Roll NO:27

NAME : Mali Anjali Prakash

DIV: FYMCA-A

Assignment no -9

#include <stdio.h>

#include <string.h>

#include <stdlib.h>

#define MAX\_STUDENTS 100

// Define a structure to store student information

typedef struct {

    int rollNo;

    char name[50];

    float percentage;

} Student;

// Comparison function for sorting students by roll number

int compare(const void \*a, const void \*b) {

    return ((Student\*)a)->rollNo - ((Student\*)b)->rollNo;

}

// Binary search function to find a student by roll number

int binarySearch(Student students[], int n, int rollNo) {

    int left = 0, right = n - 1;

    // Perform binary search

    while (left <= right) {

        int mid = left + (right - left) / 2;

        if (students[mid].rollNo == rollNo) {

            return mid;  // Found the student at index mid

        }

        else if (students[mid].rollNo < rollNo) {

            left = mid + 1;  // Move right

        }

        else {

            right = mid - 1;  // Move left

        }

    }

    return -1;  // Return -1 if the student is not found

}

int main() {

    int n, rollNo, result;

    Student students[MAX\_STUDENTS];

    // Input the number of students

    printf("Enter number of students: ");

    scanf("%d", &n);

    // Input student data

    for (int i = 0; i < n; i++) {

        printf("\nEnter details for student %d:\n", i + 1);

        printf("Roll No: ");

        scanf("%d", &students[i].rollNo);

        getchar();  // Consume the newline character left by scanf

        printf("Name: ");

        fgets(students[i].name, 50, stdin);

        students[i].name[strcspn(students[i].name, "\n")] = '\0';  // Remove the trailing newline

        printf("Percentage: ");

        scanf("%f", &students[i].percentage);

    }

    // Sort the students by Roll No using qsort

    qsort(students, n, sizeof(Student), compare);

    // Input the roll number to search for

    printf("\nEnter Roll No to search for: ");

    scanf("%d", &rollNo);

    // Perform binary search

    result = binarySearch(students, n, rollNo);

    // Display the result of the search

    if (result != -1) {

        printf("\nStudent found!\n");

        printf("Roll No: %d\n", students[result].rollNo);

        printf("Name: %s\n", students[result].name);

        printf("Percentage: %.2f\n", students[result].percentage);

    } else {

        printf("\nStudent with Roll No %d not found.\n", rollNo);

    }

    return 0;

}



