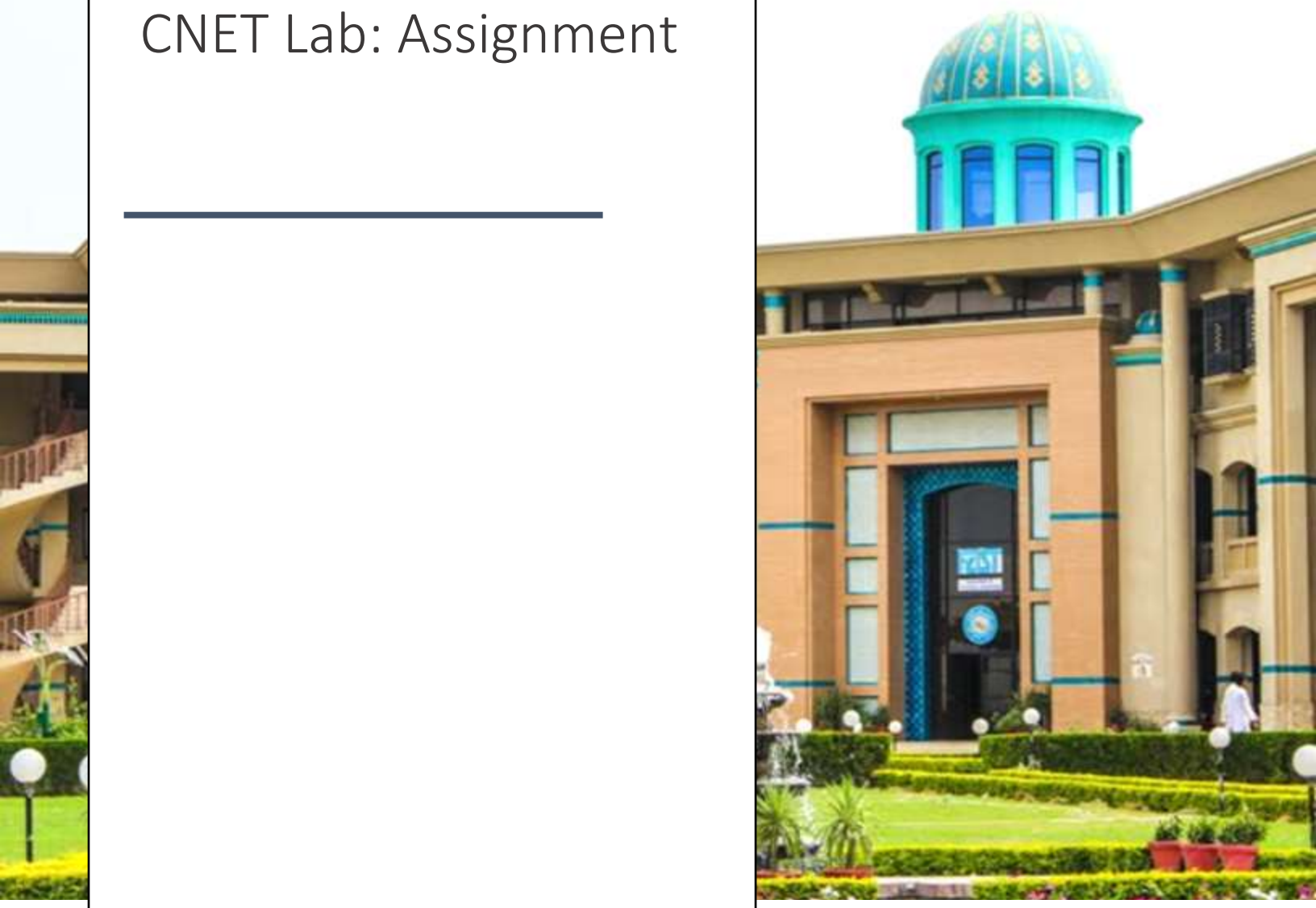
CNET Lab: Assignment



MAY 14, 2023

Daniyal Khan : 20i-1847

Report

Contents

1. Header Files:

The code includes several standard C header files necessary for socket communication, file operations, process handling, and string manipulation.

2. Macro Definitions:

- `PORT`: Specifies the port number on which the server will listen for incoming connections.
- `BUFFER_SIZE`: Defines the size of the buffer used for sending and receiving data.
- `DATABASE_PATH`: Specifies the path where student information files will be stored.

3. Function Declarations:

The code declares several functions used for handling different types of requests:

- `handleAdmin()`: Handles admin requests, such as adding, editing, and viewing student information.
- `handleTeacher()`: Handles teacher requests, including creating and performing operations on a teacher's file.

- `handleStudent()`: Handles student requests.
- `addStudentInfo()`: Adds student information to the database.
- `editStudentInfo()`: Edits student information in the database.
- `viewStudentInfo()`: Views student information from the database.
- `calculatePercentage()`: Calculates the percentage and writes it to a file.

4. `main()` Function:

The main function serves as the entry point of the program. It performs the following steps:

- Initializes variables and data structures required for socket communication.
- Creates a socket using the `socket()` function.
- Binds the socket to the specified port using the `bind()` function.
- Listens for incoming connections using the `listen()` function.
- Enters a loop to accept client connections and fork child processes to handle each client.
- In the child process, based on the received user type (admin, teacher, or student), it calls the corresponding handling functions.
- In the parent process, it waits for child processes to finish before accepting new connections.
- Closes the server socket when the program exits.

5. `handleAdmin()` Function:

This function handles admin requests within an infinite loop until an invalid request is received. It receives an admin request from the client and performs the requested operation, such as adding, editing, or viewing student information.

6. `handleTeacher()` Function:

This function handles teacher requests. It checks if the teacher file exists and, if not, asks the teacher if they want to create it. If the teacher confirms, the file is created. Otherwise, the operation is canceled.

It's important to note that the code is incomplete, as some function implementations are missing. However, it is a simple server-client program that allows admins, teachers, and students to interact with the server to manage student information.