

# STUDENT MARKS SYSTEM PROJECT REPORT

Made by: Shabd Kheterpal

UPES, School of Computer Science

B.Tech CSE, Year 1

-----

## ABSTRACT

This project is a basic C program created as part of the first-year B.Tech CSE curriculum.

The aim is to build a simple Student Marks System that can store multiple students, calculate totals and averages, search by SAP ID, and save data to a text file.

-----

## PROBLEM DEFINITION

The objective is to develop a C program that:

1. Takes student details such as name, SAP ID, phone and email.
2. Allows entering marks of up to seven subjects.
3. Stores multiple students.
4. Provides search functionality based on SAP ID.
5. Saves all data to a text file for record-keeping.

-----

## SYSTEM DESIGN

The system follows a modular approach divided into four .c files:

1. main.c – controls program flow and menu logic.
2. student.c – handles storing, displaying, searching, and file saving.
3. menu.c – prints menu and processes user choices.
4. utils.c – displays banner and basic UI text.

Flow:

User → Menu → Enter Student / Search / Save → Output

-----

## IMPLEMENTATION DETAILS

The program uses:

- Struct for storing student data.
- Arrays to store multiple students.
- Functions for modularity.
- File handling to save student details.
- Basic string and input handling.
- Simple loops for marks and student iteration.

Each student entry records:

Name, SAP ID, Phone, Email, Marks (max 7), Total, Average, Grade.

-----

## TESTING & RESULTS

Test cases performed:

1. Entering first student → Data stored successfully.
2. Adding multiple students → Works correctly.
3. Searching by SAP ID → Accurate retrieval.
4. Saving to students.txt → File created with correct formatting.
5. Invalid inputs (marks >100) → Handled by resetting values.

Results: Program runs successfully in terminal without errors.

-----

## CONCLUSION

The Student Marks System fulfills all requirements of a basic data-oriented C project.

As a first-year B.Tech CSE project, it demonstrates understanding of structures, functions, loops, conditionals, and file handling. Future improvements can include editing student data, deleting records, and a better UI.

-----

END OF REPORT