

# Vivekanand Education Society's Institute Of Technology Department Of Information Technology DSA mini Project A Y 2025-26

Title: Health Checker

Sustainability Goal: Advancing Health Awareness and Accessible

Medical Guidance

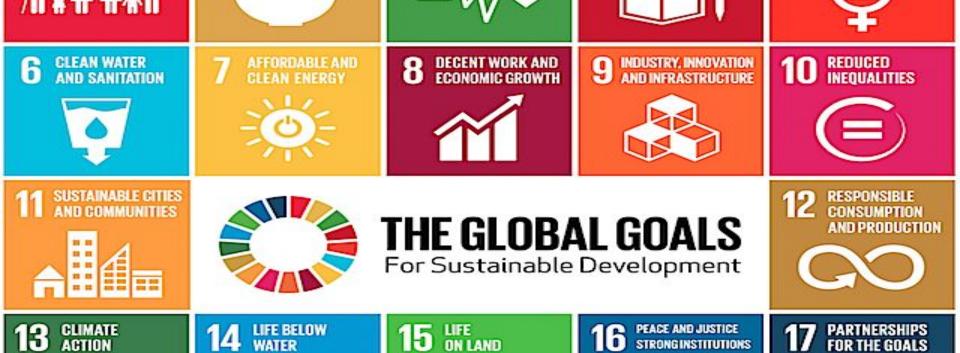
Domain: Data Structures & Algorithms

Member: Yash Rai

D10B

50

Mentor Name: Kajal Jewani



**GOOD HEALTH** 

AND WELL-BEING

NO

POVERTY

**ZERO** 

HUNGER

QUALITY EDUCATION

**STRONG INSTITUTIONS** 

GENDER

EQUALITY



### Content

- 1. Introduction to the Project
- 2. Problem Statement
- 3. Objectives of the Project
- 4. Scope of the Project
- 5. Requirements of the System (Hardware, Software)
- 6. ER Diagram of the Proposed System
- 7. Data Structure & Concepts Used
- 8. Algorithm Explanation
- 9. Time and Space Complexity
- 10. Front End
- 11. Implementation
- 12. Gantt Chart

- 13. Test Cases
- 14. Challenges and Solutions
- 15. Future Scope
- 16. Code
- 17. Output Screenshots
- 18. Conclusion
- 19. References (in IEEE Format)



## Introduction to Project

Health Checker is a healthcare technology project developed using C programming with a binary decision tree architecture. This system serves as a command-line based medical assistant that conducts preliminary health assessments through intelligent symptom analysis.

By guiding users through an interactive question-and-answer process, Health Checker evaluates symptoms for over fifteen common medical conditions while providing reliable medication recommendations and implementing a crucial severity-based triage system to support informed healthcare decisions.



## **Problem Statement**

In today's healthcare landscape, several critical issues exist:

- Limited Access to Healthcare:
  - Long waiting times at clinics for non-emergency cases
  - Healthcare costs for routine consultations
- > Information Overload:
  - Difficulty distinguishing between reliable and unreliable sources
  - Lack of structured guidance for symptom assessment
- Delayed Care:
  - People often delay seeking medical help due to uncertainty about severity

**Solution:** A systematic, tree-based diagnostic tool that provides reliable preliminary assessment, clear recommendations, and guidance on when professional medical care is necessary.



## Objectives of the project

- Implement binary tree for symptom assessment
- Provide preliminary health evaluation
- Offer treatment guidance (meds + home remedies)
- Determine urgency levels
- Health education & prevention tips
- Create user-friendly CLI interface



## Requirements of the system (Hardware, software)

#### Hardware:

- > Processor: Intel i3 or higher (Any modern processor)
- > RAM: Minimum 2GB (4GB recommended)
- Storage: 50MB free disk space

#### **Software:**

- ➤ OS: Windows 10/11
- > Compiler: GCC (GNU Compiler Collection) MinGW/TDM-GCC for Windows
- Terminal: CMD/PowerShell



## Front End

Health Checker provides a simple, user-friendly and intuitive CLI interface which features:

- ➤ Interactive Q&A Flow Guided yes/no questions about symptoms
- Color-coded Severity Levels Emergency (red), Urgent (yellow), Moderate (cyan), Mild (green)
- > Comprehensive Diagnosis Reports including:
  - Condition description and severity assessment
  - Home remedies.
  - Specific medication guidance with dosages
  - When to seek professional medical help
- Multiple Health Categories covering: Respiratory issues, Digestive problems, Headaches and pain, General wellness, etc



## **Implementation**

- Uses binary decision tree to navigate symptom assessment in C language.
- Tree nodes contain:
  - Question nodes for symptom interrogation
  - Diagnosis nodes for final condition assessment
- Tree Architecture:
  - Root Node: Starts with emergency symptom detection (chest pain, breathing difficulty)
  - Internal Nodes: Question nodes that branch based on Yes/No responses
  - <u>Leaf Nodes</u>: Diagnosis nodes that provide final assessment
- Dynamic memory allocation using malloc and free functions in tree management.



## **Gantt Chart**

Task	Week 1	Week 2	Week 3	Week 4
Project Planning	<b>✓</b>			
Data Structure Design	<b>~</b>			
Core Logic (Tree)		<		
Traversal Algorithm				
UI/CLI Implementation			<b>\</b>	
Testing & Debugging			~	
Documentation & Report				<b>&gt;</b>



## Conclusion

- Successfully implemented a binary tree-based health diagnostic system
- > Provided specific medication recommendations with dosages
- Applied DSA concepts to solve real-world healthcare problems

#### **➤** Impacts:

- Provides preliminary health assessment without immediate doctor visit
- Helps users make informed decisions about seeking medical care
- Reduces unnecessary emergency room visits for minor conditions
- Educates users about common health conditions



## References

- > The C Programming Language Brian W. Kernighan & Dennis M. Ritchie
- C Programming: A Modern Approach K. N. King
- Data Structures Using C Aaron M. Tenenbaum
- > Programming in ANSI C E. Balagurusamy
- C Programming Tutorial GeeksForGeeks