1. **PROJECT TITLE**

“Company Scheme Simulation”

1. **PROJECT DESCRIPTION**
2. **Overview**

This project simulates a company management system where the CEO manages employee hiring, firing, and organizational structure using a binary tree. It tracks employee data, commissions, and downlines, with features for data persistence and an interactive command-line interface. The system provides a simplified, engaging way to explore business operations.

1. **Objective**

The project aims to address the following problems:

• Simplifies the process of hiring, firing, and viewing employee data.

• Automates updates to hierarchical details like downline counts and commissions.

• Ensures quick and efficient data storage, retrieval, and modification.

• Provides long-term data storage and integrity through file handling.

• Tracks employee performance metrics and financial incentives dynamically.

1. **Scope**

This project focuses on simulating basic company management, particularly employee operations, using a binary tree structure.

• Manages employee hiring, firing, roles, IDs, commissions, and downlines.

• Represents the organizational hierarchy using a binary tree.

• Automates commission calculations based on downline data.

• Ensures data continuity through file handling.

• Provides a text-based interface for user interaction. Limitations:

• Focuses only on employee hierarchy and basic data (name, age, position, ID).

• Supports single-user interaction (CEO only).

• Commission system is fixed and non-customizable.

• Uses a binary search tree, limiting it to binary hierarchical structures.

• Data is completely wiped when the company is demolished, with no recovery option.

1. **FEATURES**
   1. **Check Employees:** Displays the company’s entire workforce in a hierarchical order resembling a binary tree structure.
   2. **Hire Employee:** Allows user to facilitate the hiring process to add new employees in the organization.
   3. **Fire Employee:** Intended for offboarding employees with proper send off and rebate of contributed finance.
   4. **Save File:** This feature is designed to store and manage all the names entered into the system, serving as a central repository for maintaining accurate and organized records.
   5. **See Pocket Money:** Shows the money calculated and pocketed from total hired employees.
   6. **Demolish Company:** a critical and irreversible function designed to completely erase all data and structures associated with the company within the system. This feature ensures that no trace of the company's information remains, effectively resetting the system as if the company never existed.
   7. **Clock Out:** designed to enable users to officially exit or close the program. It marks the conclusion of a user's session or interaction with the application, ensuring that essential tasks like saving data, releasing resources, or recording activity logs are completed before the program shuts down.
2. **TECHNOLOGIES USED**
   1. Programming Language/s: C
   2. Tools: VSCode, GitHub, Copilot AI, ChatGPT
   3. Databases: Notepad
3. **PROJECT STRUCTURE**

**/scheme-company-tycoon**

**/project**

**├── src/ # Source code folder**

**│ ├── main.c # Main entry point of the program**

**│**

**├── data/ # Data files and assets**

**│ └── EmployeeData.txt # Stores employee-related data.**

**│ └── Asset.txt # Stores user and company related data.**

**│**

**├── docs/ # Documentation**

**│ └── README.md # Project description**

**│└── README.md # Project description**

**│**

**├── build/ # Compiled files (add to .gitignore)**

**│ └── main.exe # Compiled executable file**

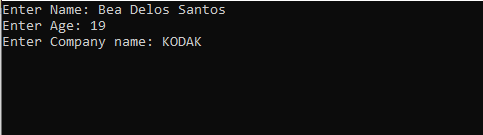
1. **USAGE**

* **Start the Program**

When you run the compiled program, the first thing it does is check for the existence and content of the files EmployeeData.txt and Asset.txt.

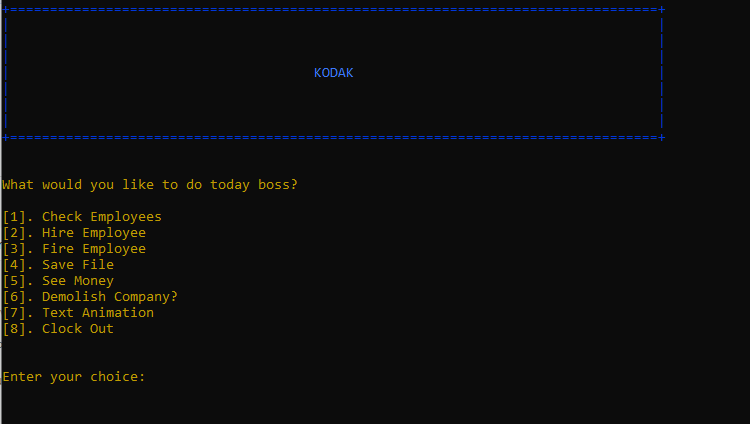
**If the files are empty:**

The program will prompt you to enter the CEO's name, age, and the company name.



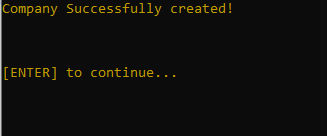
**If the files are not empty:**

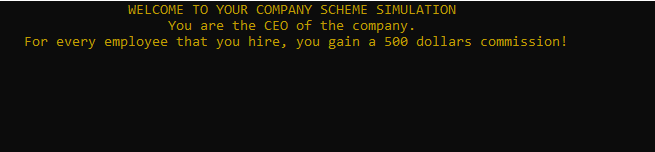
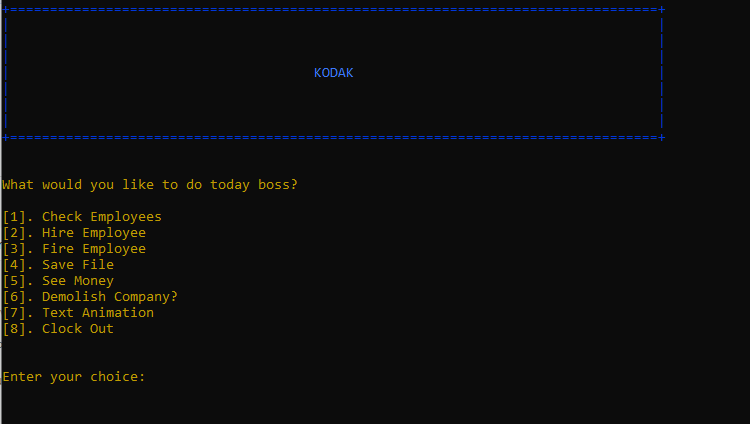
The program will load the existing data from EmployeeData.txt and Asset.txt, reconstructing the company's state from the saved information. It will then directly proceed to the main menu.

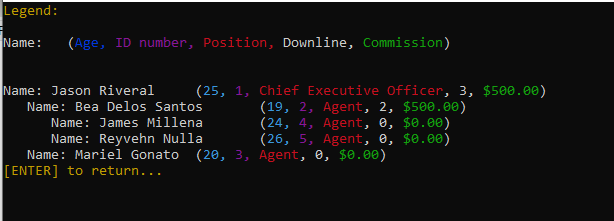


After you provide the CEO's name, age, and company name, the display will refresh. A short sequence of messages will then appear, each briefly showing the progress of the company's setup. These messages—such as "Finalizing contracts," "Building Company" and "Booting Assistant". Once the setup messages are finished, the program displays a confirmation message.

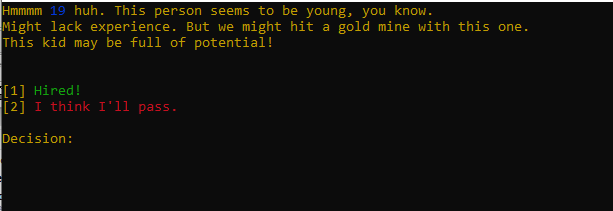
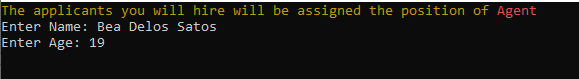
* **Confirmation Message:** A success message appears, confirming the completion of the setup phase. This message typically indicates that the company has been successfully created.

The user needs to press the Enter key to proceed.

* **Prompt to Continue:** Immediately after the confirmation message, a prompt will appear, instructing the user to press the Enter key to proceed. This pause allows the user to acknowledge the successful setup before moving to the next stage of the program.
* **Main Menu:** Once the user presses the Enter key, the screen will clear (or refresh), and the main menu of the company simulation will be displayed.

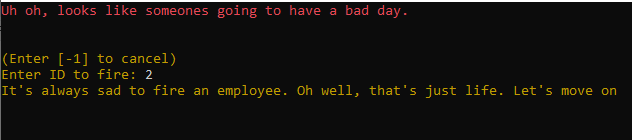
**Menu 1 – Check Employees:** "Check Employees" shows the hierarchical structure of your employees

**Menu 2 - Hire Employee:** If you choose "Hire Employee," you'll be asked for the name and age. The system will give you feedback based on the age (e.g., comments about experience or potential). You then choose whether to hire them or not,

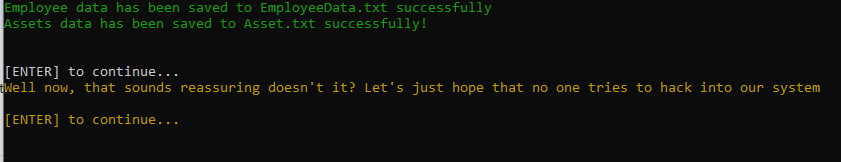
****

****

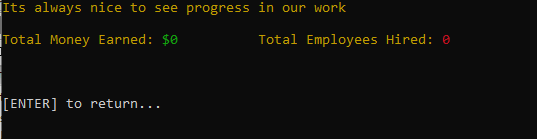
**Menu 3: Fire Employee:** If you select "Fire Employee," enter the ID of the employee you want to remove. The program confirms the action and removes the employee.

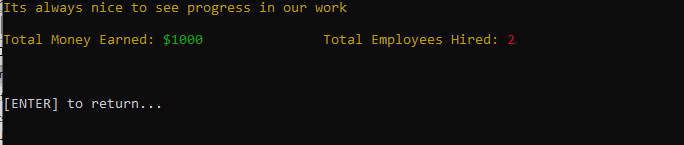


**Menu 4 – Save File:** A simple confirmation message indicating successful saving.

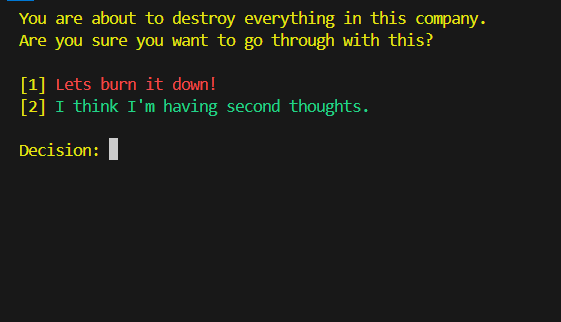
****

**Menu 5 - See Money:** Displays the total CEO commission and the number of employees hired.

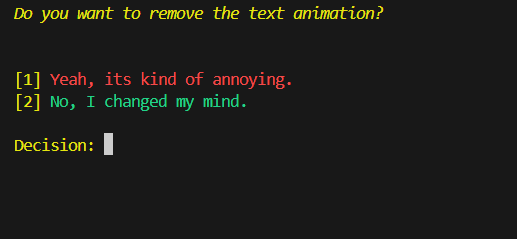




**Menu 6 - Demolish Company?:** This menu displays a confirmation prompt before deleting all data.



**Menu 7 - Text Animation:** This menu provides the option to toggle the text animation on or off.

****

1. **ACKNOWLEDGEMENTS**

*We would like to give appreciation and acknowledgement to the various sources and contributors throughout the development of this project:*

**Sources\Webites:**

* **Stack Overflow** – For aiding with debugging and language setup.
* **Reddit** – programming community helped with various coding problems.
* **Quora** – Plethora of programming related content aiding the development of the project.
* **YouTube** – Useful tutorials and teaching materials.
* **GeekForGeeks** – Fundamentals and basic structures in C language.

**Contributing Tools:**

* **ChatGPT** – Generating and brainstorming features and ideas.
* **CoPilot** – Debugging.

**Project Contributors**

* Riveral, Lord Jason
* Delos Santos, Bea
* Millena, James
* Batusbatusan, Romel
* Gonato, Mariel
* Nulla, Reyvehn

*We would like to voice out our deep appreciation to the unwavering dedication of our professor, Mr. Kenneth Roi Novabos. Your passion for teaching and commitment to our education are truly inspiring. You continuously put in extra effort to help us understand the lessons and develop the skills we need to thrive.*

1. **CONTACT INFORMATION**Name: Lord Jason G. Riveral

Email: riverallordjason@gmail.com

Contact No.: 09183613374

Name: Bea T. Delos Santos

Email: beadelossantos203@gmail.com

Contact No.: 09935988086

Name: Romel Batusbatusan

Email: romelbatusbatusan@gmail.com

Contact no.: 09484260240

Name: Reyvehn Nulla

Email: reyvehnalegado@gmail.com

Contact no.: 09568773566

Name: Mariel Gonato

Email: gonatomariel12@gmail.com

Contact no.: 09939828926

Name: James Millena

Email: jamesmillena05@gmail.com

Contact no.: 09949070418