ASSIGNMENT NO 12

Implementation of a direct access file -Insertion and deletion of a record from a direct access file

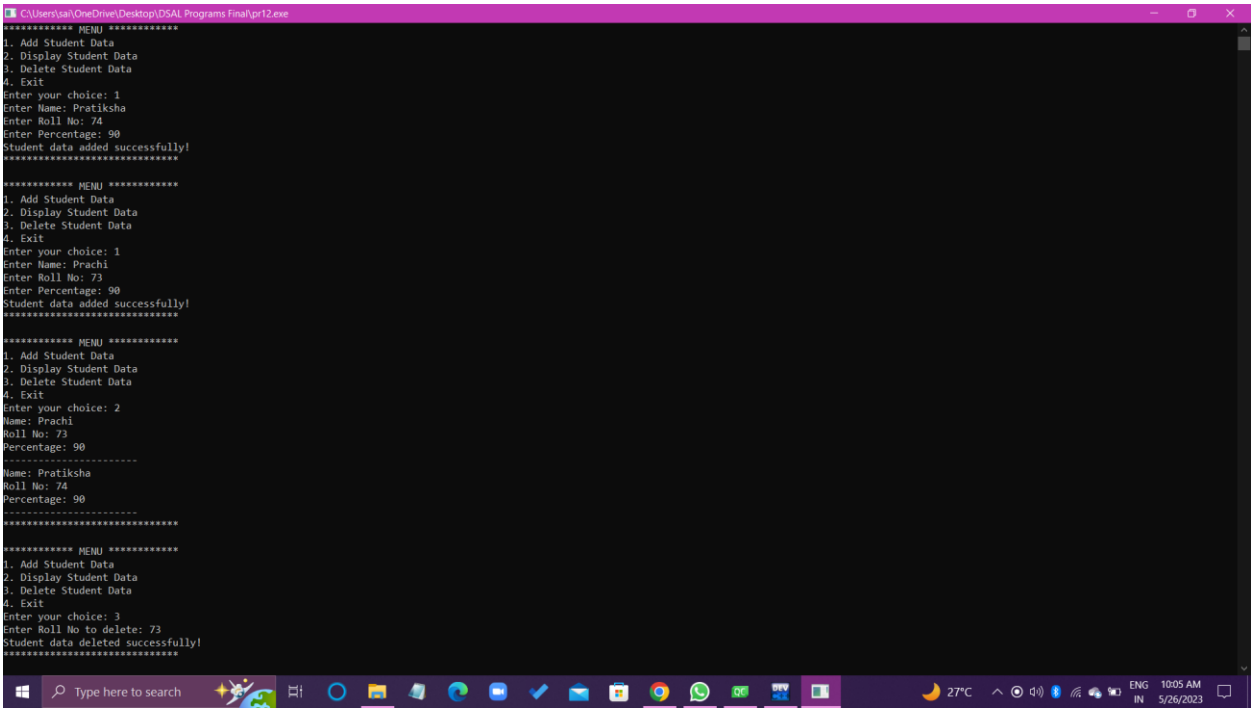
```
#include <iostream>
#include <fstream>
#include <cstring>

using namespace std;

const int MAX_RECORDS = 10;

class Student {
public:

    char name[50];
    int rollNo;
    float percentage;
    Student(){}

    Student(const char* _name, int _rollNo, float _percentage) {
        strcpy(name, _name);
        rollNo = _rollNo;
        percentage = _percentage;
    }

    void display() const {
        cout << "Name: " << name << endl;
        cout << "Roll No: " << rollNo << endl;
        cout << "Percentage: " << percentage << endl;
        cout << "----------------------" << endl;
    }

    void writeToFile(fstream& outFile) {
        outFile.write(reinterpret_cast<const char*>(this), sizeof(Student));
    }

    void readFromFile(fstream& inFile) {
        inFile.read(reinterpret_cast<char*>(this), sizeof(Student));
    }

    void readFromFile(ifstream& inFile) {
        inFile.read(reinterpret_cast<char*>(this), sizeof(Student));
    }

    int getRollNo() const {
```

```
        return rollNo;
    }
};

int hashFunction(int rollNo) {
    return rollNo % MAX_RECORDS;
}

void createFile() {
    fstream file("student_data.txt", ios::out | ios::binary);
    if (!file) {
        cout << "Error creating file." << endl;
        return;
    }

    int hashValue = -1;

     for (int i = 0; i < MAX_RECORDS; i++) {
        //Student *Pos = i*sizeof(Student);
        file.write(reinterpret_cast<const char*>(&hashValue), sizeof(Student));
    }

    file.close();
}

void addStudentData() {
    char name[50];
    int rollNo;
    float percentage;

    cout << "Enter Name: ";
    cin.ignore();
    cin.getline(name, 50);
    cout << "Enter Roll No: ";
    cin >> rollNo;
    cout << "Enter Percentage: ";
    cin >> percentage;

    int hashValue = hashFunction(rollNo);
    fstream file("student_data.txt", ios::in | ios::out | ios::binary);
    if (!file) {
        cout << "Error opening file." << endl;
        return;
    }
```

```
    file.seekp(hashValue * sizeof(Student) + sizeof(int), ios::beg);
    file.write(reinterpret_cast<const char*>(&hashValue), sizeof(int));

    Student student(name, rollNo, percentage);
    student.writeToFile(file);

    cout << "Student data added successfully!" << endl;
    file.close();
}

void displayStudentData() {
    ifstream file("student_data.txt", ios::binary);
    if (!file) {
        cout << "Error opening file." << endl;
        return;
    }

    Student student;
    int hashValue;

    for (int i = 0; i < MAX_RECORDS; i++) {
        file.seekg(i * sizeof(Student) + sizeof(int), ios::beg);
        file.read(reinterpret_cast<char*>(&hashValue), sizeof(int));
        if (hashValue != -1) {
            student.readFromFile(file);
            if(student.rollNo >0){
            student.display();
                        }
        }
    }

    file.close();
}

void deleteStudentData(int rollNo) {
    ifstream inputFile("student_data.txt", ios::binary);
    if (!inputFile) {
        cout << "Error opening file." << endl;
        return;
    }

    fstream outputFile("temp_data.txt", ios::out | ios::binary);
    if (!outputFile) {
        cout << "Error creating temporary file." << endl;
        inputFile.close();
```

```
        return;
    }

    Student student;
    int hashValue;

    for (int i = 0; i < MAX_RECORDS; i++) {
        inputFile.seekg(i * sizeof(Student) + sizeof(int), ios::beg);
        inputFile.read(reinterpret_cast<char*>(&hashValue), sizeof(int));
        if (hashValue != -1) {
            student.readFromFile(inputFile);
            if (student.getRollNo() != rollNo) {
                outputFile.seekp(i * sizeof(Student) + sizeof(int), ios::beg);
                outputFile.write(reinterpret_cast<const char*>(&hashValue),
sizeof(int));
                student.writeToFile(outputFile);
            }
        }
    }

    inputFile.close();
    outputFile.close();

    remove("student_data.txt");
    rename("temp_data.txt", "student_data.txt");

    cout << "Student data deleted successfully!" << endl;
}

int main() {
    createFile();

    int choice;
    do {
        cout << "************ MENU ************" << endl;
        cout << "1. Add Student Data" << endl;
        cout << "2. Display Student Data" << endl;
        cout << "3. Delete Student Data" << endl;
        cout << "4. Exit" << endl;
        cout << "Enter your choice: ";
        cin >> choice;

        switch (choice) {
            case 1:
                addStudentData();
```

```
                break;
            case 2:
                displayStudentData();
                break;
            case 3: {
                int rollNo;
                cout << "Enter Roll No to delete: ";
                cin >> rollNo;
                deleteStudentData(rollNo);
                break;
            }
            case 4:
                cout << "Exiting program." << endl;
                break;
            default:
                cout << "Invalid choice. Try again." << endl;
                break;
        }

        cout << "*****************************" << endl;
        cout << endl;
    } while (choice != 4);

    return 0;
}
```


C:\Users\sai\OneDrive\Desktop\DSAL Programs Final\pr12.exe
********** MENU **********
1. Add Student Data
2. Display Student Data
3. Delete Student Data
4. Exit
Enter your choice: 1
Enter Name: Pratiksha
Enter Roll No: 74
Enter Percentage: 90
Student data added successfully!
****
********** MENU **********
1. Add Student Data
2. Display Student Data
3. Delete Student Data
4. Exit
Enter your choice: 1
Enter Name: Prachi
Enter Roll No: 73
Enter Percentage: 90
Student data added successfully!
****
********** MENU **********
1. Add Student Data
2. Display Student Data
3. Delete Student Data
4. Exit
Enter your choice: 2
Name: Prachi
Roll No: 73
Percentage: 90

Name: Pratiksha
Roll No: 74
Percentage: 90

****
********** MENU **********
1. Add Student Data
2. Display Student Data
3. Delete Student Data
4. Exit
Enter your choice: 3
Enter Roll No to delete: 73
Student data deleted successfully!
****
Type here to search
27°C
ENG
IN
10:05 AM
5/26/2023