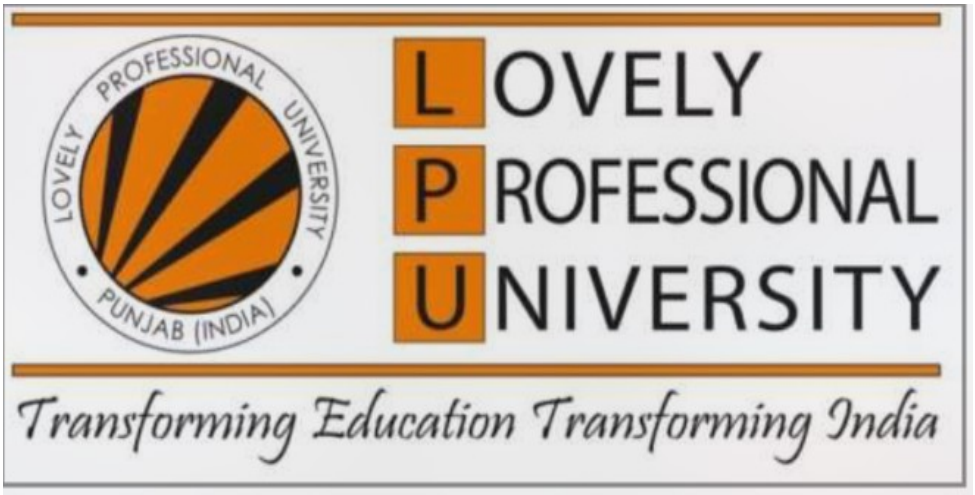
DATA STRUCTURES

CSE228

INITIAL PROJECT REPORT



Submitted By:

CHIRAG

K22UR

R.No: 64

Submitted To:

Shubham Sharma

UID: 64339

TOPIC: People in nth Generation of a Family Tree

PROBLEM STATEMENT:

Create an application that allows users to add members to family tree

Assume a simple parent child relationship between members (not taking into accospouses)

Members can be male or female

Each member has a unique name

No child has more than one parent

Children in the tree should be in lexicographical order of names (dictionary-based from left to right

Tree has only one ROOT (only one member can be at the top of the tree)

NOTE: IT IS MANDATORY TO USE A DATA STRUCTURE

Title: A Simple Family Tree Application in Java

Abstract:

This report presents the design and implementation of a basic family tree application in Java. The application allows users to create and manage a family tree, with support for adding family members, specifying their gender, and viewing the family tree structure. The application represents the family tree using object-oriented principles and a text-based interface.

1. Introduction:

A family tree is a visual representation of familial relationships, often depicting the parent-child connections within a family. This report outlines a simple Java application for creating and managing a family tree, focusing on essential requirements for representing family members and their relationships.

2. Functional Requirements:

The Java family tree application includes the following features and functional requirements:

- Creation of a family tree with a single root member.
- Addition of new members to the tree, specifying their gender (Male/Female).
- Ensuring that each child has a single parent.
- Displaying the family tree structure, sorted lexicographically by member names.

- Implementing basic error handling and input validation.

3. Implementation:

The family tree application is implemented in Java and includes two primary classes: `FamilyMember` and `FamilyTreeApp`.

- `FamilyMember` class represents a family member, storing their name, gender, and a list of children.
- `FamilyTreeApp` class is the main application that allows users to interact with the family tree.

4. Usage:

The application is executed via the command line. Users can interact with the application through a menu system, enabling the following actions:

- Launching the application.
- Displaying the family tree structure.
- Adding new family members, specifying the parent's name, child's name, and gender.
- Exiting the application.

5. Sample Usage:

Here's an example of using the Java family tree application:

- Launch the application.
- Add a root member named "John" as the initial member.
- Add child members under "John."
- Display the family tree structure, showing hierarchical relationships.
- Exit the application.

6. Limitations and Future Improvements:

The current Java family tree application has some limitations:

- It does not support multiple spouses for any member.
- It lacks a graphical user interface (GUI) and is text-based.
- The error handling is basic and can be improved.
- It does not save the family tree data beyond the current session.

To enhance this application, the following improvements can be considered:

- Implement a GUI for a more user-friendly experience.
- Allow multiple spouses and handle complex family structures.
- Persist the family tree data using a database or file storage.
- Improve error handling and provide more informative error messages.

7. Conclusion:

The Java family tree application provides a basic framework for creating and managing family trees. It demonstrates the use of Java classes and objects to represent familial relationships and a simple command-line interface. While it meets the fundamental requirements, there is potential for expansion and improvement, making it a starting point for a more comprehensive family tree management system.