Project is to implement Single linked list, Double linked list, Binary Search tree and AVL tree data structures by developing a video store (1) maintain the information of videos and customers (by using the data structures Linked Lists and Trees), and (2) operate and record customers’ transactions.

When project starts, the main Menu is printed which would look like this:

===========================

Select one of the following

1: To add a video

2: To delete a video

3: To add a customer

4: To delete a customer

5: To check if a particular video is in store

6: To check out a video

7: To check in a video

8: To print all customers

9: To print all videos

10: To print in store videos

11: To print all rent videos

12: To print the videos rent by a customer

13: To exit

In the main Menu, requests 1-4 are to maintain the information of videos and customers, requests 5-7 are for customers’ transactions, and requests 8-12 are to test the correctness of the code.

After the main Menu is shown,

1. Enter 1

(1) a message is printed to ask the information of the video;

(2) the information is entered;

(3) the video is added to the data structure if the provided information is valid;

(4) the main Menu is shown.

2. Enter 2

(1) a message is printed to ask the information of the video;

(2) the information is entered;

(3) the video is deleted from the data structure if the provided information is valid;

(4) the main Menu is shown.

3. Enter 3

(1) a message is printed to ask the information of the customer;

(2) the information is entered;

(3) the customer is added to the data structure if the provided information is valid;

(4) the main Menu is shown.

4. Enter 4

(1) a message is printed to ask the information of the customer;

(2) the information is entered;

(3) the customer is deleted from the data structure if the provided information is valid;

(4) the main Menu is shown.

5. Enter 5

(1) a message is printed to ask the information of the video;

(2) the information is entered;

(3) “true” is printed if the video is in store; “false” is printed otherwise;

(4) the main Menu is shown.

6. Enter 6

(1) a message is printed to ask the information of the customer and video;

(2) the information is entered;

(3) the video is checked out if the video is in store;

(4) the main Menu is shown.

7. Enter 7

(1) a message is printed to ask the information of the video;

(2) the information is entered;

(3) the video is checked in if the provided information is valid;

(4) the main Menu is shown.

8. Enter 8

(1) all customers are printed;

(2) the main Menu is shown.

9. Enter 9

(1) all videos (in store videos and rent videos) are printed;

(2) the main Menu is shown.

10. Enter 10

(3) the in store videos are printed;

(4) the main Menu is shown.

11. Enter 11

(1) all rent videos are printed;

(2) the main Menu is shown.

12. Enter 12

(1) a message is printed to ask the information of the customer;

(2) the information is entered;

(3) the videos rent by the customer are printed if the provided information is valid;

(4) the main Menu is shown.

13. Enter 13

(1) “Goodbye” message is printed and program terminates.

Project can be run in 8 stages:

**Stage 1:** “java VideoStore SLL”. The parameter “SLL” controls program to use Singly Linked List to store data. Program prints the main Menu and wait to interact with the operator. Then some videos and customers will be added, some customers will check in/check out videos, and information will be printed to verify the correctness of your code. Similar operations will be worked in stages upto 4 when project is run with other data structures like “java VideoStore DLL”, java VideoStore BST”, java VideoStore AVL”. From stage 5 project can be run with parameters like “java VideoStore SLL 300 20 100” where the parameter “SLL” controls the project to use Singly Linked Lists to store data, the parameter 300 controls the number of videos, the parameter 20 controls the number of customers, and the parameter 100 controls the number of requests. (Note that 300, 20, 100 could be replaced by other positive integers). Project without any **human interaction performs** (1) initialize 300 videos and 20 customers (no video has been rented); (2) randomly generate 100 **transaction requests (a transaction request is request 5 or request 6 or request 7)**, and arrange the requests in a queue; (3) process the requests in the queue one by one; (4) print the total service time (**i.e. process time of (3)**); (5) terminate. Similar Operations are performed up to stage 8 with other data structure implementation when project is run with parameters like “java VideoStore DLL 300 20 100”, “java VideoStore BST 300 20 100”, “java VideoStore AVL 300 20 100”