**IT206-PARADIGMS OF PROGRAMMING**

**Mini Project Report**

**BACHELOR OF TECHNOLOGY IN**

**INFORMATION TECHNOLOGY**

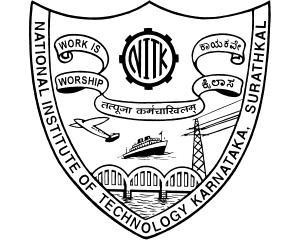
**BY**

**UJJWAL PASUPULETY**

**RAJEEV ANIRUDH**

**AIMAN ABDULLAH ANEES**

**JYOTI PRAKASH SAHOO**



Under The Guidance

**Mr. PRADEEP JAGANNATH**

**DEPARTMENT OF INFORMATION TECHNOLOGY**

**NATIONAL INSTITUTE OF TECHNOLOGY KARNATAKA, SURATHKAL SRINIVASNAGAR- 575025 KARNATAKA,**

**NOVEMBER 2016**

D E C L A R A T I O N

We hereby declare that the Project Work Report entitled “**PARADIGMS OF PROGRAMMING Mini Project**” which is being submitted to **National Institute of Technology Karnataka, Surathkal** for the completion of the mini project for the degree of **Bachelor of Technology in INFORMATION TECHNOLOGY** is a bonafide report of the work carried out by us.

|  |  |  |
| --- | --- | --- |
|  | **NAME** | **REGISTRATION NO.** |
| **1.** | **UJJWAL PASUPULETY** | **15IT150** |
| **2.** | **RAJEEV ANIRUDH** | **15IT230** |
| **3.** | **AIMAN ABUDULLAH ANEES** | **15IT106** |

**4. JYOTI PRAKASH SAHOO 15IT213**

Department of INFORMATION AND TECHNOLOGY

**Place: NITK, Surathkal**

**Date: 9 November,2016**

CERTIFICATE

This is to certify that the B.Tech Mini Project Work Report entitled “**PARADIGMS OF PROGRAMMING**” submitted by

|  |  |  |
| --- | --- | --- |
| **Sl.No.** | **Registration No** | **Name** |
| 1. | **15IT150** | **UJJWAL PASUPULETY** |
| 2. | **15IT230** | **RAJEEV ANIRUDH** |
| 3. | **15IT106** | **AIMAN ABDULLAH ANEES** |

4**. 15IT213** **JYOTI PRAKASH SAHOO**

as the record of the work carried out by them is accepted as the **B.Tech Mini Project Work Report Submission** in partial fulfillment of the requirements for the credits in **PARADIGMS OF PROGRAMMING** of **Bachelor of Technology in INFORMATION OF TECHNOLOGY**.

Project Guide:

**Mr. Pradeep Jagannath**

Department of Information Of Technology

NITK Surathkal

ACKNOWLEDGEMENT

We take this opportunity to express our deepest gratitude and appreciation to all those who have helped us directly or indirectly towards the successful completion of this project.

First and foremost, we would like to express our sincere appreciation and gratitude to our esteemed guide Mr. Pradeep Jagannath Department of Information Of Technology, NITK Surathkal for her insightful advice, encouragement, guidance, critics, and valuable suggestions throughout the course of our project work. Without her continued support and interest, this thesis would not have been the same as presented here.

We would like to take this opportunity to express our thanks towards the teaching and non-teaching staff in the Department of Information Of Technology, NITK for their invaluable help and support in this semester of our study. We are also grateful to all our classmates for their help, encouragement and invaluable suggestions.

Last but not least, we thank God Almighty for his blessings without which the completion of this project work would not have been possible.

UJJWAL PASUPULETY, 15IT150

RAJEEV ANIRUDH, 15IT230

AIMAN ABDULLAH ANEES, 15IT106

JYOTI PRAKASH SAHOO, 15IT213

**TABLE OF CONTENT**

1. Introduction…………………………………1
2. Requirements………………………………..2
3. Architecture…………………………………3
4. Class diagram……………………………….4
5. Code………………………………………...5
6. Flow diagram………………………………..29

**Introduction**

As we can see while joining into an engineering college we need to fill the course registration form to choose respective courses of particular credit.as for graduation, we required a minimum number of credits to be completed. So by keeping this idea in mind we created a GUI which will show the buttons of a student who have the roll number and will ask for the semester and according to the semester the program will ask for electives to be chosen and a number of credits the student completed and the graduation status.

**Requirements**

* import java.awt.Dimension
* import java.awt.FlowLayout
* import java.awt.event.ActionEvent
* import java.awt.event.ActionListener
* import java.util.\*
* import java.lang.\*
* import javax.swing.JButton
* import javax.swing.JFrame
* import javax.swing.JLabel
* import javax.swing.SwingUtilities
* import javax.swing.WindowConstants
* eclipse IDE

**Architecture diagram**

The program includes inheritance,GUI,ArrayList

**Class Diagram**

**Code**

**import java.awt.Dimension;**

**import java.awt.FlowLayout;**

**import java.awt.event.ActionEvent;**

**import java.awt.event.ActionListener;**

**import java.util.\*;**

**import javax.swing.JButton;**

**import javax.swing.JFrame;**

**import javax.swing.JLabel;**

**import javax.swing.SwingUtilities;**

**import javax.swing.WindowConstants;**

**abstract class Course**

**{**

**String ccode;**

**String cname;**

**int credits;**

**}**

**class Core extends Course**

**{**

**public Core(String c, String s, int cr)**

**{**

**ccode=c;**

**cname=s;**

**credits=cr;**

**}**

**}**

**class Elective extends Course**

**{**

**public Elective(String c, String s, int cr)**

**{**

**ccode=c;**

**cname=s;**

**credits=cr;**

**}**

**}**

**class Student**

**{**

**int semno;**

**String name;**

**String rollno;**

**int credits;**

**ArrayList <Elective>e=new ArrayList<Elective>();**

**boolean desc;**

**public Student(String a, String b)**

**{**

**semno=1;**

**name=a;**

**rollno=b;**

**credits=0;**

**desc=false;**

**}**

**void update(int cr)**

**{**

**credits+=cr;**

**++semno;**

**}**

**void display()**

**{**

**System.out.println("name:"+name);**

**System.out.println("roll number:"+rollno);**

**System.out.println("credits completed:"+credits);**

**if(semno<=8)**

**System.out.println("semester:"+semno);**

**else System.out.println("Graduated");**

**System.out.println("Electives taken:");**

**for(int i=0;i<e.size();++i)**

**System.out.println(e.get(i).cname);**

**}**

**}**

**public class gui**

**{**

**static JFrame frame;**

**static JLabel lab;**

**public static void main(String[] args)**

**{**

**// schedule this for the event dispatch thread (edt)**

**SwingUtilities.invokeLater(new Runnable()**

**{**

**public void run()**

**{**

**displayJFrame();**

**}**

**});**

**}**

**static void displayJFrame()**

**{**

**frame = new JFrame("GUI Menu");**

**ArrayList <Core>s1 = new ArrayList<Core>();**

**s1.add(new Core("MA110","MATH",3));**

**s1.add(new Core("CY110","CHEMISTRY",3));**

**s1.add(new Core("AM110","ENGINEERING MECHANICS",3));**

**s1.add(new Core("EE110","ELEMENTS OF ELECTRICAL ENGINEERING",3));**

**s1.add(new Core("ME111","ENGINEERING GRAPHICS",3));**

**s1.add(new Core("HU110","PROFESSIONAL COMMUNICATION",3));**

**s1.add(new Core("CY111","CHEMISTRY LAB",2));**

**s1.add(new Core("CV110","ENVIORNMENTAL STUDIES",1));**

**ArrayList <Core>s2 = new ArrayList<Core>();**

**s2.add(new Core("MA111","MATH",3));**

**s2.add(new Core("PH110","PHYSICS",4));**

**s2.add(new Core("CO110","COMPUTER PROGRAMMING",4));**

**s2.add(new Core("ME110","ELEMENTS OF MECHANICAL ENGINEERING",3));**

**s2.add(new Core("EC110","ELEMENTS OF ELECTRONICS & COMMUNICATION ENGINEERING",3));**

**s2.add(new Core("PH111","PHYSICS LAB",1));**

**s2.add(new Core("CO111","COMPUTER LAB",1));**

**s2.add(new Core("HU111","PROFESSIONAL ETHICS AND HUMAN VALUES",1));**

**ArrayList <Core>s3 = new ArrayList<Core>();**

**s3.add(new Core("IT200","Data Structures and Algorithms",4));**

**s3.add(new Core("IT204","Data Structures and Algorithms Lab",2));**

**s3.add(new Core("IT201","Digital Design And Computer Organization",4));**

**s3.add(new Core("IT202","Unix Programming and Practice",3));**

**s3.add(new Core("IT203","Computer Systems Organization Lab",3));**

**s3.add(new Core("MA200","Mathematical Foundations of Information Technology",4));**

**s3.add(new Core("IT206","Paradigms of Programming-I",4));**

**ArrayList <Core>s4 = new ArrayList<Core>();**

**s4.add(new Core("IT250","Operating Systems",4));**

**s4.add(new Core("IT251","Computer Communication and Networks",4));**

**s4.add(new Core("IT252","Design and Analysis of Algorithms",4));**

**s4.add(new Core("IT253","Paradigms of Programming-II",4));**

**s4.add(new Core("IT290","Seminar",2));**

**ArrayList <Core>s5 = new ArrayList<Core>();**

**s5.add(new Core("IT300","Parallel Computing",4));**

**s5.add(new Core("IT301","Database Systems",4));**

**s5.add(new Core("IT302","Web Technologies and Applications",4));**

**s5.add(new Core("IT303","Automata and Compiler Design",4));**

**ArrayList <Core>s6 = new ArrayList<Core>();**

**s6.add(new Core("IT350","Software Engineering",4));**

**s6.add(new Core("IT351","Human Computer Interaction",4));**

**s6.add(new Core("IT352","Information Assurance and Security",4));**

**s6.add(new Core("IT399","Minor Project",2));**

**ArrayList <Core>s7 = new ArrayList<Core>();**

**s7.add(new Core("IT440","Practical Training",4));**

**s7.add(new Core("IT449","Major Project-I",2));**

**ArrayList <Core>s8 = new ArrayList<Core>();**

**s8.add(new Core("IT499","Major Project-II",6));**

**ArrayList <Student>al = new ArrayList<Student>();**

**JButton addStudent = new JButton("Click to Add a Student");**

**// add the listener to the jbutton to handle the "pressed" event**

**addStudent.addActionListener(new ActionListener()**

**{**

**public void actionPerformed(ActionEvent e)**

**{**

**lab.setText("Adding Student....");**

**// display/center the jdialog when the button is pressed**

**// display/center the jdialog when the button is pressed**

**Scanner in=new Scanner(System.in);**

**System.out.println("Enter Name:");**

**String a=new String(in.nextLine());**

**System.out.println("Enter RollNo.:");**

**String b=new String(in.nextLine());**

**al.add(new Student(a,b));**

**System.out.println("Student Added");**

**}**

**});**

**JButton searchStudent = new JButton("Search Student");**

**// add the listener to the jbutton to handle the "pressed" event**

**searchStudent.addActionListener(new ActionListener()**

**{**

**public void actionPerformed(ActionEvent e)**

**{**

**// display/center the jdialog when the button is pressed**

**Scanner in=new Scanner(System.in);**

**System.out.println("Enter Roll:");**

**String a=new String(in.nextLine());**

**boolean flag=false;**

**for (int i = 0; i < al.size(); i++) {**

**if(a.equalsIgnoreCase(al.get(i).rollno))**

**{ flag=true;**

**al.get(i).display();**

**System.out.println("Core courses taken:");**

**switch(al.get(i).semno){**

**case 1:for (int j = 0; j < s1.size(); j++) {**

**System.out.println(s1.get(j).cname);**

**}**

**break;**

**case 2:for (int j = 0; j < s2.size(); j++) {**

**System.out.println(s2.get(j).cname);**

**} break;**

**case 3:for (int j = 0; j < s3.size(); j++) {**

**System.out.println(s3.get(j).cname);**

**}**

**break;**

**case 4:for (int j = 0; j < s4.size(); j++) {**

**System.out.println(s4.get(j).cname);**

**} break;**

**case 5:for (int j = 0; j < s5.size(); j++) {**

**System.out.println(s5.get(j).cname);**

**} break;**

**case 6:for (int j = 0; j < s6.size(); j++) {**

**System.out.println(s6.get(j).cname);**

**} break;**

**case 7:for (int j = 0; j < s7.size(); j++) {**

**System.out.println(s7.get(j).cname);**

**} break;**

**case 8:for (int j = 0; j < s8.size(); j++) {**

**System.out.println(s8.get(j).cname);**

**} break;**

**}**

**}**

**}**

**if(flag==false)**

**System.out.println("Student not found");**

**}**

**});**

**JButton resolve = new JButton("Resolve Credit Discrepancies");**

**// add the listener to the jbutton to handle the "pressed" event**

**resolve.addActionListener(new ActionListener()**

**{**

**public void actionPerformed(ActionEvent e)**

**{**

**// display/center the jdialog when the button is pressed**

**String opt1=new String();**

**String opt2=new String();**

**String opt3=new String();**

**String opt4=new String();**

**for (int i = 0; i < al.size(); i++) {**

**if(al.get(i).desc==false){**

**if(al.get(i).semno<=3){**

**if(al.get(i).semno==1)**

**System.out.println("there are no electives offered in semester "+al.get(i).semno);**

**if(al.get(i).semno==2)**

**System.out.println("there are no electives offered in semester "+al.get(i).semno);**

**if(al.get(i).semno==3)**

**System.out.println("you have a compulsory elective called Paradigms of Programming-I for 4 credits in semester "+al.get(i).semno);**

**}**

**else{**

**System.out.println("Please enter the number of credits you wish to complete through electives:");**

**Scanner in=new Scanner(System.in);**

**int a=in.nextInt();**

**switch(al.get(i).semno){**

**case 1:**

**System.out.println("there are no electives offered in semester "+al.get(i).semno);**

**break;**

**case 2:**

**System.out.println("there are no electives offered in this semester "+al.get(i).semno);**

**break;**

**case 3:**

**System.out.println("you have a compulsory elective called Paradigms of Programming-I for 4 credits in semester "+al.get(i).semno);**

**break;**

**case 4:**

**if(a<6 || a>8)**

**System.out.println("Please Enter a number between 6 and 8 next time:");**

**else{**

**if(a==6)**

**{**

**System.out.println("you have chosen Paradigms of Programming-II and Information Systems");**

**al.get(i).e.add(new Elective("IT253","Paradigms of Programming-II",3));**

**al.get(i).e.add(new Elective("IT205","Information Systems",3));**

**}**

**if(a==7)**

**{**

**System.out.println("enter 1.Paradignms of Programming-II 2.Information Systems");**

**in.nextLine();**

**opt1=in.nextLine();**

**System.out.println("enter 1.Computer Graphics 2.Microprocessors and Interfaces");**

**in.nextLine();**

**opt2=in.next();**

**System.out.println("you have chosen "+ opt1 +" and " +opt2);**

**if(opt1.equalsIgnoreCase("Paradigms of Programming-II"))**

**al.get(i).e.add(new Elective("IT253",opt1,3));**

**if(opt1.equalsIgnoreCase("Information Systems"))**

**al.get(i).e.add(new Elective("IT205",opt1,3));**

**if(opt2.equalsIgnoreCase("Computer Graphics"))**

**al.get(i).e.add(new Elective("IT254",opt2,4));**

**if(opt2.equalsIgnoreCase("Microprocessors and Interfacing"))**

**al.get(i).e.add(new Elective("IT255",opt2,4));**

**}**

**if(a==8)**

**{**

**System.out.println("You have chosen Microprocessors and Interfacing and Computer Graphics");**

**al.get(i).e.add(new Elective("IT254","Computer Graphics",4));**

**al.get(i).e.add(new Elective("IT255","Microprocessors and Interfacing",4));**

**}**

**System.out.println("Discrepancies Resolved for "+al.get(i).rollno);**

**al.get(i).desc=true;**

**}**

**break;**

**case 5:**

**if(a<6 || a>8)**

**System.out.println("Please Enter a number between 6 and 8 next time:");**

**else{**

**if(a==6)**

**{**

**System.out.println("you have chosen object oriented analysis and design and advanced computer networks");**

**al.get(i).e.add(new Elective("IT306","Object Oriented Analysis and Design",3));**

**al.get(i).e.add(new Elective("IT307","Advanced Computer Networks",3));**

**}**

**if(a==7)**

**{**

**System.out.println("enter 1.Performance Modelling 2.Multimedia Signal Computing");**

**in.nextLine();**

**opt1=in.nextLine();**

**System.out.println("enter 1.Object Oriented Analysis 2.Advanced Computer Networks");**

**in.nextLine();**

**opt2=in.nextLine();**

**if(opt1.equalsIgnoreCase("Performance Modelling"))**

**al.get(i).e.add(new Elective("IT305",opt1,3));**

**if(opt1.equalsIgnoreCase("Multimedia Signal Computing"))**

**al.get(i).e.add(new Elective("IT304",opt1,3));**

**if(opt2.equalsIgnoreCase("Object Oriented Analysis"))**

**al.get(i).e.add(new Elective("IT306",opt2,4));**

**if(opt2.equalsIgnoreCase("Advanced Computer Networks"))**

**al.get(i).e.add(new Elective("IT307",opt2,4));**

**System.out.println("you have chosen " + opt1+" and "+opt2);**

**}**

**if(a==8)**

**{**

**System.out.println("you have chosen Multimedia Signal Computing and Performance Modelling");**

**al.get(i).e.add(new Elective("IT304","Multimedia Signal Computing",4));**

**al.get(i).e.add(new Elective("IT305","Performance Modelling",4));**

**}**

**System.out.println("Discrepancies Resolved for "+al.get(i).rollno);**

**al.get(i).desc=true;**

**}**

**break;**

**case 6:**

**if(a<7 || a>8)**

**System.out.println("Please Enter either 7 or 8 next time:");**

**else{**

**if(a==7)**

**{**

**System.out.println("enter 1.Perceptual Audio Processing 2.Perceptual Video Processing 3.Soft Computing");**

**in.nextLine();**

**opt1=in.nextLine();**

**System.out.println("you have chosen " +opt1+ " and Artificial Intelligence");**

**if(opt1.equalsIgnoreCase("Perceptual Audio Processing"))**

**al.get(i).e.add(new Elective("IT353",opt1,4));**

**if(opt1.equalsIgnoreCase("Perceptual Audio Processing"))**

**al.get(i).e.add(new Elective("IT354",opt1,4));**

**if(opt1.equalsIgnoreCase("Soft Computing"))**

**al.get(i).e.add(new Elective("IT355",opt1,4));**

**al.get(i).e.add(new Elective("IT357","Artificial Intelligience",3));**

**}**

**if(a==8)**

**{**

**System.out.println("enter 1.Perceptual Audio Processing 2.Perceptual Video Processing 3.Soft Computing");**

**in.nextLine();**

**opt1=in.next();**

**in.nextLine();**

**System.out.println("enter 1.Perceptual Audio Processing 2.Perceptual Video Processing 3.Soft Computing");**

**in.nextLine();**

**opt2=in.next();**

**System.out.println("You have chosen " +opt1+ " and "+opt2);**

**if(opt1.equalsIgnoreCase("Perceptual Audio Processing"))**

**al.get(i).e.add(new Elective("IT353",opt1,4));**

**if(opt1.equalsIgnoreCase("Perceptual Audio Processing"))**

**al.get(i).e.add(new Elective("IT354",opt1,4));**

**if(opt1.equalsIgnoreCase("Soft Computing"))**

**al.get(i).e.add(new Elective("IT355",opt1,4));**

**if(opt2.equalsIgnoreCase("Perceptual Audio Processing"))**

**al.get(i).e.add(new Elective("IT353",opt2,4));**

**if(opt2.equalsIgnoreCase("Perceptual Audio Processing"))**

**al.get(i).e.add(new Elective("IT354",opt2,4));**

**if(opt2.equalsIgnoreCase("Soft Computing"))**

**al.get(i).e.add(new Elective("IT355",opt2,4));**

**}**

**System.out.println("Discrepancies Resolved for "+al.get(i).rollno);**

**al.get(i).desc=true;**

**}**

**break;**

**case 7:**

**if(a<12 || a>16)**

**System.out.println("Please Enter a number between 12 and 16 next time:");**

**else{**

**if(a==16)**

**{**

**System.out.println("you have chosen Data Warehousing and Data Mining, MiddleWare Technologies,Computer Vision,Cloud Computing ");**

**al.get(i).e.add(new Elective("IT405","Data Mining",4));**

**al.get(i).e.add(new Elective("IT406","Middleware Technologies",4));**

**al.get(i).e.add(new Elective("IT407","Computer Vision",4));**

**al.get(i).e.add(new Elective("IT409","Cloud Computing",4));**

**}**

**if(a==15)**

**{**

**System.out.println("Enter 1.Data Warehousing and Mining 2.Computer Vision 3.Middleware Technologies 4.Cloud Computing");**

**in.nextLine();**

**opt1=in.next();**

**System.out.println("Enter 1.Data Warehousing and Mining 2.Computer Vision 3.Middleware Technologies 4.Cloud Computing");**

**in.nextLine();**

**opt2=in.next();**

**System.out.println("Enter 1.Data Warehousing and Mining 2.Computer Vision 3.Middleware Technologies 4.Cloud Computing");**

**in.nextLine();**

**opt3=in.next();**

**System.out.println("Enter 1.Mobile Computing 2.Embedded Systems 3.Bioinformatics 4.Knowledge Management");**

**in.nextLine();**

**opt4=in.next();**

**System.out.println("You have chosen"+opt1+","+opt2+","+opt3+","+opt4);**

**if(opt1.equalsIgnoreCase("Data Warehousing and Mining"))**

**al.get(i).e.add(new Elective("IT405",opt1,4));**

**if(opt1.equalsIgnoreCase("Middleware Technologies"))**

**al.get(i).e.add(new Elective("IT406",opt1,4));**

**if(opt1.equalsIgnoreCase("Computer Vision"))**

**al.get(i).e.add(new Elective("IT407",opt1,4));**

**if(opt1.equalsIgnoreCase("Cloud Computing"))**

**al.get(i).e.add(new Elective("IT409",opt1,4));**

**if(opt2.equalsIgnoreCase("Data Warehousing and Mining"))**

**al.get(i).e.add(new Elective("IT405",opt2,4));**

**if(opt2.equalsIgnoreCase("Middleware Technologies"))**

**al.get(i).e.add(new Elective("IT406",opt2,4));**

**if(opt2.equalsIgnoreCase("Computer Vision"))**

**al.get(i).e.add(new Elective("IT407",opt2,4));**

**if(opt2.equalsIgnoreCase("Cloud Computing"))**

**al.get(i).e.add(new Elective("IT409",opt2,4));**

**if(opt3.equalsIgnoreCase("Data Warehousing and Mining"))**

**al.get(i).e.add(new Elective("IT405",opt3,4));**

**if(opt3.equalsIgnoreCase("Middleware Technologies"))**

**al.get(i).e.add(new Elective("IT406",opt3,4));**

**if(opt3.equalsIgnoreCase("Computer Vision"))**

**al.get(i).e.add(new Elective("IT407",opt3,4));**

**if(opt3.equalsIgnoreCase("Cloud Computing"))**

**al.get(i).e.add(new Elective("IT409",opt3,4));**

**if(opt4.equalsIgnoreCase("Mobile Computing"))**

**al.get(i).e.add(new Elective("IT400",opt4,3));**

**if(opt4.equalsIgnoreCase("Embedded Systems"))**

**al.get(i).e.add(new Elective("IT401",opt4,3));**

**if(opt4.equalsIgnoreCase("Bioinformatics"))**

**al.get(i).e.add(new Elective("IT402",opt4,3));**

**if(opt4.equalsIgnoreCase("Knowledge Management"))**

**al.get(i).e.add(new Elective("IT403",opt4,3));**

**}**

**if(a==14)**

**{**

**System.out.println("Enter 1.Data Warehousing and Mining 2.Computer Vision 3.Middleware Technologies 4.Cloud Computing");**

**in.nextLine();**

**opt1=in.next();**

**System.out.println("Enter 1.Data Warehousing and Mining 2.Computer Vision 3.Middleware Technologies 4.Cloud Computing");**

**in.nextLine();**

**opt2=in.next();**

**System.out.println("Enter 1.Mobile Computing 2.Embedded Systems 3.Bioinformatics 4.Knowledge Management");**

**in.nextLine();**

**opt3=in.next();**

**System.out.println("Enter 1.Mobile Computing 2.Embedded Systems 3.Bioinformatics 4.Knowledge Management");**

**in.nextLine();**

**opt4=in.next();**

**System.out.println("You have chosen"+opt1+","+opt2+","+opt3+","+opt4);**

**if(opt1.equalsIgnoreCase("Data Warehousing and Mining"))**

**al.get(i).e.add(new Elective("IT405",opt1,4));**

**if(opt1.equalsIgnoreCase("Middleware Technologies"))**

**al.get(i).e.add(new Elective("IT406",opt1,4));**

**if(opt1.equalsIgnoreCase("Computer Vision"))**

**al.get(i).e.add(new Elective("IT407",opt1,4));**

**if(opt1.equalsIgnoreCase("Cloud Computing"))**

**al.get(i).e.add(new Elective("IT409",opt1,4));**

**if(opt2.equalsIgnoreCase("Data Warehousing and Mining"))**

**al.get(i).e.add(new Elective("IT405",opt2,4));**

**if(opt2.equalsIgnoreCase("Middleware Technologies"))**

**al.get(i).e.add(new Elective("IT406",opt2,4));**

**if(opt2.equalsIgnoreCase("Computer Vision"))**

**al.get(i).e.add(new Elective("IT407",opt2,4));**

**if(opt2.equalsIgnoreCase("Cloud Computing"))**

**al.get(i).e.add(new Elective("IT409",opt2,4));**

**if(opt3.equalsIgnoreCase("Mobile Computing"))**

**al.get(i).e.add(new Elective("IT400",opt3,3));**

**if(opt3.equalsIgnoreCase("Embedded Systems"))**

**al.get(i).e.add(new Elective("IT401",opt3,3));**

**if(opt3.equalsIgnoreCase("Bioinformatics"))**

**al.get(i).e.add(new Elective("IT402",opt3,3));**

**if(opt3.equalsIgnoreCase("Knowledge Management"))**

**al.get(i).e.add(new Elective("IT403",opt3,3));**

**if(opt4.equalsIgnoreCase("Mobile Computing"))**

**al.get(i).e.add(new Elective("IT400",opt4,3));**

**if(opt4.equalsIgnoreCase("Embedded Systems"))**

**al.get(i).e.add(new Elective("IT401",opt4,3));**

**if(opt4.equalsIgnoreCase("Bioinformatics"))**

**al.get(i).e.add(new Elective("IT402",opt4,3));**

**if(opt4.equalsIgnoreCase("Knowledge Management"))**

**al.get(i).e.add(new Elective("IT403",opt4,3));**

**}**

**if(a==13)**

**{**

**System.out.println("Enter 1.Data Warehousing and Mining 2.Computer Vision 3.Middleware Technologies 4.Cloud Computing");**

**in.nextLine();**

**opt1=in.next();**

**System.out.println("Enter 1.Mobile Computing 2.Embedded Systems 3.Bioinformatics 4.Knowledge Management");**

**in.nextLine();**

**opt2=in.next();**

**System.out.println("Enter 1.Mobile Computing 2.Embedded Systems 3.Bioinformatics 4.Knowledge Management");**

**in.nextLine();**

**opt3=in.next();**

**System.out.println("Enter 1.Mobile Computing 2.Embedded Systems 3.Bioinformatics 4.Knowledge Management");**

**in.nextLine();**

**opt4=in.next();**

**System.out.println("You have chosen"+opt1+","+opt2+","+opt3+","+opt4);**

**if(opt1.equalsIgnoreCase("Data Warehousing and Mining"))**

**al.get(i).e.add(new Elective("IT405",opt1,4));**

**if(opt1.equalsIgnoreCase("Middleware Technologies"))**

**al.get(i).e.add(new Elective("IT406",opt1,4));**

**if(opt1.equalsIgnoreCase("Computer Vision"))**

**al.get(i).e.add(new Elective("IT407",opt1,4));**

**if(opt1.equalsIgnoreCase("Cloud Computing"))**

**al.get(i).e.add(new Elective("IT409",opt1,4));**

**if(opt2.equalsIgnoreCase("Mobile Computing"))**

**al.get(i).e.add(new Elective("IT400",opt2,3));**

**if(opt2.equalsIgnoreCase("Embedded Systems"))**

**al.get(i).e.add(new Elective("IT401",opt2,3));**

**if(opt2.equalsIgnoreCase("Bioinformatics"))**

**al.get(i).e.add(new Elective("IT402",opt2,3));**

**if(opt2.equalsIgnoreCase("Knowledge Management"))**

**al.get(i).e.add(new Elective("IT403",opt2,3));**

**if(opt3.equalsIgnoreCase("Mobile Computing"))**

**al.get(i).e.add(new Elective("IT400",opt3,3));**

**if(opt3.equalsIgnoreCase("Embedded Systems"))**

**al.get(i).e.add(new Elective("IT401",opt3,3));**

**if(opt3.equalsIgnoreCase("Bioinformatics"))**

**al.get(i).e.add(new Elective("IT402",opt3,3));**

**if(opt3.equalsIgnoreCase("Knowledge Management"))**

**al.get(i).e.add(new Elective("IT403",opt3,3));**

**if(opt4.equalsIgnoreCase("Mobile Computing"))**

**al.get(i).e.add(new Elective("IT400",opt4,3));**

**if(opt4.equalsIgnoreCase("Embedded Systems"))**

**al.get(i).e.add(new Elective("IT401",opt4,3));**

**if(opt4.equalsIgnoreCase("Bioinformatics"))**

**al.get(i).e.add(new Elective("IT402",opt4,3));**

**if(opt4.equalsIgnoreCase("Knowledge Management"))**

**al.get(i).e.add(new Elective("IT403",opt4,3));**

**}**

**if(a==12)**

**{**

**System.out.println("Enter 1.Mobile Computing 2.Embedded Systems 3.Bioinformatics 4.Knowledge Management");**

**in.nextLine();**

**opt1=in.next();**

**System.out.println("Enter 1.Mobile Computing 2.Embedded Systems 3.Bioinformatics 4.Knowledge Management");**

**in.nextLine();**

**opt2=in.next();**

**System.out.println("Enter 1.Mobile Computing 2.Embedded Systems 3.Bioinformatics 4.Knowledge Management");**

**in.nextLine();**

**opt3=in.next();**

**System.out.println("Enter 1.Mobile Computing 2.Embedded Systems 3.Bioinformatics 4.Knowledge Management");**

**in.nextLine();**

**opt4=in.next();**

**System.out.println("You have chosen"+opt1+","+opt2+","+opt3+","+opt4);**

**if(opt1.equalsIgnoreCase("Mobile Computing"))**

**al.get(i).e.add(new Elective("IT400",opt1,3));**

**if(opt1.equalsIgnoreCase("Embedded Systems"))**

**al.get(i).e.add(new Elective("IT401",opt1,3));**

**if(opt1.equalsIgnoreCase("Bioinformatics"))**

**al.get(i).e.add(new Elective("IT402",opt1,3));**

**if(opt1.equalsIgnoreCase("Knowledge Management"))**

**al.get(i).e.add(new Elective("IT403",opt1,3));**

**if(opt2.equalsIgnoreCase("Mobile Computing"))**

**al.get(i).e.add(new Elective("IT400",opt2,3));**

**if(opt2.equalsIgnoreCase("Embedded Systems"))**

**al.get(i).e.add(new Elective("IT401",opt2,3));**

**if(opt2.equalsIgnoreCase("Bioinformatics"))**

**al.get(i).e.add(new Elective("IT402",opt2,3));**

**if(opt2.equalsIgnoreCase("Knowledge Management"))**

**al.get(i).e.add(new Elective("IT403",opt2,3));**

**if(opt3.equalsIgnoreCase("Mobile Computing"))**

**al.get(i).e.add(new Elective("IT400",opt3,3));**

**if(opt3.equalsIgnoreCase("Embedded Systems"))**

**al.get(i).e.add(new Elective("IT401",opt3,3));**

**if(opt3.equalsIgnoreCase("Bioinformatics"))**

**al.get(i).e.add(new Elective("IT402",opt3,3));**

**if(opt3.equalsIgnoreCase("Knowledge Management"))**

**al.get(i).e.add(new Elective("IT403",opt3,3));**

**if(opt4.equalsIgnoreCase("Mobile Computing"))**

**al.get(i).e.add(new Elective("IT400",opt4,3));**

**if(opt4.equalsIgnoreCase("Embedded Systems"))**

**al.get(i).e.add(new Elective("IT401",opt4,3));**

**if(opt4.equalsIgnoreCase("Bioinformatics"))**

**al.get(i).e.add(new Elective("IT402",opt4,3));**

**if(opt4.equalsIgnoreCase("Knowledge Management"))**

**al.get(i).e.add(new Elective("IT403",opt4,3));**

**}**

**System.out.println("Discrepancies Resolved for "+al.get(i).rollno);**

**al.get(i).desc=true;**

**}**

**break;**

**case 8:**

**if(a!=9)**

**System.out.println("Please Enter 9 next time:");**

**else{**

**if(a==9){**

**System.out.println("Enter 1.Web Services 2.Software Architecture 3.Computer Architecture 4.Transaction Processing 5.Software Quality Assurance");**

**in.nextLine();**

**opt1=in.next();**

**System.out.println("Enter 1.Web Services 2.Software Architecture 3.Computer Architecture 4.Transaction Processing 5.Software Quality Assurance");**

**in.nextLine();**

**opt2=in.next();**

**System.out.println("Enter 1.Web Services 2.Software Architecture 3.Computer Architecture 4.Transaction Processing 5.Software Quality Assurance");**

**in.nextLine();**

**opt3=in.next();**

**System.out.println("You have chosen"+opt1+","+opt2+","+opt3);**

**if(opt1.equalsIgnoreCase("Web Services"))**

**al.get(i).e.add(new Elective("IT450",opt1,3));**

**if(opt1.equalsIgnoreCase("Software Architecture"))**

**al.get(i).e.add(new Elective("IT451",opt1,3));**

**if(opt1.equalsIgnoreCase("Computer Architecture"))**

**al.get(i).e.add(new Elective("IT452",opt1,3));**

**if(opt1.equalsIgnoreCase("Transaction Processing"))**

**al.get(i).e.add(new Elective("IT453",opt1,3));**

**if(opt1.equalsIgnoreCase("Software Quality Assurance"))**

**al.get(i).e.add(new Elective("IT454",opt1,3));**

**if(opt2.equalsIgnoreCase("Web Services"))**

**al.get(i).e.add(new Elective("IT450",opt2,3));**

**if(opt2.equalsIgnoreCase("Software Architecture"))**

**al.get(i).e.add(new Elective("IT451",opt2,3));**

**if(opt2.equalsIgnoreCase("Computer Architecture"))**

**al.get(i).e.add(new Elective("IT452",opt2,3));**

**if(opt2.equalsIgnoreCase("Transaction Processing"))**

**al.get(i).e.add(new Elective("IT453",opt2,3));**

**if(opt2.equalsIgnoreCase("Software Quality Assurance"))**

**al.get(i).e.add(new Elective("IT454",opt2,3));**

**if(opt3.equalsIgnoreCase("Web Services"))**

**al.get(i).e.add(new Elective("IT450",opt3,3));**

**if(opt3.equalsIgnoreCase("Software Architecture"))**

**al.get(i).e.add(new Elective("IT451",opt3,3));**

**if(opt3.equalsIgnoreCase("Computer Architecture"))**

**al.get(i).e.add(new Elective("IT452",opt3,3));**

**if(opt3.equalsIgnoreCase("Transaction Processing"))**

**al.get(i).e.add(new Elective("IT453",opt3,3));**

**if(opt3.equalsIgnoreCase("Software Quality Assurance"))**

**al.get(i).e.add(new Elective("IT454",opt3,3));**

**System.out.println("Discrepancies Resolved for "+al.get(i).rollno);**

**al.get(i).desc=true;**

**}**

**}**

**break;**

**}**

**}**

**}**

**}**

**}});**

**// create our jbutton**

**JButton runSem = new JButton("Click to Run One Semester");**

**// add the listener to the jbutton to handle the "pressed" event**

**runSem.addActionListener(new ActionListener()**

**{**

**public void actionPerformed(ActionEvent e)**

**{**

**// display/center the jdialog when the button is pressed**

**//System.out.println("One Semester has Passed");**

**for (int i = 0; i < al.size(); i++) {**

**int cr=0;**

**switch(al.get(i).semno){**

**case 1:for (int j = 0; j < s1.size(); j++) {**

**cr+=s1.get(j).credits;**

**}**

**al.get(i).update(cr);**

**System.out.println("One Semester has Passed");**

**break;**

**case 2:for (int j = 0; j < s2.size(); j++) {**

**cr+=s2.get(j).credits;**

**}**

**al.get(i).update(cr);**

**System.out.println("One Semester has Passed");**

**break;**

**case 3:for (int j = 0; j < s3.size(); j++) {**

**cr+=s3.get(j).credits;**

**}**

**al.get(i).update(cr);**

**System.out.println("One Semester has Passed");**

**break;**

**case 4:for (int j = 0; j < s4.size(); j++) {**

**cr+=s4.get(j).credits;**

**}**

**for(int k = 0; k < al.get(i).e.size(); k++) {**

**cr+=al.get(i).e.get(k).credits;**

**}**

**if(al.get(i).credits+cr>=89){**

**al.get(i).update(cr);**

**al.get(i).e.clear();**

**System.out.println("One Semester has Passed");**

**al.get(i).desc=false;**

**}**

**else**

**System.out.println("Credit Discrepancy for "+ al.get(i).rollno);**

**break;**

**case 5:for (int j = 0; j < s5.size(); j++) {**

**cr+=s5.get(j).credits;**

**}**

**for(int k = 0; k < al.get(i).e.size(); k++) {**

**cr+=al.get(i).e.get(k).credits;**

**}**

**if(al.get(i).credits+cr>=111){**

**al.get(i).update(cr);**

**al.get(i).e.clear();**

**System.out.println("One Semester has Passed");**

**al.get(i).desc=false;**

**}**

**else**

**System.out.println("Credit Discrepancy for "+ al.get(i).rollno);**

**break;**

**case 6:for (int j = 0; j < s6.size(); j++) {**

**cr+=s6.get(j).credits;**

**}**

**for(int k = 0; k < al.get(i).e.size(); k++) {**

**cr+=al.get(i).e.get(k).credits;**

**}**

**if(al.get(i).credits+cr>=132){**

**al.get(i).update(cr);**

**al.get(i).e.clear();**

**System.out.println("One Semester has Passed");**

**al.get(i).desc=false;**

**}**

**else**

**System.out.println("Credit Discrepancy for "+ al.get(i).rollno);**

**break;**

**case 7:for (int j = 0; j < s7.size(); j++) {**

**cr+=s7.get(j).credits;**

**}**

**for(int k = 0; k < al.get(i).e.size(); k++) {**

**cr+=al.get(i).e.get(k).credits;**

**}**

**if(al.get(i).credits+cr>=150){**

**al.get(i).update(cr);**

**al.get(i).e.clear();**

**System.out.println("One Semester has Passed");**

**al.get(i).desc=false;**

**}**

**else**

**System.out.println("Credit Discrepancy for "+ al.get(i).rollno);**

**break;**

**case 8:for (int j = 0; j < s8.size(); j++) {**

**cr+=s8.get(j).credits;**

**}**

**for(int k = 0; k < al.get(i).e.size(); k++) {**

**cr+=al.get(i).e.get(k).credits;**

**}**

**if(al.get(i).credits+cr>=165){**

**al.get(i).update(cr);**

**al.get(i).e.clear();**

**System.out.println("One Semester has Passed");**

**al.get(i).desc=false;**

**}**

**else**

**System.out.println("Credit Discrepancy for "+ al.get(i).rollno);**

**break;**

**case 9:System.out.println( al.get(i).rollno + " has graduated");**

**al.remove(i);**

**break;**

**}**

**}**

**}**

**});**

**JButton exitPr = new JButton("Click to Exit Program");**

**lab=new JLabel("");**

**// add the listener to the jbutton to handle the "pressed" event**

**exitPr.addActionListener(new ActionListener()**

**{**

**public void actionPerformed(ActionEvent e)**

**{**

**// display/center the jdialog when the button is pressed**

**//System.out.println("Exiting program");**

**System.out.println("Program Successfully terminated");**

**System.exit(0);**

**}**

**});**

**// put the button on the frame**

**frame.getContentPane().setLayout(new FlowLayout());**

**frame.add(addStudent);**

**frame.add(searchStudent);**

**frame.add(resolve);**

**frame.add(runSem);**

**frame.add(exitPr);**

**frame.add(lab);**

**// set up the jframe, then display it**

**frame.setDefaultCloseOperation(WindowConstants.EXIT\_ON\_CLOSE);**

**frame.setPreferredSize(new Dimension(300, 300));**

**frame.pack();**

**frame.setLocationRelativeTo(null);**

**frame.setVisible(true);**

**}**

**}**

**Flow of the program**

Will insert the student name into the arraylist

Will search the student with given roll no

Resolve the discrepancy in credits (if the student does not choose any elective after third semester)

Run to the next semester by adding the credits of the previous semester into total credits.

Exit the program flow