UNIVERSITY OF WESTMINSTER

INFORMATICS INSTITUTE OF TECHNOLOGY

Informatics Institute of Technology
Department of Computing

ESCI410 – Software Development Principles 01
Coursework 1

Module Leader – Mr.Guhanathan Poravi

Student ID :16084799/1
First Name: Thanthirige
Last Name : KUMARA
Date of Submission : 03 /04/2016

UNIVERSITY OF WESTMINSTER

INFORMATICS INSTITUTE OF TECHNOLOGY

Table of Contents

Introduction .....
Functional and non functional.....
Design – Flow Charts .....
Implementation .....
Screenshots .....
Black box Testing .....
White Box Testing .....

UNIVERSITY OF WESTMINSTER

INFORMATICS INSTITUTE OF TECHNOLOGY

Introduction

This is the first course work we are doing in IIT. The aim of this course work is to make us learn javafx on our own and to practically show what we have learned.classes,objects Try catch,this,arraylist and variables etc. are used in this program.

UNIVERSITY OF WESTMINSTER

INFORMATICS INSTITUTE OF TECHNOLOGY

Functional Requirements

- **Level4 credit and marks calculation**
- **Level5,6 credit and marks calculation**
- **Condone credits**
- **CertHE**
- **DipHE**
- **Final award calculation**
- **Validation**

Non-Functional Requirements

- **Scalability**
- **Capacity**
- **Availability**
- **Reliability**
- **Recoverability**
- **Maintainability**
- **Serviceability**

UNIVERSITY OF WESTMINSTER

INFORMATICS INSTITUTE OF TECHNOLOGY

Level 5 -6 Algorithms

1. Prompt user for input ICT Exams marks of each Module.
2. Calculate module Average Marks.
3. Average of the Module = (ICT 1 +ICT 2) / 2
4. IF (Average >= 40)

    Display “PASS”

    Receive 20 Credits Per Module

   ELSE

    Display “FAIL”

5. Input Module Resit 1 marks of each ICT Exams.
6. IF (Resit 1 Average >= 40)

    Display “PASS”

    Add 20 Credits Per Module

   ELSE

    Display “FAIL”

7. Input Module Retake marks of each ICT Exams.
8. IF (Retake Average >= 40)

    Display “PASS”

    Add 20 Credits Per Module

   ELSE

    Display “FAIL”

9. Input Module Resit 2 marks of each ICT Exams.
10. IF (Resit 2 Average >= 40)

    Display “PASS”

    Add 20 Credits Per Module

    ELSE

    Display “LEAVE THE COURSE”

11. Calculate Credit for Level 5 = (Number of Pass Modules * 20)

Final Degree Calculation – Algorithms

1. Prompt User for input Level Level 5 Credits , Level 6 Credits.
2. Sum of all Level Credits = (Level 4 + Level 5 +Level6) / 3
3. IF ALL CREDIT = 360 Then
4. IF all Module (Average Marks >100)

   Display “INVALID MARKS ! ”

5. ELSE IF (Average Marks > 70 && Average Marks < 100)

   Display “1 ST CLASS HONOUR’S DEGREE ACCEPTED ! ”

6. ELSE IF (Average Marks > 60 && Average Marks < 69)

   Display “2 ND CLASS UPPER DIVISION HONOUR’S DEGREE ACCEPTED ! ”

7. ELSE IF (Average Marks > 70 && Average Marks < 100)

   Display “2 ND CLASS LOWER DEVISION HONOUR’S DEGREE ACCEPTED ! ”

8. ELSE IF (Average Marks > 70 && Average Marks < 100)

   Display “3 RD CLASS HONOUR’S DEGREE ACCEPTED ! ”

9. IF (ALL CREDIT <= 300 && ALL CREDIT <= 360)

   Display “NON HONOUR’S DEGREE ACCEPTED ! ”

   ELSE

   Display “YOU HAVE TO DO YOUR MODULES AGAIN PROPERLY ! ”

Validate Marks – Algorithms

1. Prompt User for input ICT Exam Marks.
2. IF (ICT MARKS > 0 && ICT MARKS 100)

   Display " MARKS VALIEDE ! "

3. ELSE
   Display " MARKS NOT VALIEDE'S ! "

Level 4 Algorithms

1. Prompt user for input ICT Exams marks of each Module.
2. Calculate Average Marks of ICT Marks.
3. Average Marks = (ICT 1 +ICT 2) / 2
4. IF (Average Marks >= 40)

   Display "PASS"

   Receive 20 Credits Per Module

   ELSE IF (Average Marks > 30 && Average Marks < 39)

   Display "RE-SIT"

5. IF you give Condoned for this Module, Then
   Display " CONDONED "

   IF (Average Marks < 30)

   Display "FAIL"

6. Input Module Resit 1 marks of each ICT Exams.
7. IF (Resit 1 Average >= 40)

   Display "PASS"

   Add 20 Credits Per Module

ELSE

Display “FAIL”

8. Input Module Retake marks of each ICT Exams.
9. IF (Retake Average >= 40)

Display “PASS”

Add 20 Credits Per Module

ELSE

Display “FAIL”

10. Input Module Resit 2 marks of each ICT Exams.
11. IF (Resit 2 Average >= 40)

Display “PASS”

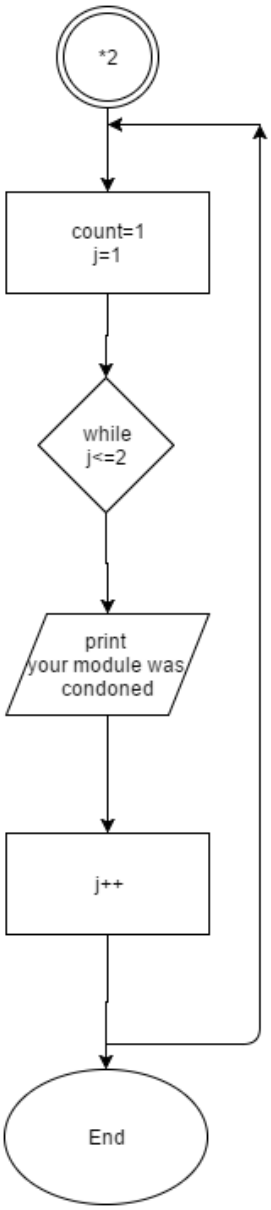
Add 20 Credits Per Module

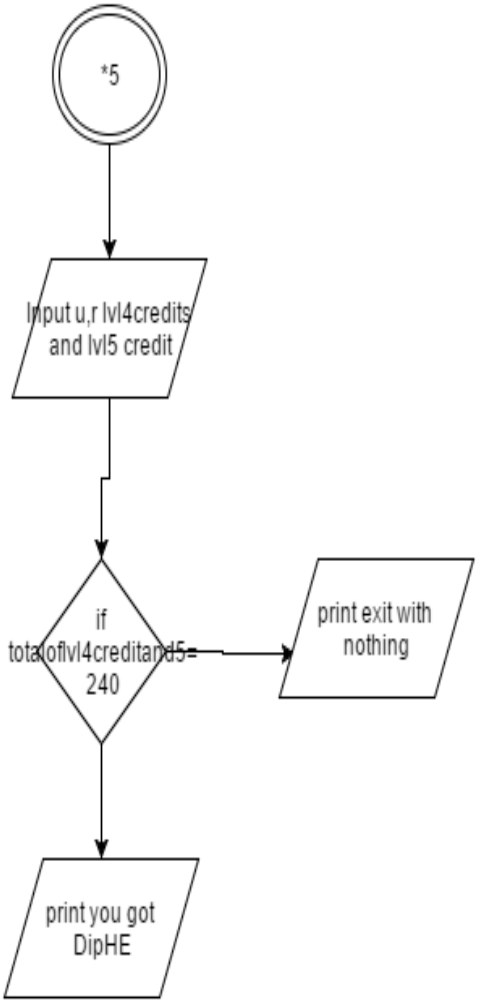
ELSE

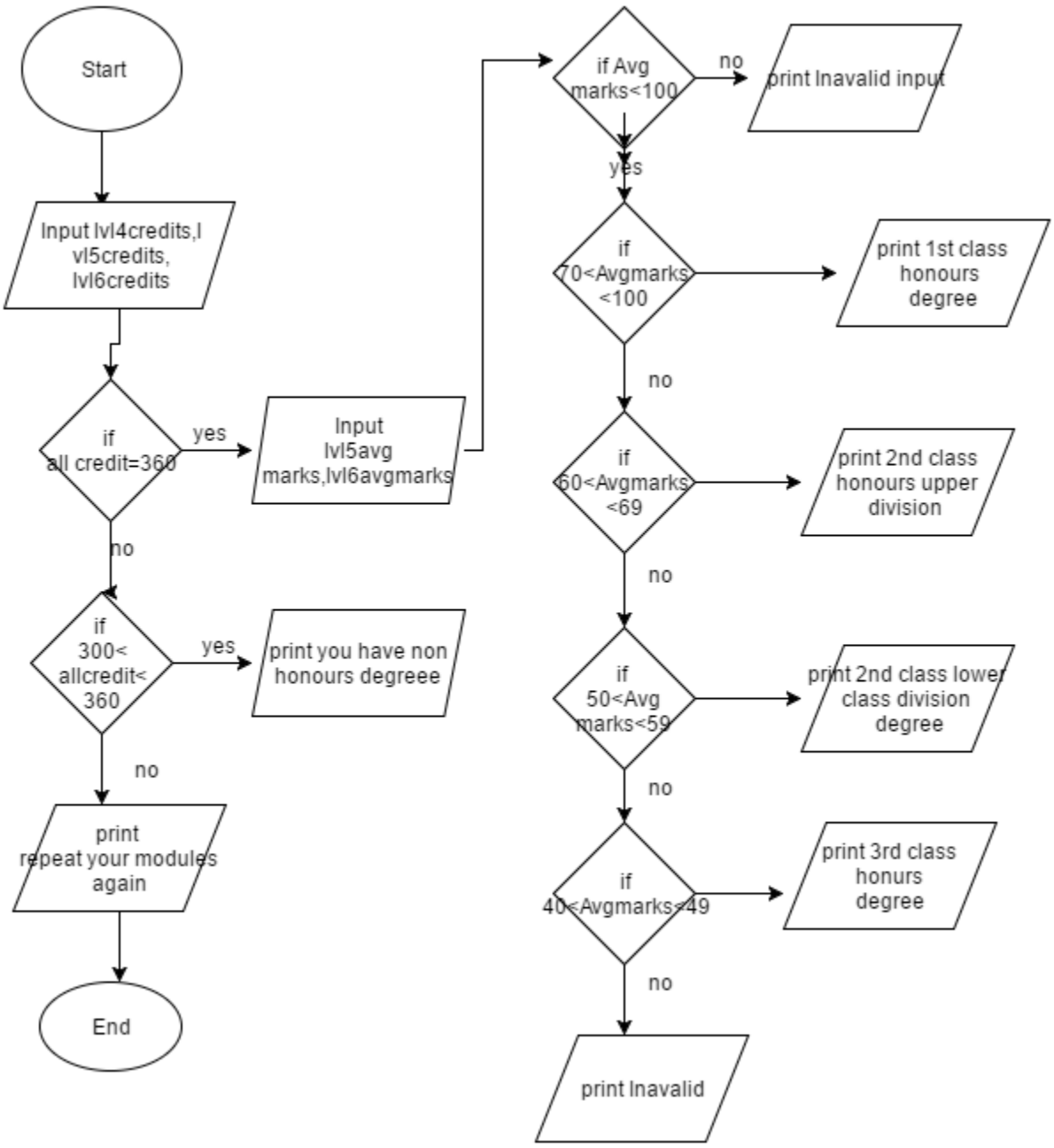
Display “LEAVE THE COURSE”

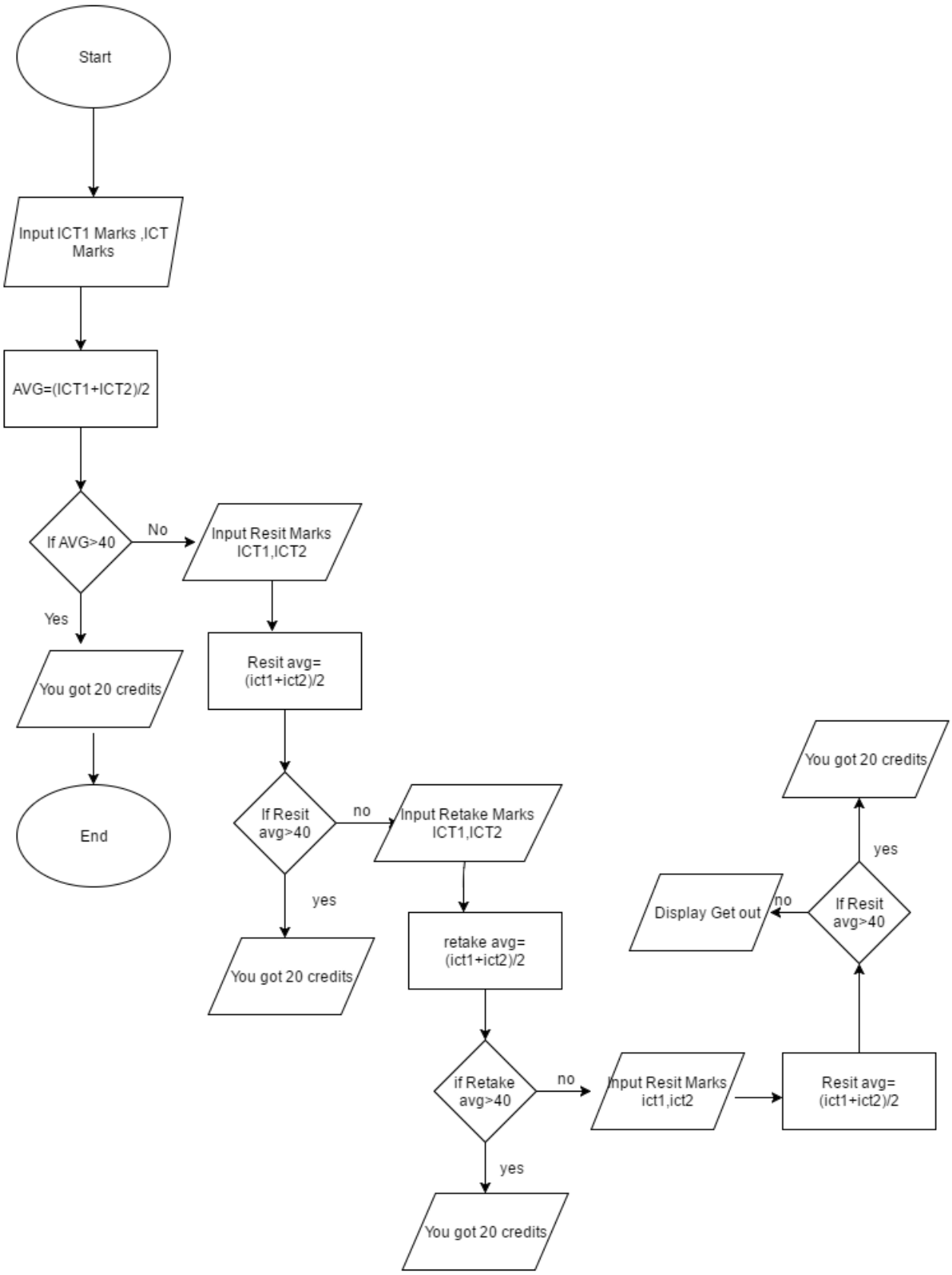
12. Prompt User for CONDONED CREDITS
13. Calculate Credit for Level 5

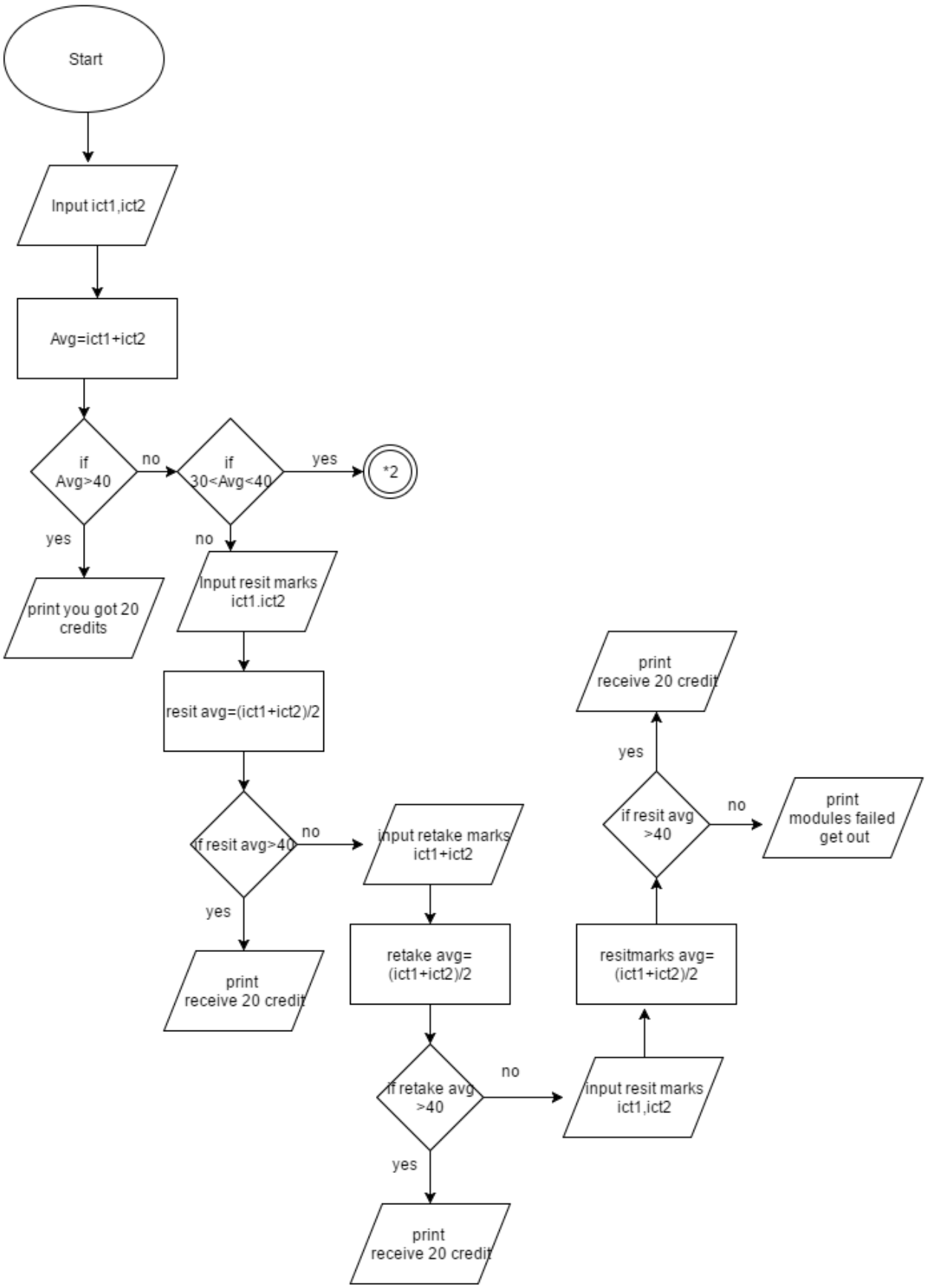
Credits = (Number of Pass Modules*20 + CONDONED)

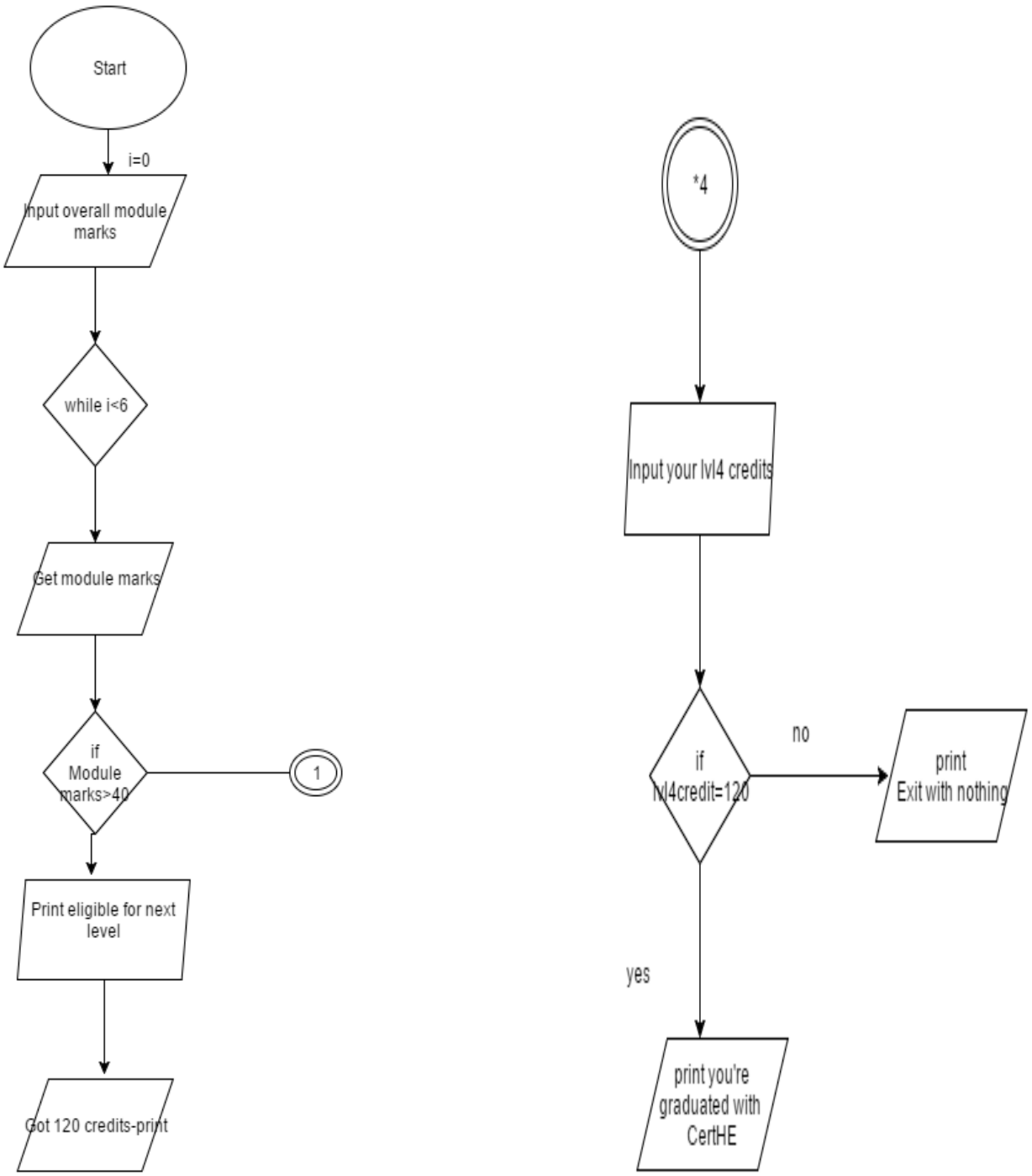

*2
count=1
j=1
while
j<=2
print
your module was
condoned
j++
End


*5
Input u,r lvl4credits
and lvl5 credit
if
totalofIvl4creditand5=
240
print exit with
nothing
print you got
DipHE

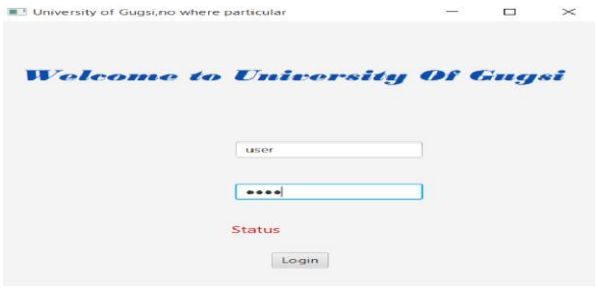

Start
Input lvl4credits,lvl5credits, lvl6credits
if all credit=360
yes
Input lvl5avg marks,lvl6avgmarks
no
if 300< allcredit< 360
yes
print you have non honours degreee
no
print repeat your modules again
End
if Avg marks<100
no
print Inavalid input
yes
if 70<Avgmarks <100
print 1st class honours degree
no
if 60<Avgmarks <69
print 2nd class honours upper division
no
if 50<Avg marks<59
print 2nd class lower class division degree
no
if 40<Avgmarks<49
print 3rd class honurs degree
no
print Inavalid

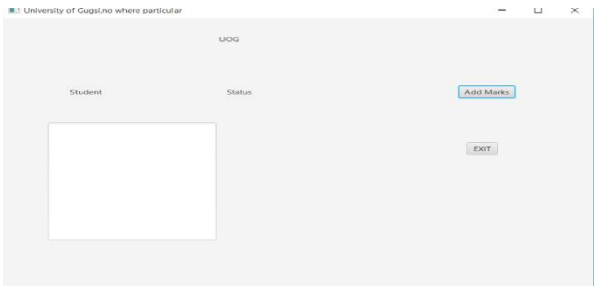

Start
Input ICT1 Marks ,ICT Marks
AVG=(ICT1+ICT2)/2
If AVG>40
No
Input Resit Marks ICT1,ICT2
Yes
You got 20 credits
End
Resit avg= (ict1+ict2)/2
If Resit avg>40
no
Input Retake Marks ICT1,ICT2
yes
You got 20 credits
retake avg= (ict1+ict2)/2
if Retake avg>40
no
Input Resit Marks ict1,ict2
Resit avg= (ict1+ict2)/2
If Resit avg>40
yes
You got 20 credits
no
Display Get out
yes
You got 20 credits

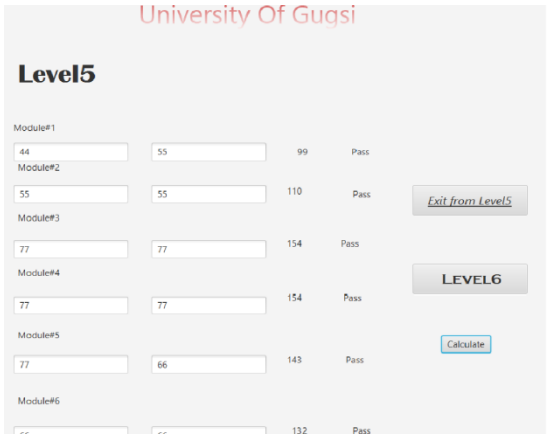

Start
Input ict1,ict2
Avg=ict1+ict2
if Avg>40
no
if 30<Avg<40
yes
*2
yes
print you got 20 credits
no
Input resit marks ict1.ict2
resit avg=(ict1+ict2)/2
if resit avg>40
no
input retake marks ict1+ict2
yes
print receive 20 credit
retake avg= (ict1+ict2)/2
if retake avg >40
no
input resit marks ict1,ict2
yes
print receive 20 credit
resitmarks avg= (ict1+ict2)/2
if resit avg >40
no
print modules failed get out
yes
print receive 20 credit

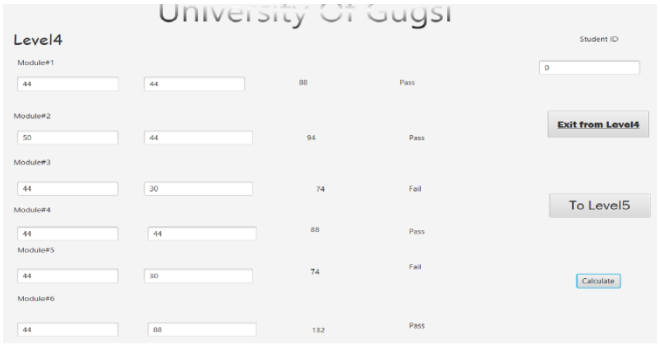

Start
i=0
Input overall module marks
while i<6
Get module marks
if Module marks>40
1
Print eligible for next level
Got 120 credits-print
*4
Input your lvl4 credits
if lvl4credit=120
no
print Exit with nothing
yes
print you're graduated with CertHE

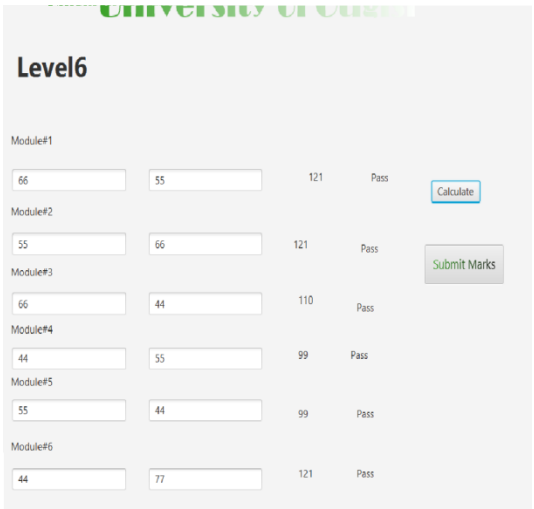
Screenshots

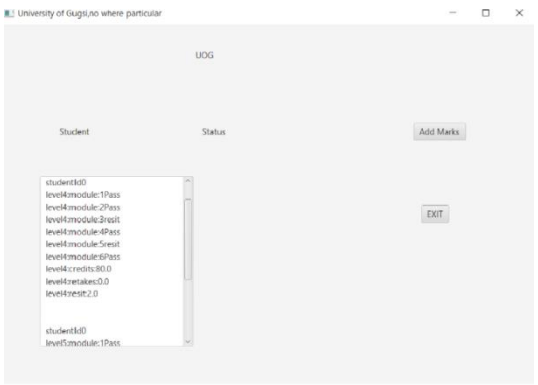


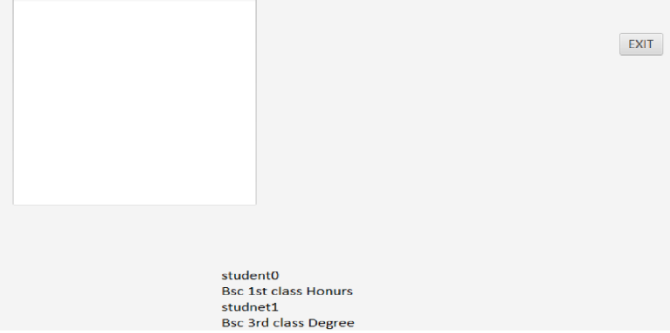












BlackBox Testing

INPUT	EXPECTED Output	ACTUAL Output
45 85 79	Average = 69% , Pass	Average = 69% , Pass
80 80 20	Average = 60% ,L4- Pass, con L5/L6-Resit	Average = 60% ,L4- Pass, con L5/L6-Resit
10 10 10	Average = 10% ,Retake	Average = 10% ,Retake
40 40 40	Average = 40% , Pass	Average = 40% , Pass
39 39 39	Average = 39% , L4-Resit, con L5/L6-Resit	Average = 39% , L4-Resit, con L5/L6-Resit
100 100 100	Average = 100% , Pass	Average = 100% , Pass
Retake L04:-85 99 78	Average = 40% , L5/L6-Pass	Average = 40% , L5/L6-Pass
45 85 79 **10 10 10** **100 100 100**	L4-Credits=60, Con=40, Credits=80, Final credits=100 Credits=60, Final credits=80	L4-Credits=60, Con=40, L5-Total = 269, Minimum = 40, Credits=60, Final credits=80
Credits=L4-100 L5-120 L6-140 **Total=L5-350 L6-400** **Minimum=L5-35 L6-50** **Retake=L5-40 L6-40**	Credits=360, Average=79, Minimum=35, Degree=B.Sc. Honors, Degree Class=1st class Honors, Display " 1st class Honors DEGREE.	Credits=360, Average=79, Minimum=35, Degree=B.Sc. Honors, Degree Class=1st class Honors, Display " 1st class Honors DEGREE.

White Box Testing

CONDITION	YES	NO
If user name=="user"&&	Login to the program.	"Login failed "error shown .

password==”pass”		
If (marks1<=100&&marks2<=100&& marks3<=100)	Move to the next condition.	Display “invalid”
If ((marks1 < 40 && marks2 < 40) || (marks2 < 40 && marks3 < 40) || (marks1 < 40 && marks3 < 40))	Display “”	Move to the next condition.
If(marks1<40||marks2<40 ||marks3<40)	Display “Condoned”	Move to the next condition.
If (module average>=40 &&(marks1<40||marks2<40 ||marks3<40))	Display “pass” , “Condoned”	Move to the next condition.
If module average>=40	Display “Pass”	Move to the next condition.
If module average>=30	Display “Resit” and “condoned”	Move to the next condition.
If module average<30	Display “Retake”	-.
If final credits==120	Can move to the next level.	Can’t move to the next level.
If final credits==100	Can move to the next level. Display “you have to Retake your failure module”	Can’t move to the next level
If final credits<100	Can’t move to the next level.	-.
If total credits==360	Display ”Honors degree”	Move to the next condition.
If total average marks>=70	Display “1st class honors degree”	Move to the next condition.
If total average marks>=60	Display “2nd class honors upper division degree”	Move to the next condition.
If total average marks>=50	Display “2nd class honors lower division degree”	Move to the next condition.
If total average marks>=40	Display “3rd class honors degree”	Display “Non Honurs Degree ”

Table 1.2

```
//credit calculation
                    double totalMarks=0;
                    double totalCredits=0;
                    if(lavl4marks!=0 && lavl5marks!=0&& lavl6marks!=0){
                    totalMarks=(lavl4marks+lavl5marks+lavl6marks)/18;

                    }
                    if(lavl4credits!=0 && lavl5credits!=0&& lavl6credits!=0){
                          totalCredits=lavl4credits+lavl5credits+lavl6credits;

                          }
```

```
                  if(totalCredits==360){

                        if(totalMarks>70){
                              sb.append("First Class");
                              sb.append("\n");
                        }
                        else if(totalMarks>60 && totalMarks<69){
                              sb.append("Second Class Upper Class");
                              sb.append("\n");
                        }
                        else if(totalMarks>50 && totalMarks<59){
                              sb.append("Second Class Lower Class");
                              sb.append("\n");
                        }
                        else if(totalMarks>40 && totalMarks<50){
                              sb.append("Third Class Class");
                              sb.append("\n");
                        }
                        else {
                              sb.append("Not Qualify For the degree");
                              sb.append("\n");
                        }

                  }
                  if(totalCredits>300&& totalCredits<360){
                        sb.append("Non Honours");
                        sb.append("\n");
                  }

                  sb.append("\n");sb.append("\n");sb.append("\n");

            }
            resultArea.setText(sb.toString());
      }

      @Override
      public void initialize(URL location, ResourceBundle resources) {
            this.setResultText();
      }



//condone
            if(failed==1){
                  total=total+condone;
            }
            if(failed==2){
                  total=total+(condone*2);
            }
            if(failed==3){
                  total=total+(condone*2);
            }

      GetStudentController.resultDictionary.put(studentId+"level4credits",Str
ing.valueOf(total));
```

```
	GetStudentController.resultDictionary.put(studentId+"level4Marks",Strin
g.valueOf(totalAvg));

	GetStudentController.resultDictionary.put(studentId+"level4retakes",Str
ing.valueOf(retake));

	GetStudentController.resultDictionary.put(studentId+"level4resit",Strin
g.valueOf(resit));

	GetStudentController.resultDictionary.put(studentId+"studentId",student
Id);

		GetStudentController.sudentList.put(String.valueOf(studentId),
GetStudentController.resultDictionary);

	}
}



//checking average and declaring all variables
		GetStudentController.resultDictionary.clear();
		double total=0;
		int failed=0;
	    double avg=0;
	    double condone=20;
	    double retake=0;
	    double resit=0;
	    double totalAvg=0;
		List<Double> avgList=Arrays.asList(av1,av2,av3,av4,av5,av6);
		for(int i=0;i<avgList.size();i++){
			avg=avgList.get(i);
			totalAvg=totalAvg+avg;
			if(avg>40){

	GetStudentController.resultDictionary.put(studentId+"level4module"+i,
"Pass");
				total=total+20;
			}
			else if(avg>30 && avg<40){

	GetStudentController.resultDictionary.put(studentId+"level4module"+i,
"resit");
				resit++;
			}else{

	GetStudentController.resultDictionary.put(studentId+"level4module"+i,
"fail");
				retake++;
			}
		}

//checking pass or fail
public void CalcTotal1(long n1,long n2,Label marks, Label results){
```

```
	long totalmarks1= n1+n2;
	if(totalmarks1/2>40&&n1>30&& n2>30){
		results.setText("Pass");
	}else{
		results.setText("Fail");
	}
	marks.setText(Long.toString(totalmarks1));
}

public double CalcAvg(long n1,long n2){
	double avg= (n1+n2)/2;

	return avg;

//putting marks into array
public void backTogetStudentSceen1(ActionEvent event)throws Exception{
	Level5Model model= new Level5Model();
	List<String> level5Result= new ArrayList<String>();
	GetStudentController.studId++;
	Stage currentstage=(Stage)toLevel6.getScene().getWindow();
	currentstage.close();

	model.CalcOverallResults(this.CalcAvg(Long.parseLong(lvl5mod1paper1.get
Text()), Long.parseLong(lvl5mod1paper2.getText())),
			this.CalcAvg(Long.parseLong(lvl5mod2paper1.getText()),
Long.parseLong(lvl5mod2paper2.getText())),
			this.CalcAvg(Long.parseLong(lvl5mod3paper1.getText()),
Long.parseLong(lvl5mod3paper2.getText())),

	this.CalcAvg(Long.parseLong(lvl5mod4paper1.getText()),
Long.parseLong(lvl5mod4paper2.getText())),

	this.CalcAvg(Long.parseLong(lvl5mod5paper1.getText()),
Long.parseLong(lvl5mod5paper2.getText())),

	this.CalcAvg(Long.parseLong(lvl5mod6paper1.getText()),
Long.parseLong(lvl5mod6paper2.getText())),GetStudentController.studId);


	//load getstudent window
	Parent root=
FXMLLoader.load(getClass().getResource("GetStudent.fxml"));
	Scene scene = new Scene(root);
	Stage lvl4Stage = new Stage();
	scene.getStylesheets().add(getClass().getResource("application.css").to
ExternalForm());
	lvl4Stage.setScene(scene);
	lvl4Stage.setTitle("University of Gugsi,no where particular");
	lvl4Stage.show();
}
}



//getting input from user
		@FXML
```

```
            public void calculateBtnClicked2(ActionEvent event){
                  System.out.println("clicked");
                  Level6Model model= new Level6Model();
                  GetStudentController SutendRecords= new
GetStudentController();
                  List<String> level6Result= new ArrayList<String>();


      if(lvl6mod1paper1.getText()!=""&&lvl6mod1paper2.getText()!=""){

      this.CalcTotal2(Long.parseLong(lvl6mod1paper1.getText()),
Long.parseLong(lvl6mod1paper2.getText()),lvl6mod1marks,lvl6mod1result);
                  }

      if(lvl6mod2paper1.getText()!=""&&lvl6mod2paper2.getText()!=""){

      this.CalcTotal2(Long.parseLong(lvl6mod2paper1.getText()),
Long.parseLong(lvl6mod2paper2.getText()),lvl6mod2marks,lvl6mod2result);
                  }

      if(lvl6mod3paper1.getText()!=""&&lvl6mod3paper2.getText()!=""){

      this.CalcTotal2(Long.parseLong(lvl6mod3paper1.getText()),
Long.parseLong(lvl6mod3paper2.getText()),lvl6mod3marks,lvl6mod3result);
                  }

      if(lvl6mod4paper1.getText()!=""&&lvl6mod4paper2.getText()!=""){

      this.CalcTotal2(Long.parseLong(lvl6mod4paper1.getText()),
Long.parseLong(lvl6mod4paper2.getText()),lvl6mod4marks,lvl6mod4result);
                  }

      if(lvl6mod5paper1.getText()!=""&&lvl6mod5paper2.getText()!=""){

      this.CalcTotal2(Long.parseLong(lvl6mod5paper1.getText()),
Long.parseLong(lvl6mod5paper2.getText()),lvl6mod5marks,lvl6mod5result);
                  }

      if(lvl6mod6paper1.getText()!=""&&lvl6mod6paper2.getText()!=""){

      this.CalcTotal2(Long.parseLong(lvl6mod6paper1.getText()),
Long.parseLong(lvl6mod6paper2.getText()),lvl6mod6marks,lvl6mod6result);
                  }

//etc……
```

Thank You

-T.P.T.L.Kumara-