Problem 6: Attendance Management System

Specifications:

Variables: Student name, roll number, and attendance.

Static & Const: Static variable for total students; const for total classes.

Switch Case: Menu for marking, viewing, and calculating attendance.

Looping Statements: Loop through students to mark attendance.

Pointers: Pointer for dynamically updating attendance records.

Functions: Separate functions for marking and viewing attendance.

Arrays: Store attendance records.

Structures: Structure for student details.

Nested Structures: Nested structures for student personal and attendance details.

Unions: Union for different types of leave.

Nested Unions: Nested union for categorizing leave types.

Output Expectations: Display attendance records and summaries.

Menu Example:

1. Mark Attendance

2. View Attendance

3. Calculate Attendance Percentage

4. Exit

```
#include <stdio.h>
#include <stdlib.h>

#define TOTAL_CLASSES 30

union LeaveType {
    int sickLeave;
    int casualLeave;
    int paidLeave;
};

union LeaveCategory {
    union LeaveType regularLeave;
    union LeaveType emergencyLeave;
};

struct Student {
    char name[50];
    int rollNumber;
    int attendance[TOTAL_CLASSES];
    union LeaveCategory leaveCategory;
};

void markAttendance(struct Student* students, int totalStudents) {
    int rollNo, classNum, status;
```

```
    printf("Enter roll number: ");
    scanf("%d", &rollNo);

    for (int i = 0; i < totalStudents; i++) {
        if (students[i].rollNumber == rollNo) {
            printf("Enter class number (1 to %d): ", TOTAL_CLASSES);
            scanf("%d", &classNum);
            if (classNum < 1 || classNum > TOTAL_CLASSES) {
                printf("Invalid class number!\n");
                return;
            }
            printf("Enter attendance (1 for present, 0 for absent): ");
            scanf("%d", &status);
            students[i].attendance[classNum - 1] = status;
            printf("Attendance marked successfully!\n");
            return;
        }
    }
    printf("Student with roll number %d not found.\n", rollNo);
}

void viewAttendance(struct Student* students, int totalStudents) {
    int rollNo;
    printf("Enter roll number: ");
    scanf("%d", &rollNo);
```

```
    for (int i = 0; i < totalStudents; i++) {
        if (students[i].rollNumber == rollNo) {
            printf("Attendance record for %s (Roll No: %d):\n", students[i].name,
students[i].rollNumber);
            for (int j = 0; j < TOTAL_CLASSES; j++) {
                printf("Class %d: %s\n", j + 1, students[i].attendance[j] ? "Present" : "Absent");
            }
            return;
        }
    }
    printf("Student with roll number %d not found.\n", rollNo);
}


void calculateAttendance(struct Student* students, int totalStudents) {
    int rollNo;
    printf("Enter roll number: ");
    scanf("%d", &rollNo);


    for (int i = 0; i < totalStudents; i++) {
        if (students[i].rollNumber == rollNo) {
            int presentCount = 0;
            for (int j = 0; j < TOTAL_CLASSES; j++) {
                if (students[i].attendance[j] == 1) {
                    presentCount++;
                }
```

```
            }
            float percentage = ((float)presentCount / TOTAL_CLASSES) * 100;
            printf("Attendance percentage for %s (Roll No: %d): %.2f%%\n", students[i].name,
students[i].rollNumber, percentage);
            return;
        }
    }
    printf("Student with roll number %d not found.\n", rollNo);
}

int main() {
    int totalStudents;
    printf("Enter total number of students: ");
    scanf("%d", &totalStudents);

    struct Student* students = (struct Student*)malloc(totalStudents * sizeof(struct Student));



    for (int i = 0; i < totalStudents; i++) {
        printf("\nEnter details for student %d\n", i + 1);
        printf("Enter name: ");
        scanf("%s", students[i].name);
        printf("Enter roll number: ");
        scanf("%d", &students[i].rollNumber);

        for (int j = 0; j < TOTAL_CLASSES; j++) {
            students[i].attendance[j] = 0;
        }
```

```
}

int choice;
do {
    printf("\nMenu:\n");
    printf("1. Mark Attendance\n");
    printf("2. View Attendance\n");
    printf("3. Calculate Attendance Percentage\n");
    printf("4. Exit\n");
    printf("Enter your choice: ");
    scanf("%d", &choice);


    switch (choice) {
        case 1:
            markAttendance(students, totalStudents);
            break;
        case 2:
            viewAttendance(students, totalStudents);
            break;
        case 3:
            calculateAttendance(students, totalStudents);
            break;
        case 4:
            printf("Exiting program...\n");
            break;
        default:
            printf("Invalid choice! Please try again.\n");
```

```
        }
    } while (choice != 4);

    free(students);
    return 0;
}
```