Assignment 3

(Classes, Inheritance, Linked Lists)

Due Date: June, 11 at the beginning of the class

Late assignments will not be accepted.

Write a Java program that has the following classes:

```java
public abstract class Person {
    private String fName;
    private String lName;
    public Person (String fName, String lName)
    public int compareTo(String fn, String ln); // returns (fName + " " + lName).compareTo(fn + " " + ln)
    public String toString(); // returns a string containing fName and lName separated by a space
}

public abstract class Student extends Person {
    private double gpa;
    private String major;
    public Student(String fName, String lName, double gpa, String major);
    public abstract String standing();
    public double getGPA(); // returns this student’s gpa
    public String toString(); // returns the information returned by toString method of the class Person and this student’s gpa, major and standing all separated by a space
}

public class UnderGrad extends Student {
    private String club;
    public UnderGrad(String fName, String lName, double gpa, String major, String club);
    public String standing(); // returns “good” if gpa >= 2.0, otherwise returns “probation”
    public String toString(); // returns the information returned by toString method of the class Student and the this undergraduate student’s club separated by a space
}

public class Grad extends Student {
    private Advisor advisor;
    public Grad(String fName, String lName, double gpa, String major, Advisor advisor);
    public String standing(); // returns “good” if gpa >= 3.0, otherwise returns “probation”
```
public String toString(); // returns the information returned by toString method
    of the class Student and toString method of this graduate
    student’s advisor separated by a space
    }

    public class Advisor extends Person {
        public Advisor (String fName, String lName);
    }

Test your classes using a simple test method that you will write.

Write the following classes to implement a linked list

    public class Node {
        public Student value;
        public Node next;
        public Node(Student s);
    }

    public class LinkedList {
        private Node first;
        public LinkedList();
        public void add(Student s); //add s Student s to the list. The list is maintained in the ascending
        order of fName + " " + lName of students in the linked list;
        public boolean remove(String fn, String ln); removes student whose fName=fn and lName= ln
        from the list. If the student is in the list and is removed, returns true otherwise returns false.
        public void display(); // if the list empty, display “List is empty” otherwise, for each student in
        the list, display the information returned by the student’s toString method.
    }

Use the main method below to test your program. This method and a sample testfile (datafile.txt) is available from www.cs.fiu.edu/~milani/cop3337

    public static void main(String[] args) throws IOException {
        // TODO code application logic here
        String fName;
        String lName;
        double gpa;
        String advisorFn;
        String advisorLn;
        String club;
LinkedList students = new LinkedList();
File inFile = new File("dataFile.txt");
Scanner in = new Scanner(inFile);

String cmd;
while (in.hasNext()) {
    cmd = in.next();
    if (cmd.charAt(0) == 'd') {
        students.display();
    } else {
        switch (cmd.charAt(0)) {
            case 'g':
                fName = in.next();
                lName = in.next();
                gpa = in.nextDouble();
                major = in.next();
                advisorFn = in.next();
                advisorLn = in.next();
                Advisor advisor = new Advisor(advisorFn, advisorLn);
                Grad g = new Grad(fName, lName, gpa, major, advisor);
                students.add(g);
                break;
            case 'u':
                fName = in.next();
                lName = in.next();
                gpa = in.nextDouble();
                major = in.next();
                club = in.next();
                UnderGrad u = new UnderGrad(fName, lName, gpa, club);
                students.add(u);
                break;
            case 'r':
                fName = in.next();
                lName = in.next();
                if (!students.remove(fName, lName))
                    System.out.println(fName + " " + lName + " not found");
                break;
            case 'd':
                students.display();
            default:
                System.out.println("invalid command");
                break;
        }
    }
}

**Turn in class:** Printed listings of all your java source code and a sample output
Submit via email: Zip your entire project folder and email to milani@cs.fiu.edu. The title of the email should be “Asg3 – Your Name”