ONLINE EXAMINATION SYSTEM – MAJOR PROJECT REPORT (C PROJECT)

Course Code: CSEG1041

Name: Tammineni Nikhil

SAP ID: *590027800*
University of Petroleum and Energy Studies

---

1. Title Page

Title: Online Examination System in C
Course: C Programming (CSEG1041)
Submitted to: Dr. Prashant Trivedi
Submitted by: Thammineni Nikhil, 590027800
Semester: 1st semester
Date: 01/12/2025

---

2. Abstract

This project presents a simple **Online Examination System** implemented in the C programming language.
The system allows users to **register**, **log in**, and **attempt a multiple-choice examination**.
All login details are stored in a text file, and the code is divided into multiple modules for readability and reusability.

The objective of the project is to demonstrate understanding of **modular programming**, **file handling**, **functions**, **header files**, and **basic control structures** in C.
The entire program follows a clean, easy-to-understand structure suitable for beginners.

---

3. Problem Definition

The purpose of this project is to design a simple **command-line-based examination system** that:

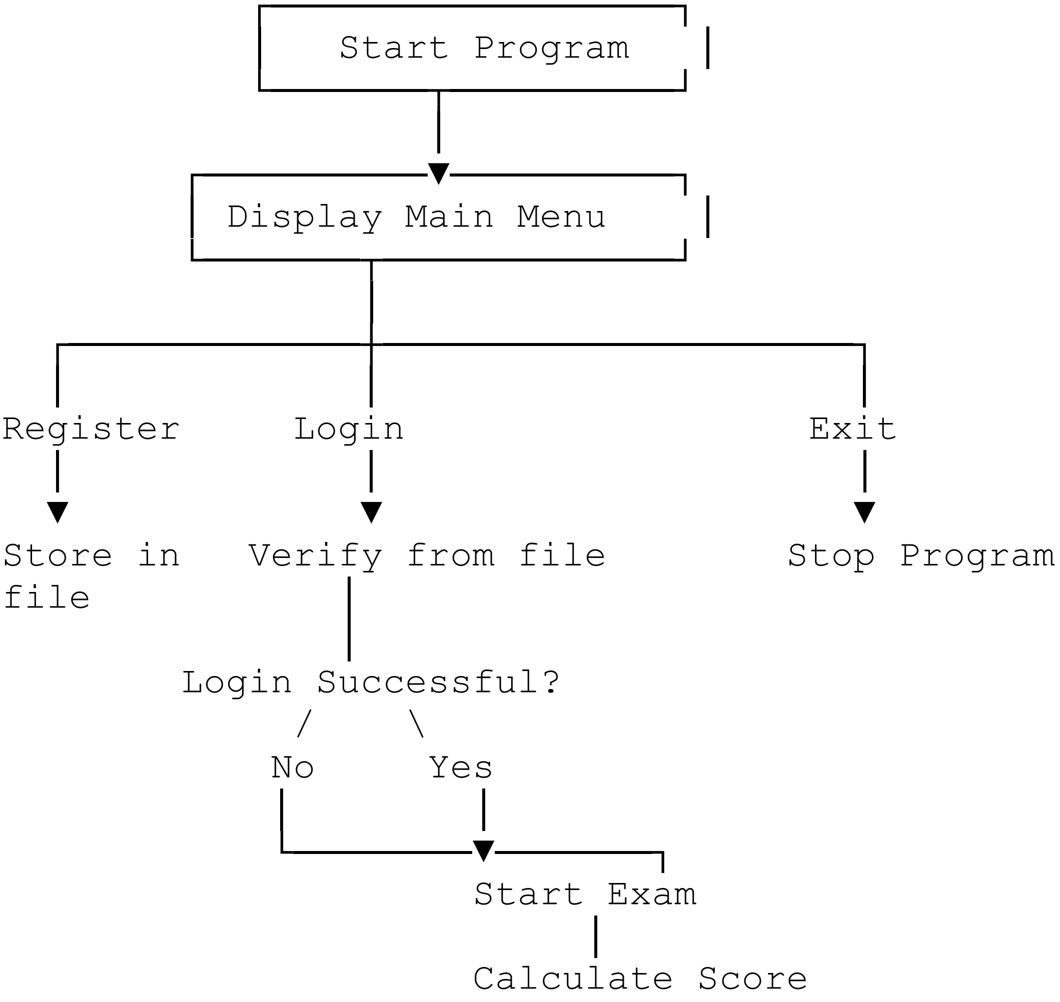
- Allows new users to register with a username and password
- Stores login credentials securely in a file
- Allows existing users to log in
- Conducts a 5-question MCQ exam
- Calculates and displays the user's score
- Uses C programming concepts such as functions, file handling, conditionals, loops, and modularity

The problem focuses on building a functional system that mimics a real online test but in a simple text-based interface.

---

4. System Design

4.1 Flowchart



```
        |
  Display Result
        |
   Go to Menu
```

4.2 Algorithm

Main Menu Algorithm

1. Start program
2. Display menu
3. If user selects Register → call `registerUser()`
4. If user selects Login → call `loginUser()`
5. If login is successful → call `startExam()`
6. Display score
7. Loop back to menu until Exit is chosen

Registration Algorithm

1. Ask for username
2. Ask for password
3. Open file `users.txt`
4. Store credentials in file
5. Close file

Login Algorithm

1. Read username & password
2. Open `users.txt`
3. Compare input with each line
4. If matched → return success
5. Else → return failure

Exam Algorithm

1. Display 5 MCQs
2. For each question:
   - Get answer
   - Compare with correct answer
   - Increment score
3. Return score

---

5. Implementation Details

The project uses **modular C programming**.
The source code is divided into multiple `.c` and `.h` files.

5.1 File Structure

```
main.c
exam.c        exam.h
user.c        user.h
utils.c       utils.h
users.txt (auto-created)
```

5.2 Explanation of Modules

main.c

- Controls the menu
- Calls login, register, and exam functions
- Uses `switch` for choices

user.c / user.h

- Handles registration
- Handles login
- Uses file handling (`fopen`, `fprintf`, `fscanf`)

exam.c / exam.h

- Contains 5 MCQ questions
- Counts correct answers

utils.c / utils.h

- Contains screen clear function (`system("cls || clear")`)

---

6. Testing & Results

Test Case 1: Register New User

Input:
username = "john"
password = "1234"

Expected Output:

```
Registration Successful!
```

Result: Pass ✓

Test Case 2: Login with Correct Credentials

Input:
username = john
password = 1234

Expected Output:

```
Login Successful!
```

Result: Pass ✓

Test Case 3: Login with Wrong Credentials

Expected Output:

```
Invalid username or password
```

Result: Pass ✓

Test Case 4: Exam Attempt

Answered all correctly:

Expected Output:

```
Your Score = 5 / 5
```

Result: Pass ✓

---

7. Conclusion & Future Work

Conclusion

This Online Examination System demonstrates the use of C programming fundamentals like **functions, loops, conditionals, file handling, and modularity**.
The system successfully performs registration, login, and conducting a simple multiple-choice exam.

Future Enhancements

The system can be improved by adding:

- Admin panel for adding questions
- Random question generation
- Storing scores in a separate file
- Negative marking
- Timer-based exam
- Graphical user interface

---

8. References

- UPES C Programming Lecture Notes
- ANSI C Programming Language – Kernighan & Ritchie
- GCC Compiler Documentation
- Class materials and slides

---

9. Appendix

- Complete code (already uploaded to GitHub)
- Screenshots of output
- Commit history