Lab Assignment 4

Name: Kamran Ansari
Reg No: 22MCA0223

Question

Implement a student management system using LinkedList in Java. The system should have the following functionalities:
1. Add Student: Implement a method addStudent that takes the student's name and ID as parameters and adds a new student record to the system.
2. Remove Student: Implement a method removeStudent that takes the student's ID as a parameter and removes the corresponding student record from the system.
3. Search Student: Implement a method searchStudent that takes the student's ID as a parameter and returns the student's name if found in the system or an appropriate message if the student is not found.
4. Update Student Grade: Implement a method updateStudentGrade that takes the student's ID and new grade as parameters and updates the grade for the corresponding student in the system.
5. Display All Students: Implement a method displayAllStudents that displays the names and IDs of all the students in the system.

Write a Java program that demonstrates the above functionalities by interacting with the user. The program should provide a menu-driven interface where the user can choose options to add, remove, search, or display students.

Code

Student.java

```
public class Student {
    public String name;
    public String id;
    public char grade;

    public Student(String name, String id, char grade) {
        this.name = name;
        this.id = id;
```

```
        this.grade = grade;
    }

    public String toString() {
        StringBuilder sb = new StringBuilder();

        sb.append("Student:\n");
        sb.append("ID: " + this.id + "\n");
        sb.append("Name: " + this.name + "\n");
        sb.append("Grade: " + this.grade + "\n");

        return sb.toString();
    }
}
```

StudentList.java

```
import java.util.LinkedList;

public class StudentList {
    LinkedList<Student> list;

    public StudentList() {
        this.list = new LinkedList<>();
    }

    public boolean addStudent(String id, String name, char grade) {
        Student newStudent = new Student(name, id, grade);

        return this.list.add(newStudent);
    }

    public Student searchStudent(String id) {
        Student studentToFind = null;

        for (Student student : this.list) {
            if (student.id.equals(id)) {
                studentToFind = student;
                break;
            }
        }

        return studentToFind;
    }

    public Student removeStudent(String id) {
        Student studentToRemove = this.searchStudent(id);

        this.list.remove(studentToRemove);

        return studentToRemove;
```

```
    }

    public boolean updateStudentGrade(String id, char grade) {
        Student studentToUpgrade = searchStudent(id);

        if (studentToUpgrade != null) {
            studentToUpgrade.grade = grade;
            return true;
        }

        return false;
    }

    public void displayAllStudents() {
        System.out.println("Students: ");
        for (Student student : this.list) {
            System.out.println(student);
        }
    }
}
```

Main.java

```
import java.util.Scanner;

public class Main {
    public static void main(String[] args) {
        Scanner scan = new Scanner(System.in);
        StudentList studentList = new StudentList();

        while (true) {
            System.out.println("");
            System.out.println("Student Management System");
            System.out.println("Menu:");
            System.out.println("1. Add Student");
            System.out.println("2. Remove Student");
            System.out.println("3. Search Student");
            System.out.println("4. Update Student Grade");
            System.out.println("5. Display All Student");
            System.out.println("6. Exit");

            System.out.println("Choice:");
            int choice = Integer.valueOf(scan.nextLine());

            switch(choice) {
                case 1: {
                    System.out.println("Enter Student Details: ");
                    System.out.println("Enter Student id: ");
                    String id = scan.nextLine();
                    System.out.println("Enter Student name: ");
                    String name = scan.nextLine();
```

```
                    System.out.println("Enter Student grade: ");
                    char grade = scan.nextLine().charAt(0);

                    boolean res = studentList.addStudent(id, name,
grade);
                    if (res) {
                        System.out.println("Student Added Successfully");
                    } else {
                        System.out.println("Unable to add Student");
                    }

                    break;
                }

                case 2: {
                    System.out.println("Enter Student id: ");
                    String id = scan.nextLine();

                    Student removedStudent =
studentList.removeStudent(id);

                    if (removedStudent != null) {
                        System.out.println("Removed Student - ");
                        System.out.println(removedStudent);
                    } else {
                        System.out.println("No student found!");
                    }

                    break;
                }

                case 3: {
                    System.out.println("Enter Student id: ");
                    String id = scan.nextLine();

                    Student student = studentList.searchStudent(id);

                    if (student != null) {
                        System.out.println("Student - ");
                        System.out.println(student);
                    } else {
                        System.out.println("No student found!");
                    }

                    break;
                }

                case 4: {
                    System.out.println("Enter Student id: ");
                    String id = scan.nextLine();
                    System.out.println("Enter new grade:");
```

```
                    char grade = scan.nextLine().charAt(0);

                    boolean updated = studentList.updateStudentGrade(id,
grade);

                    if (updated) {
                        System.out.println("Grades Updated!");
                    } else {
                        System.out.println("No student found!");
                    }

                    break;
                }

                case 5: {
                    studentList.displayAllStudents();

                    break;
                }

                case 6: {
                    return;
                }

                default: {
                    System.out.println("Invalid Choice");
                }
            }
        }
    }
}
```

Output

```
Student Management System
Menu:
1. Add Student
2. Remove Student
3. Search Student
4. Update Student Grade
5. Display All Student
6. Exit
Choice:
1
Enter Student Details:
Enter Student id:
321
Enter Student name:
name2
Enter Student grade:
B
Student Added Successfully
```

```
Student Management System
Menu:
1. Add Student
2. Remove Student
3. Search Student
4. Update Student Grade
5. Display All Student
6. Exit
Choice:
3
Enter Student id:
123
Student -
Student:
ID: 123
Name: name1
Grade: A


Student Management System
Menu:
1. Add Student
2. Remove Student
3. Search Student
4. Update Student Grade
5. Display All Student
6. Exit
Choice:
3
Enter Student id:
12
No student found!
```



```
Student Management System
Menu:
1. Add Student
2. Remove Student
3. Search Student
4. Update Student Grade
5. Display All Student
6. Exit
Choice:
4
Enter Student id:
321
Enter new grade:
A
Grades Updated!
```

```
Student Management System
Menu:
1. Add Student
2. Remove Student
3. Search Student
4. Update Student Grade
5. Display All Student
6. Exit
Choice:
5
Students:
Student:
ID: 123
Name: name1
Grade: A

Student:
ID: 321
Name: name2
Grade: A


Student Management System
Menu:
1. Add Student
2. Remove Student
3. Search Student
4. Update Student Grade
5. Display All Student
6. Exit
Choice:
6
```



```
Student Management System
Menu:
1. Add Student
2. Remove Student
3. Search Student
4. Update Student Grade
5. Display All Student
6. Exit
Choice:
2
Enter Student id:
123
Removed Student -
Student:
ID: 123
Name: name1
Grade: A
```

```
Student Management System
Menu:
1. Add Student
2. Remove Student
3. Search Student
4. Update Student Grade
5. Display All Student
6. Exit
Choice:
5
Students:
Student:
ID: 321
Name: name2
Grade: A
```