# Tournament Tree Simulation Report

## 1. Introduction

A tournament tree is a binary tree used to determine the winner among participants through successive rounds of pairwise comparisons. It is commonly applied in game tournaments, job scheduling, and certain algorithmic problems.

## 2. Purpose of the Program

The C program simulates a tournament tree where each node represents a match between two players characterized by skill levels. The winner of each match is determined by comparing skill levels, and the simulation continues until one winner remains.

## 3. Code Functionality

The program performs the following tasks:  
- Builds an initial level of the tournament tree from an array of player skill levels.  
- Constructs upper levels iteratively by comparing player pairs and assigning winners to parent nodes.  
- Uses dynamic memory allocation and linked structures for flexible node creation.  
- Displays each round's matches and identifies the final winner.

## 4. Advantages

- Efficient simulation of knockout-style tournaments.  
- Allows for visualization of each round's outcome.  
- Modular code structure enables easy customization.  
- Can be extended for multi-dimensional skill evaluation or team-based comparisons.

## 5. Applications

This program can be used in:  
- Game and AI tournaments to determine dominant strategies.  
- Educational simulations for teaching recursion and tree structures.  
- Backend logic for elimination-based ranking systems.

## 6. Conclusion

The tournament tree model implemented in C provides a structured and visual approach to simulate competitive elimination. Its extendibility and clarity make it an effective tool for educational, experimental, and practical applications.