**NBKR INSTITUTE OF SCIENCE AND TECHNOLOGY**

**COLLEGE ADMISSION PORTAL**

**COURSE:** DATA STRUCTURES

**DEPARTMENT:** COMPUTER SCIENCE

**SECTION:** D

**YEAR:** 1

**SEMESTER:** 1

**DATE:** MAY 01 2025

**SUBMITTED TO:** ASHOK SELVA KUMAR E

**SUBMITTED BY:**

1. K.K. SAI NEERAJ 1 11(24KB1A05U5)11
2. SK. AKMAL 1(24KB1A05HA)11
3. T. HEMANTH 1 1(24KB1A05KV)1
4. T. SAPTHAGIRI 1(24KB1A05LP)1

# 1. Acknowledgments

I would like to express my sincere gratitude to my mentor, Ashok Kumar, for their valuable guidance and support throughout the development of this project. I also thank my faculty, friends, and family who provided encouragement and constructive feedback.

# 2. Abstract

This project is a simple College Admission System developed in C. It allows users to register student details and assign them to different departments. The data is saved to a file for future reference. The system uses a menu-driven interface for interaction and handles basic file operations to store student records.

# 3. Introduction

This project focuses on simplifying and automating the student admission process. Traditionally, college admissions are carried out manually, leading to human errors and data mismanagement. To tackle these issues, I chose to create this project, aiming to streamline the admission procedure through a structured software approach.

# 4. Objectives

- Develop a system for registering student details.

- Display a list of available departments.

- Save student information persistently in a file.

- Provide a user-friendly and menu-driven interface.

# 5. System Requirements

Hardware Requirements:

- Processor: Intel Pentium or higher

- RAM: Minimum 1GB

- Storage: 50MB

Software Requirements:

- Operating System: Windows/Linux

- Compiler: GCC or Turbo C

- Editor: Code::Blocks, Dev-C++, or any C IDE

# 6. Methodology

1. Defined a structure to store student information.

2. Created a menu-based system for user interaction.

3. Added functions to display departments, register students, and save data to a file.

4. Used file handling for persistent data storage.

5. Tested the program with various inputs.

# 7. Project Description

Problem Statement:  
Manual admission processes are prone to inefficiency, data loss, and inaccuracies.

Proposed Solution:  
Automate the admission process using a simple C program that registers and saves student information.

Key Features:

- Register students

- View departments

- Save student data in a text file

# 8. Algorithm

1. Start the program.

2. Display the main menu.

3. If the user selects 'Register Student':

- Prompt for name and ID

- Show department list and get selection

- Save the entered details to a file

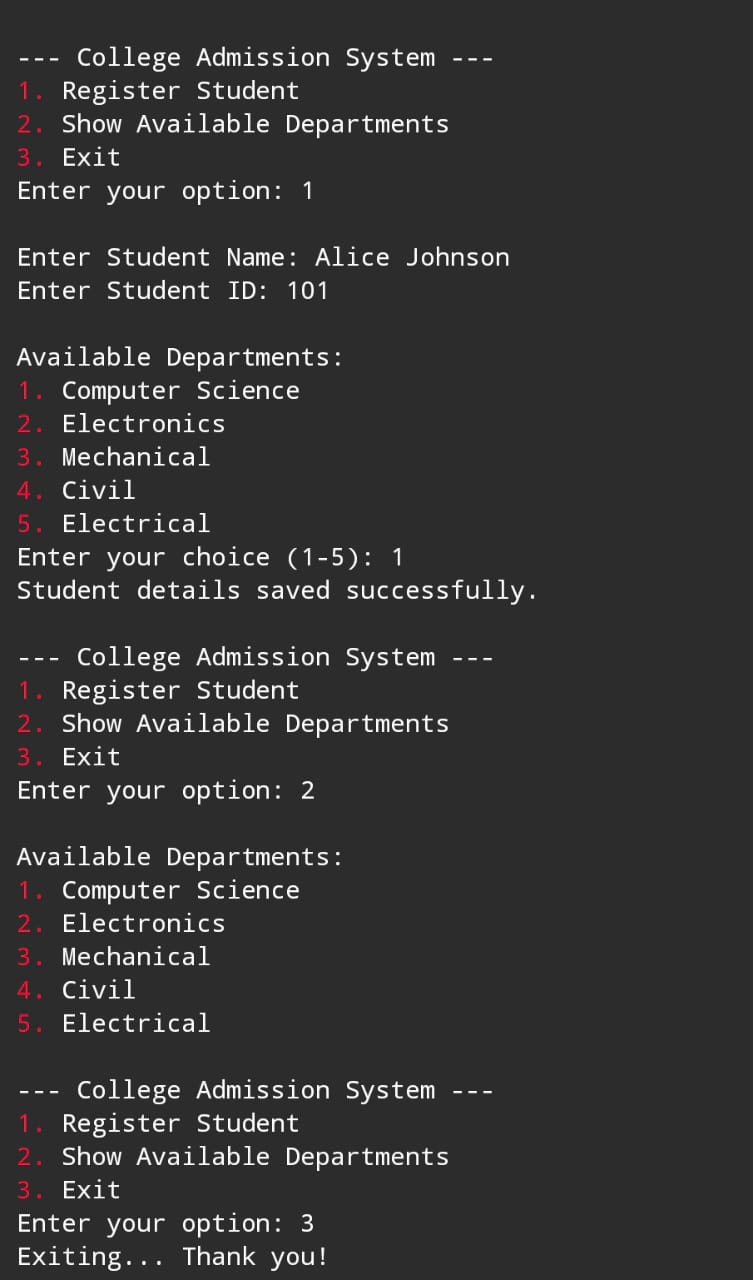
4. If the user selects 'Show Departments', display the list

5. If the user selects 'Exit', terminate the program

# 19. Program Code

#include <stdio.h>  
#include <stdlib.h>  
#include <string.h>  
  
struct Student {  
 char name[50];  
 int id;  
 char department[50];  
};  
  
void showDepartments() {  
 printf("\nAvailable Departments:\n");  
 printf("1. Computer Science\n");  
 printf("2. Electronics\n");  
 printf("3. Mechanical\n");  
 printf("4. Civil\n");  
 printf("5. Electrical\n");  
}  
  
void saveToFile(struct Student s) {  
 FILE \*file = fopen("admission\_list.txt", "a");  
 if (file == NULL) {  
 printf("Error opening file!\n");  
 return;  
 }  
 fprintf(file, "Name: %s | ID: %d | Department: %s\n", s.name, s.id, s.department);  
 fclose(file);  
 printf("Student details saved successfully.\n");  
}  
  
void registerStudent() {  
 struct Student s;  
 int choice;  
   
 printf("\nEnter Student Name: ");  
 scanf(" %[^  
]", s.name);  
 printf("Enter Student ID: ");  
 scanf("%d", &s.id);  
   
 showDepartments();  
 printf("Enter your choice (1-5): ");  
 scanf("%d", &choice);  
   
 switch (choice) {  
 case 1: strcpy(s.department, "Computer Science"); break;  
 case 2: strcpy(s.department, "Electronics"); break;  
 case 3: strcpy(s.department, "Mechanical"); break;  
 case 4: strcpy(s.department, "Civil"); break;  
 case 5: strcpy(s.department, "Electrical"); break;  
 default: strcpy(s.department, "Unknown"); break;  
 }  
   
 saveToFile(s);  
}  
  
int main() {  
 int option;  
   
 do {  
 printf("\n--- College Admission System ---\n");  
 printf("1. Register Student\n");  
 printf("2. Show Available Departments\n");  
 printf("3. Exit\n");  
 printf("Enter your option: ");  
 scanf("%d", &option);  
   
 switch (option) {  
 case 1:  
 registerStudent();  
 break;  
 case 2:  
 showDepartments();  
 break;  
 case 3:  
 printf("Exiting... Thank you!\n");  
 break;  
 default:  
 printf("Invalid option! Please try again.\n");  
 }  
 } while (option != 3);  
   
 return 0;  
}

# 10. Output Screenshot

\*(Please paste a screenshot of the console output here, e.g., showing the menu, registration, and confirmation message.)\* 

# 11. Testing and Validation

Sample Input:

Name: John Doe  
ID: 101  
Department: 1 (Computer Science)

Sample Output:

Student details saved successfully.

File Output in 'admission\_list.txt':  
Name: John Doe | ID: 101 | Department: Computer Science

# 12. Limitations

- No data validation (e.g., duplicate IDs or invalid names).

- Department choice outside 1-5 results in 'Unknown'.

- No GUI, only command-line interface.

- Cannot view registered students within the program.

# 13. Future Enhancements

- Add feature to view all registered students.

- Include student editing and deletion functionality.

- Validate inputs for errors or duplicates.

- Implement a graphical user interface.

- Use a database instead of a text file.

# 14. Conclusion

This project has helped me understand file handling, structures, and modular programming in C. I’ve learned how to collect input, store data, and interact with users through menus. This system can be a foundation for larger educational software systems.

# 15. References

- 'Let Us C' by Yashavant Kanetkar

- TutorialsPoint - C Programming

- GeeksforGeeks - File Handling in C

- https://cplusplus.com

# 16. Appendix

Refer to Section 10 for the full source code.