DATA STRUCTURES AND ALGORITHMS
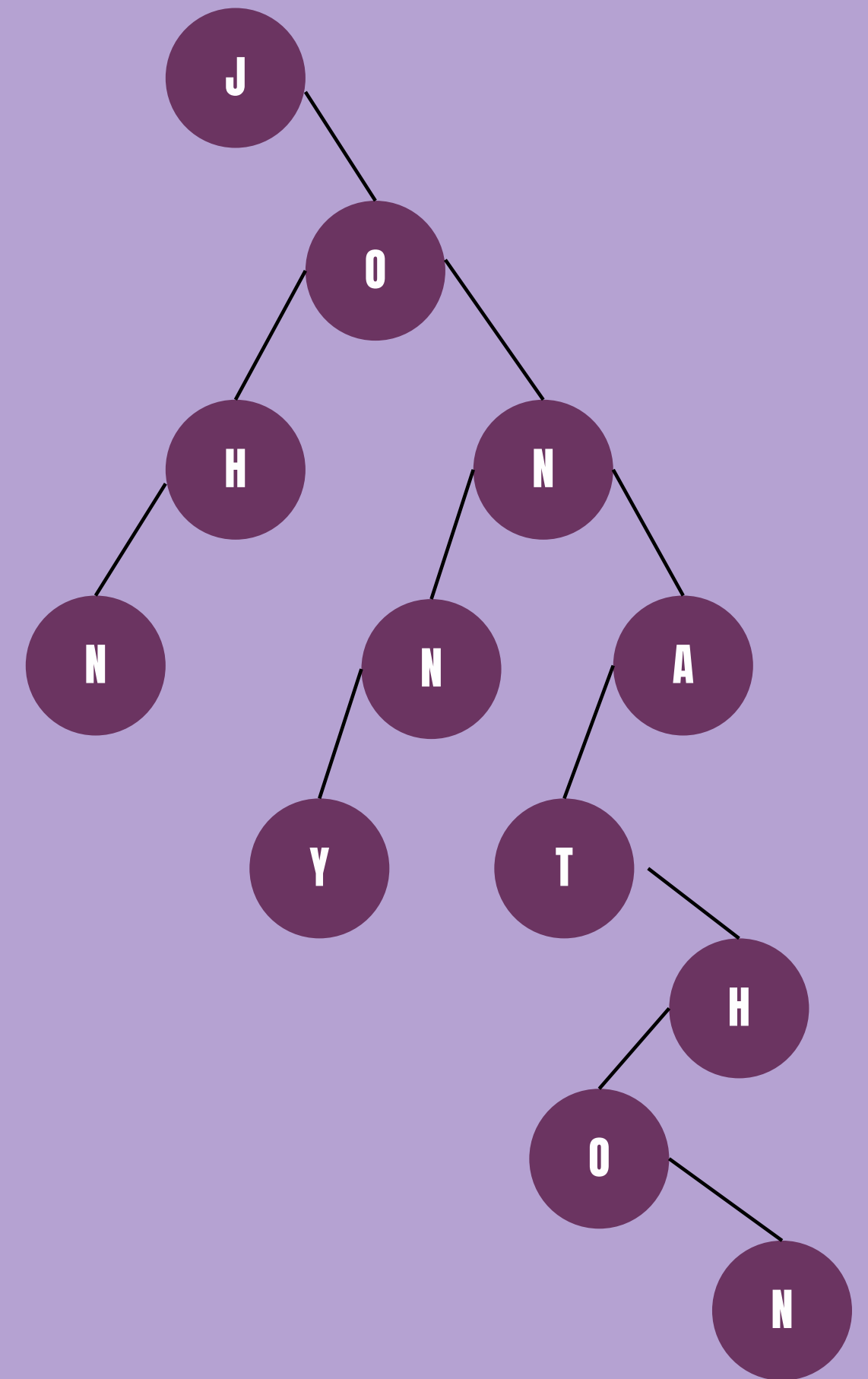
Contacts Managament using Tries

RHEA CHAINANI
22070126086
AIML-B

What are Tries ?

A TRIE (DERIVED FROM RETRIEVAL) IS A MULTIWAY TREE DATA STRUCTURE USED FOR STORING STRINGS OVER AN ALPHABET. IT IS USED TO STORE A LARGE AMOUNT OF STRINGS. THE PATTERN MATCHING CAN BE DONE EFFICIENTLY USING TRIES.

THE TRIE SHOWS WORDS LIKE JOHN, JONNY, JONATHON. THE IDEA IS THAT ALL STRINGS SHARING COMMON PREFIX SHOULD COME FROM A COMMON NODE.

Problem
Statement

DESIGN AND IMPLEMENT A CONTACTS MANAGEMENT SYSTEM USING THE TRIE DATA STRUCTURE IN C. THE SYSTEM AIMS TO EFFICIENTLY STORE, SEARCH FOR, UPDATE AND DELETE CONTACT INFORMATION WHILE PROVIDING A USER-FRIENDLY INTERFACE WITH CASE-INSENSITIVE SEARCH CAPABILITIES FOR AN IMPROVED USER EXPERIENCE.

Motivation

THE MOTIVATION BEHIND THIS PROJECT STEMS FROM THE NEED TO SIMPLIFY AND ENHANCE THE MANAGEMENT OF CONTACT INFORMATION IN A DIGITAL ERA. WITH THE INCREASING RELIANCE ON DIGITAL DEVICES, THERE IS A COMPELLING MOTIVATION TO DEVELOP A STREAMLINED AND EFFICIENT CONTACTS MANAGEMENT SYSTEM THAT CATERS TO THE USER'S NEED FOR QUICK ACCESS AND ORGANIZATION OF THEIR CONTACT DETAILS. THIS PROJECT IS DRIVEN BY THE DESIRE TO OFFER A PRACTICAL SOLUTION FOR USERS TO EFFICIENTLY STORE, SEARCH FOR, UPDATE AND DELETE THEIR CONTACT INFORMATION.

Objectives

- **EFFICIENT CONTACT MANAGEMENT**
- **USER-FRIENDLY INTERFACE**
- **CASE-INSENSITIVE SEARCH**
- **STREAMLINED CONTACT OPERATIONS**

Methodology of Implementation

Result Analysis & Outputs

```
1. Create a New Contact
2. Search For a Contact
3. Update a Contact
4. Delete a Contact
5. Exit
Please Enter Your Choice: 1
Enter the number of contacts to be updated in PhoneBook: 2
Enter the name of the Person: john
Enter the phone number (10 digits): 11111111111111
Invalid phone number format. Please enter a 10-digit number.
Enter the phone number (10 digits): 9988776655
Enter the email: john@gmail.com
Enter the name of the Person: JohnAthOn
Enter the phone number (10 digits): 4455224433
Enter the email: johnathon@yahoo.com
1. Create a New Contact
2. Search For a Contact
3. Update a Contact
4. Delete a Contact
5. Exit
Please Enter Your Choice: 2
Enter the name to be searched: jonathon
Contact not found.
1. Create a New Contact
2. Search For a Contact
3. Update a Contact
4. Delete a Contact
```

```
5. Exit
Please Enter Your Choice: 2
Enter the name to be searched: JOHN
Contact Found!
Name: JOHN
Phone Number: 9988776655
Email: john@gmail.com
1. Create a New Contact
2. Search For a Contact
3. Update a Contact
4. Delete a Contact
5. Exit
Please Enter Your Choice: 3
Enter the name to update: jonathon
Contact not found. Would you like to create it? (1/0): 1
Enter the phone number (10 digits): 2255334411
Enter the email: jonny@hitme.com
1. Create a New Contact
2. Search For a Contact
3. Update a Contact
4. Delete a Contact
5. Exit
Please Enter Your Choice: 3
Enter the name to update: john
Contact Found!
Name: john
Phone Number: 9988776655
```

```
Email: john@gmail.com
What would you like to update?
1. Phone Number
2. Email
3. Both
2
Enter the new email: john@email.com
Contact updated!
1. Create a New Contact
2. Search For a Contact
3. Update a Contact
4. Delete a Contact
5. Exit
Please Enter Your Choice: 4
Enter the name to delete: jonny
Contact not found. Deletion failed.
1. Create a New Contact
2. Search For a Contact
3. Update a Contact
4. Delete a Contact
5. Exit
Please Enter Your Choice: 4
Enter the name to delete: john
Contact deleted!
1. Create a New Contact
2. Search For a Contact
3. Update a Contact
```

Result Analysis & Outputs

```
4. Delete a Contact
5. Exit
Please Enter Your Choice: 2
Enter the name to be searched: john
Contact not found.
1. Create a New Contact
2. Search For a Contact
3. Update a Contact
4. Delete a Contact
5. Exit
Please Enter Your Choice: 2
Enter the name to be searched: johnathon
Contact Found!
Name: johnathon
Phone Number: 4455224433
Email: johnathon@yahoo.com
1. Create a New Contact
2. Search For a Contact
3. Update a Contact
4. Delete a Contact
5. Exit
Please Enter Your Choice: 5
Exiting program...
```

- THE SYSTEM IS CASE - INSENSITIVE.
- IT CHECKS THE VALIDITY OF A PHONE NUMBER BEFORE ADDING IT IN CONTACT DETAILS.
- BEFORE ADDING A CONTACT IT CHECKS IF IT ALREADY EXISTS AND IN THAT CASE GIVES THE USER AN OPTION TO UPDATE THE EXISTING CONTACT DETAILS INSTEAD.
- BEFORE UPDATING A CONTACT IT CHECKS WHETHER THE CONTACT EXISTS IN PHONE BOOK AND IF NOT, GIVES AN OPTION TO CREATE NEW CONTACT INSTEAD.
- DELETING A CONTACT DOES NOT HAVE ANY IMPACT
- ON OTHER CONTACTS WITH THE SAME PREFIX.

Symbol Table &
AVL Tree

TRIE, IN ESSENCE, ACTS AS A SYMBOL TABLE:
- KEY-VALUE ASSOCIATION
- EFFICIENT RETRIEVAL
- DYNAMIC DATA STRUCTURE
- SUPPORT FOR KEY-BASED OPERATIONS

Symbol Table & AVL Tree

AVL TREES ARE NOT USED AS:
- THE PRIMARY DATA, CONTACT NAMES, ARE STRINGS WITHOUT INHERENT SORTED ORDER.
- AVL TREES ARE MORE COMPLEX TO IMPLEMENT AND MAINTAIN.
- TRIES OFFER SPACE EFFICIENCY BY SHARING COMMON PREFIXES.

Learning
Outcomes

- DATA STRUCTURES PROFICIENCY
- INPUT VALIDATION AND ERROR HANDLING
- REAL-WORLD APPLICATION

Thank You