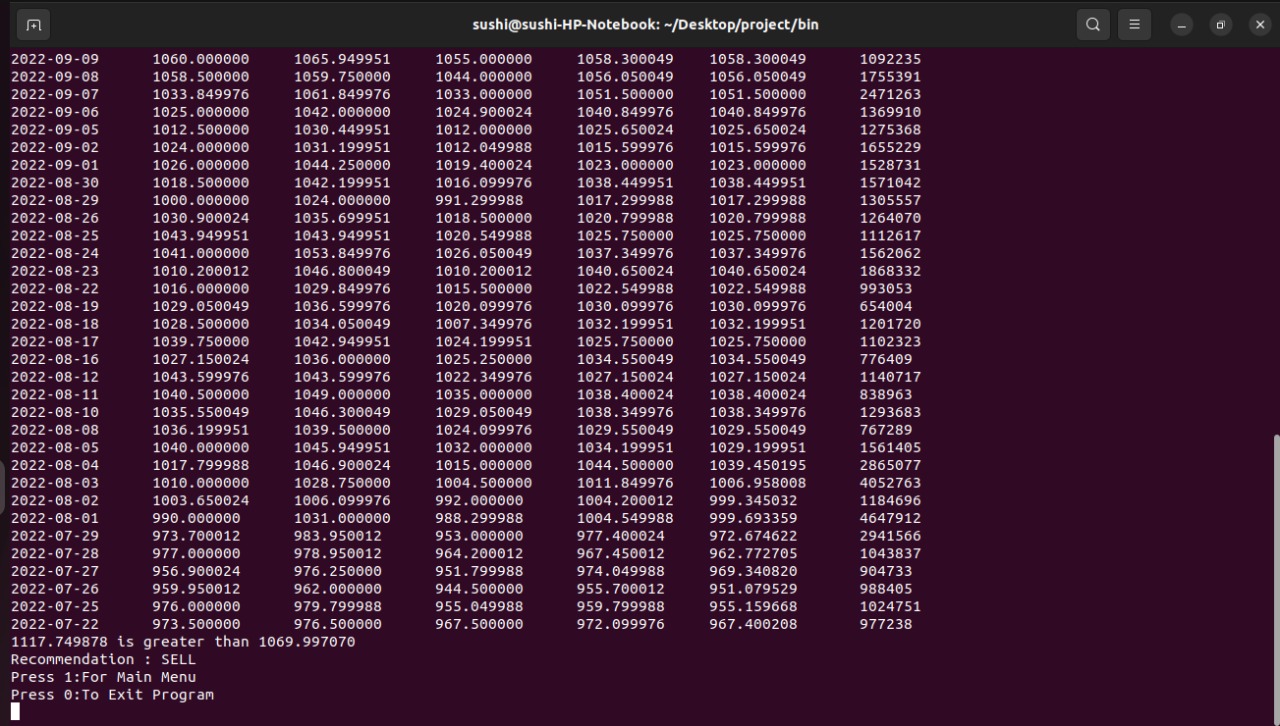
**7.Testing:**

**7.1 Integration testing:**

**Test\_case 1 ( Recommendation Buy):**

****

**Test case-2(Recommendation Sell):**

****