



Informatics Institute of Technology

Department Of Computing

Module: ECSI410 – Software Development Principles 01

Module Leader: Mr. Guganathan Poravi

Coursework

Date of submission: 03/04/2016

Student ID: w1608490

Student First Name: Chamod Oshan

Student Surname: Hettigodage

Content

• Acknowledgement	02
• Introduction	02
• List of Functional Requirements	02
• Non-Functional Requirements	02
• Algorithms	02-15
• Login to the program	02
• Calculate the Level04 credits and marks	03
• Exit with CerHE	06
• Calculate the Level05 credits and marks	07-08
• Exit with DipHE	11
• Calculate Level06 credits and marks	11-12
• Final Award calculation	15
• Flow Charts	03-16
• Java Code	03-17
• Test Cases	17
• Conclusion	20

Figures

• Figure 1.0 - Flow Chart of Login	03
• Figure 1.1 - Flow Chart of Level 04	04
• Figure 1.2 - Flow Chart of Condoned	05
• Figure 1.3 - Flow Chart of CerHE	07
• Figure 1.4 - Flow Chart of Level 05	08-09
• Figure 1.5 - Flow Chart of Level 05 Final Credits	09
• Figure 1.6 - Flow Chart of DipHE	11
• Figure 1.7 - Flow Chart of Level 06	12-13
• Figure 1.8 - Flow Chart of Level 06 Final Credits	13
• Figure 1.9 - Flow Chart of Award Calculation	16
• Figure 2.0 - Screenshot of Level 04 Test 1	19
• Figure 2.1 - Screenshot of Level 04 Test 2	19
• Figure 2.2- Screenshot of Level 05 Test 3	19
• Figure 2.3- Screenshot of Level 06 Test 4	19
• Figure 2.4- Screenshot of Award Calculation Test 5	19

Table

• Table 1.0 - White Box Testing 1	17-18
• Table 1.1 - White Box Testing 2	18
• Table 1.2 - Black Box Testing	20

Acknowledgment

My sincere gratitude goes to our module leader Mr.Guhanathan Poravi for providing us with the knowledge and guiding us as we were accomplish this task. My parents for their endless support and encouragement without whom I may not be where I am today.

Introduction

This gui based java program is created as a Course work of ECSI410 – Software Development Principles 01. This program build to do Awards calculation of a university using student marks. This program designed to get exam marks for each module. After it calculate average of each module and show whether the module is Pass ,Resit or Retake. Then calculate credits in the Leve04,If credits enough to move to the next level, program will be providing the next gui . At last this program will show the degree , class of degree, by concerning about the total credits .

List of Functional Requirements

- Login to the program.
- Calculate the Level04 credits and marks.
- Exit with CerHE.
- Calculate the Level05 credits and marks.
- Exit with DipHE.
- Calculate Level06 credits and marks.
- Final Award calculation.

Non-Functional Requirements

- Design of gui.
- Number of buttons.
- Font size and font face.
- Switching animations.
- Number of windows.

Algorithms

❖ Login to the program.

- 1) Prompt user for username and password.
- 2) Get username and password.
- 3) IF (username="chamod" AND password="2015300")
Move to the next window
Else
Display "Login Failed"
ENDIF

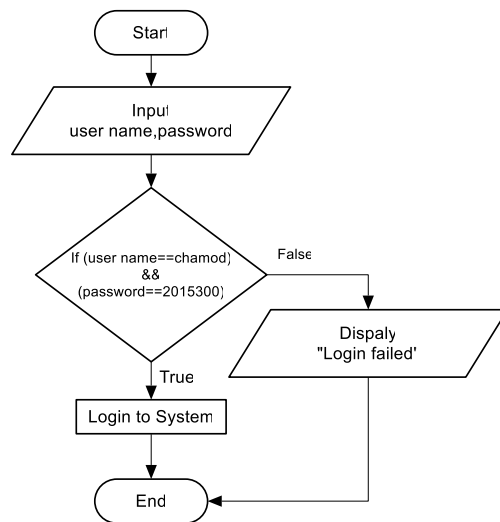


Figure 1.0

Login window code

```

17 //Login Window
18 @FXML
19 private Label lblvalid;
20 @FXML
21 private TextField txtuser,txtpass;
22
23 public void Login(ActionEvent event) throws Exception{
24     if(txtuser.getText().equals("chamod") && txtpass.getText().equals("2015300")){
25         Stage primaryStage=new Stage();
26         Parent root =FXMLLoader.load(getClass().getResource("/application/Level04.fxml"));
27         Scene scene = new Scene(root);
28         scene.getStylesheets().add(getClass().getResource("application.css").toExternalForm());
29         primaryStage.setScene(scene);
30         primaryStage.show();
31     }else{
32         lblvalid.setText("Login Failed");
33     }
34 }
35
36 //Cancel Button
37 @FXML
38 private Button btncancel;
39 @FXML
40 private void cancelAction(){
41     Stage primaryStage = (Stage) btncancel.getScene().getWindow();
42     primaryStage.close();
43 }

```

❖ Calculate the Level04 credits and marks.

- 1) Prompt user for marks of each ICT exams.
- 2) Calculate Total marks of each module separately.
- 3) Calculate Average marks of each module separately.
- 4) IF(Average>=40)
 - Display "pass"
- ELSEIF(Average>=30)
 - Display "resit"
- ELSE
 - Display "retake"
- ENDIF
- 5) If can give condoned, Display "condoned".
- 6) Prompt user for condoned credit.
- 7) Calculate credits = (Number of pass modules*20+condoned).

8) If get enough credits move to the Level05.

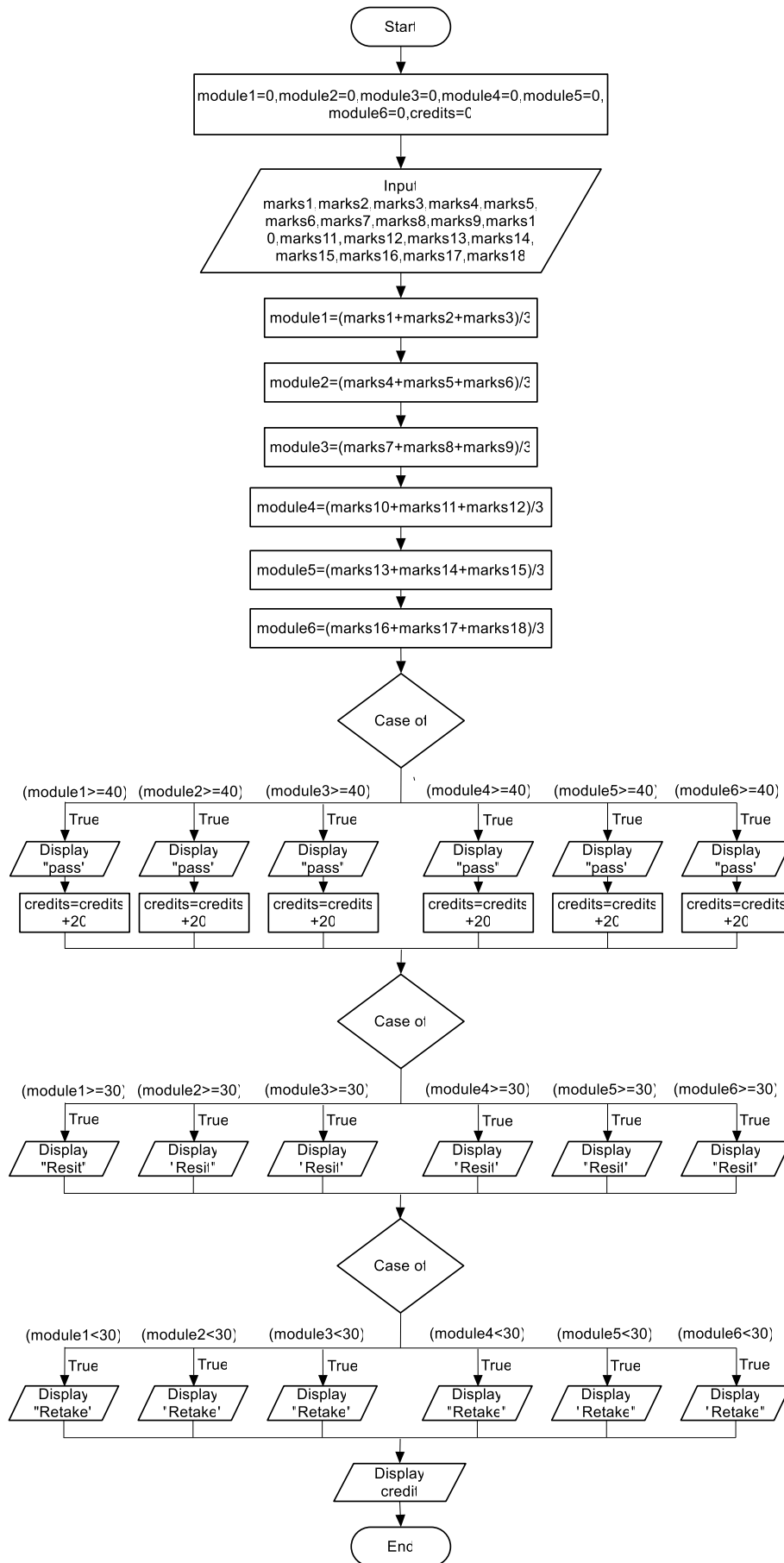


Figure 1.1

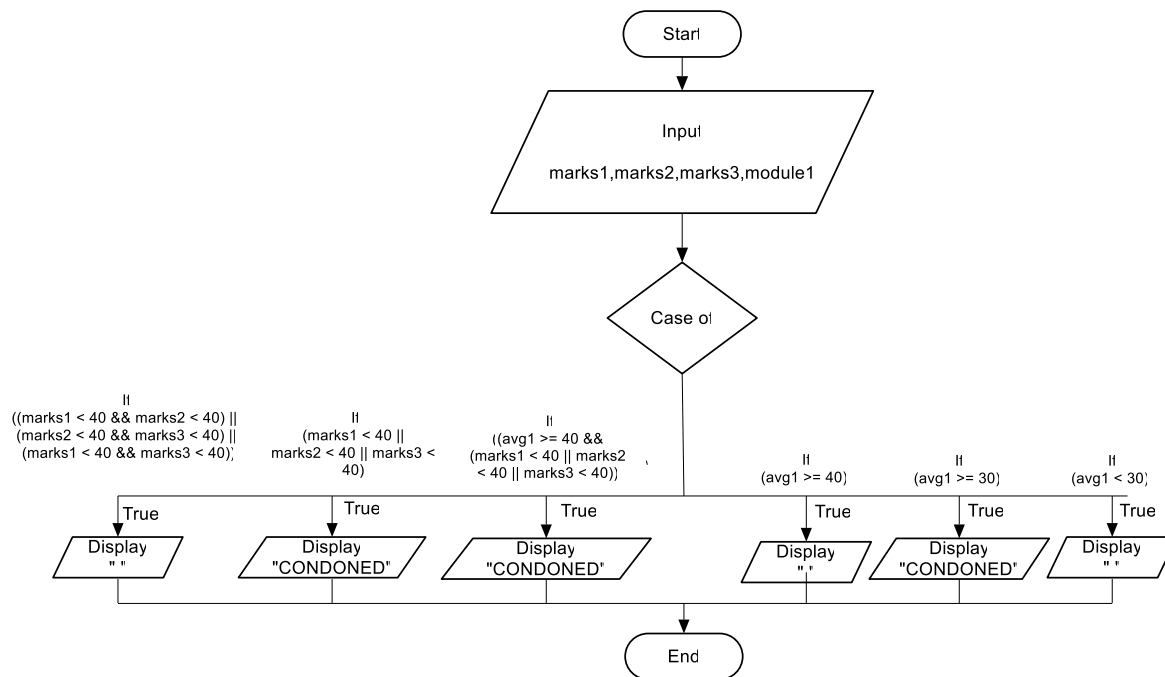


Figure 1.2

```

45 // Level 04 Modules Window
46 @FXML
47 private TextField L4M1Ict01, L4M1Ict02, L4M1Ict03, L4M2Ict01, L4M2Ict02, L4M2Ict03, L4M3Ict01, L4M3Ict02, L4M3Ict03,
48 L4M4Ict01, L4M4Ict02, L4M4Ict03, L4M5Ict01, L4M5Ict02, L4M5Ict03, L4M6Ict01, L4M6Ict02, L4M6Ict03, txtcon;
49 @FXML
50 private Label lblM1, lblM2, lblM3, lblM4, lblM5, lblM6, lblP1, lblP2, lblP3, lblP4, lblP5, lblP6, lblcon1, lblcon2,
51 lblcon3, lblcon4, lblcon5, lblcon6, lblcredit, lblmessage, lblmessage1, lblce;
52
53 public void getinputs(ActionEvent event) {
54 // Module-01
55 int marks1 = Integer.parseInt(L4M1Ict01.getText());
56 int marks2 = Integer.parseInt(L4M1Ict02.getText());
57 int marks3 = Integer.parseInt(L4M1Ict03.getText());
58 int total1 = 0, avg1 = 0;
59 if (marks1<=100&&marks2<=100&&marks3<=100){
60 total1 = (marks1 + marks2 + marks3);
61 avg1 = total1 / 3;
62 lblM1.setText(Integer.toString(avg1));
63 if ((marks1 < 40 && marks2 < 40) || (marks2 < 40 && marks3 < 40) || (marks1 < 40 && marks3 < 40)) {
64 lblcon1.setText("");
65 } else if (marks1 < 40 || marks2 < 40 || marks3 < 40) {
66 lblcon1.setText("CONDONED");
67 }
68 if (avg1 >= 40 && (marks1 < 40 || marks2 < 40 || marks3 < 40)) {
69 lblP1.setText("PASS");
70 lblcon1.setText("CONDONED");
71 } else if (avg1 >= 40) {
72 lblP1.setText("PASS");
73 lblcon1.setText("");
74 } else if (avg1 >= 30) {
75 lblP1.setText("RESIT");
76 lblcon1.setText("CONDONED");
77 } else if (avg1 < 30) {
78 lblP1.setText("RETAKE");
79 lblcon1.setText("");
80 }
81 }else{
82 lblP1.setText("Invalid");
83 }
  
```

```

235     int[] marks = new int[6];
236     marks[0] = avg1;
237     marks[1] = avg2;
238     marks[2] = avg3;
239     marks[3] = avg4;
240     marks[4] = avg5;
241     marks[5] = avg6;
242     int credit = 0;
243     for (int i = 0; i < marks.length; i++) {
244         if (marks[i] >= 30) {
245             credit = credit + 20;
246             lblce.setText(Integer.toString(credit));
247         }
248     }
249
250
251     String[] con = new String[6];
252     con[0] = lblcon1.getText();
253     con[1] = lblcon2.getText();
254     con[2] = lblcon3.getText();
255     con[3] = lblcon4.getText();
256     con[4] = lblcon5.getText();
257     con[5] = lblcon6.getText();
258     String CON = "CONDONED";
259     for (int j = 0; j < con.length; j++) {
260         if (con[j] == CON) {
261             credit = credit - 20;
262             if (credit >= 0) {
263                 lblce.setText(Integer.toString(credit));
264             }
265         }
266     }

```

```

299 public void finish(ActionEvent event) {
300     int credit = Integer.parseInt(lblce.getText());
301     int con = Integer.parseInt(txtcon.getText());
302     finalCredit4 = credit + con;
303     lblcredit.setText(Integer.toString(finalCredit4));
304     if (finalCredit4 > 120) {
305         lblmessage.setText("Invalid");
306     } else if (finalCredit4 == 120) {
307         lblmessage.setText("You are eligible for move to the Level-05");
308         lblmessage1.setText("");
309     } else if (finalCredit4 == 100) {
310         lblmessage.setText("You are eligible for move to the Level-05");
311         lblmessage1.setText("but you have to RETAKE your failure Module");
312     } else if (finalCredit4 < 100) {
313         lblmessage.setText("You are not eligible for move to the Level-05");
314         lblmessage1.setText("");
315     }
316
317 }

```

```

291 public void next1(ActionEvent event) throws Exception {
292     if (lblmessage.getText().equals("You are eligible for move to the Level-05")) {
293         Stage primaryStage = new Stage();
294         Parent root = FXMLLoader.load(getClass().getResource("/application/Level05.fxml"));
295         Scene scene = new Scene(root);
296         scene.getStylesheets().add(getClass().getResource("application.css").toExternalForm());
297         primaryStage.setScene(scene);
298         primaryStage.show();
299     }
300 }

```

❖ Exit with CerHE.

- 1) Read Level04 credits.
- 2) IF(credits=120)
 - Display CerHE details.
 - ENDIF

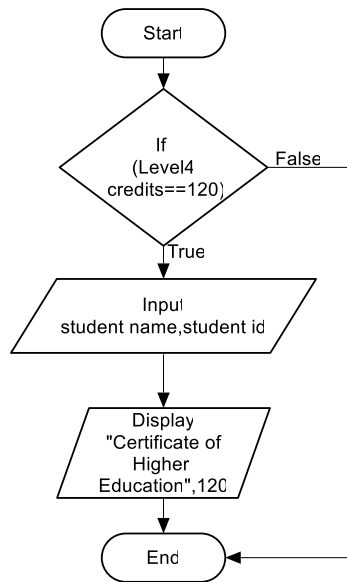


Figure 1.3

```

471 //cerhe
472 @FXML
473 private Label lblDipTopic, lblDiphe1, lblDiphe2, lblDiphe3, lblDiphe4, lblDiphe5, lblDiphe6;
474 @FXML
475 private TextField txtNamediphe, txtIdDiphe;
476
477 public void diphe(ActionEvent event) throws Exception {
478     if (L5lblmessage.getText().equals("You are eligible for move to the Level-06") &&
479         L5lblmessage1.getText().equals("") ||
480         L5lblmessage1.getText().equals("and you passed the RETAKE module of Level-04")) {
481         Stage primaryStage = new Stage();
482         Parent root = FXMLLoader.load(getClass().getResource("/application/diphe.fxml"));
483         Scene scene = new Scene(root);
484         scene.getStylesheets().add(getClass().getResource("application.css").toExternalForm());
485         primaryStage.setScene(scene);
486         primaryStage.show();
487     }
488 }
489
490 public void Proceed1(ActionEvent event) throws Exception {
491     lblCerTopic.setText("Certificate of Higher Education");
492     lblCerhe1.setText("120");
493     lblCerhe2.setText("---");
494     lblCerhe3.setText("---");
495     lblCerhe4.setText("No Degree");
496     lblCerhe5.setText("---");
497     lblCerhe6.setText("YOU HAVE THE CERTIFICATE OF HIGHER EDUCATION");
498     lblNameCerhe.setText(name);
499     lblIdCerhe.setText(Integer.toString(id));
500 }

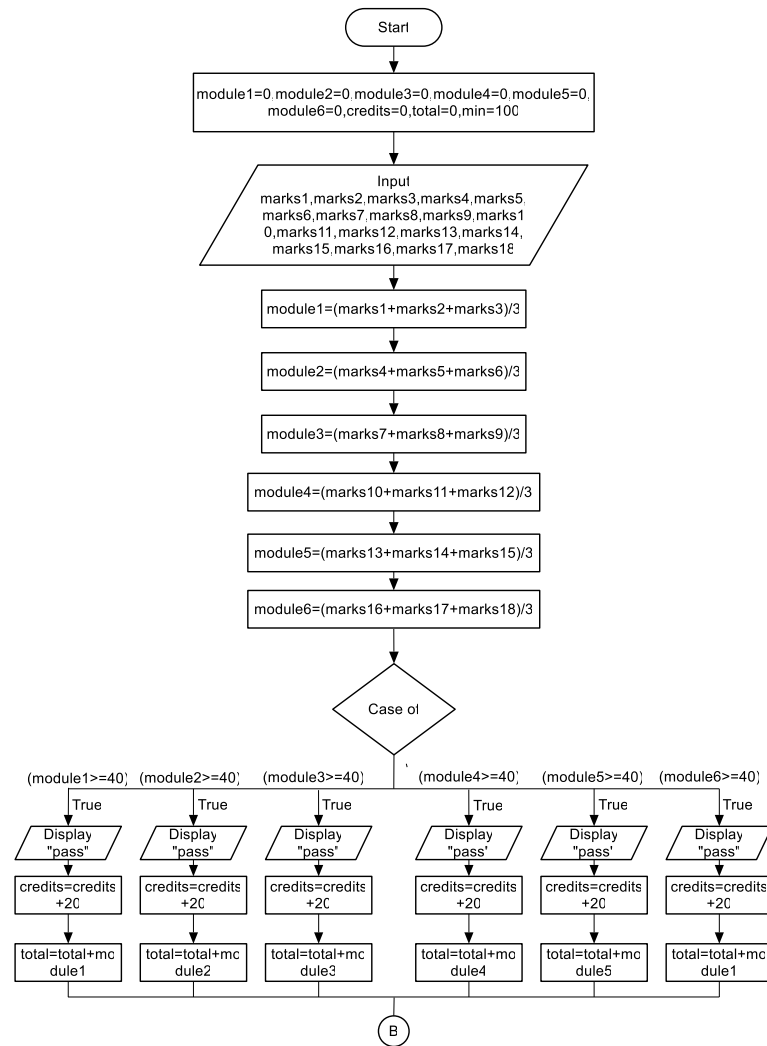
```

❖ Calculate the Level05 credits and marks.

- 1) Prompt user for marks of each ICT exams.
- 2) Calculate Total marks of each module separately.
- 3) Calculate Average marks of each module separately.
- 4) IF(Average >= 40)
 - Display "pass"
 - ELSEIF(Average >= 30)
 - Display "resit"
 - ELSE
 - Display "retake"

ENDIF

- 5) If have a failure module of Level04 Prompt user for retake marks.
- 6) Calculate retake module credits.
- 7) Calculate credits = (Number of pass modules*20+retake credits).
- 8) If get enough credits move to the Level05.



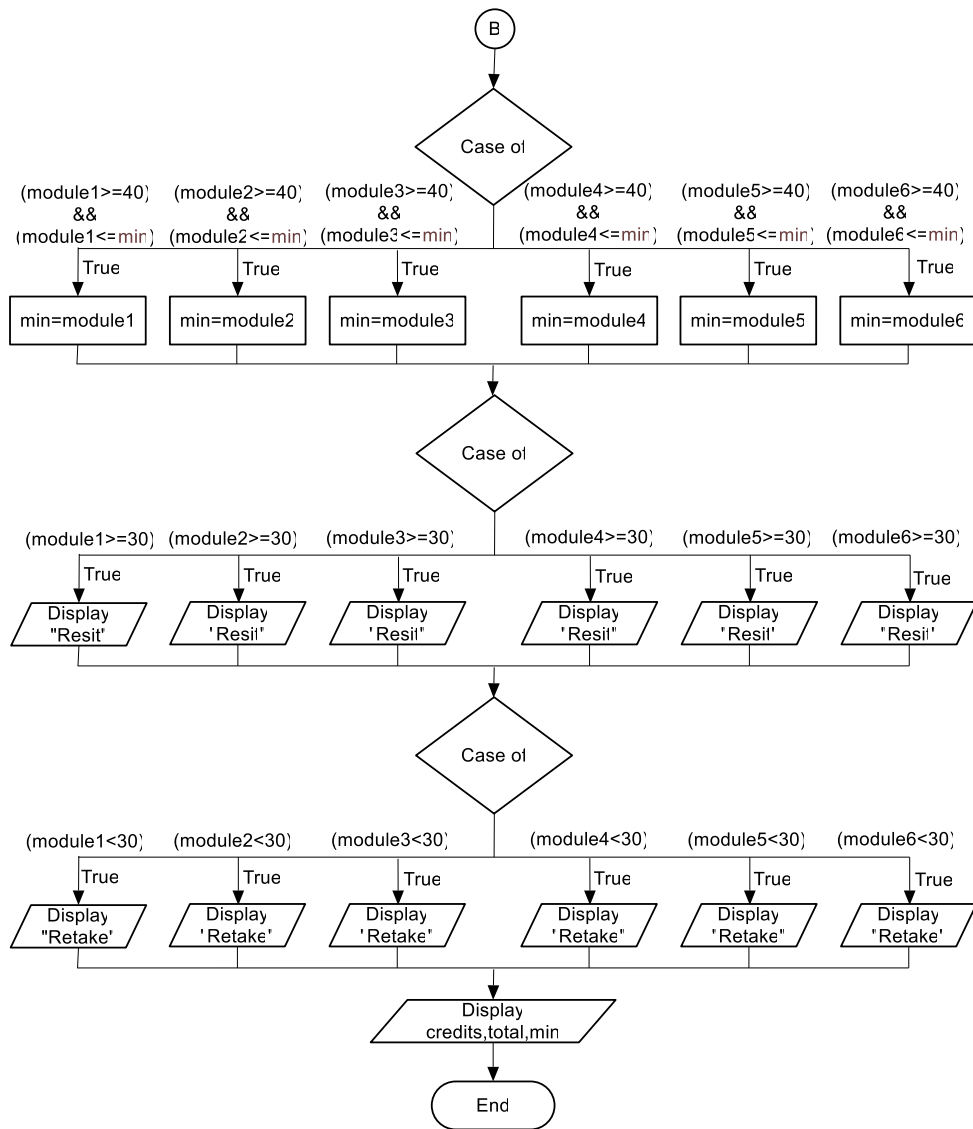


Figure 1.4

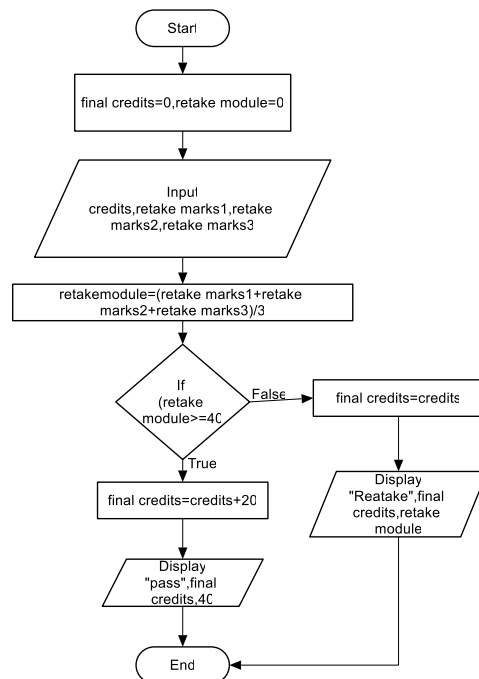


Figure 1.5

```

332 // Level 05 Modules Window
333 @FXML
334 private TextField L5M1Ict01, L5M1Ict02, L5M1Ict03, L5M2Ict01, L5M2Ict02, L5M2Ict03, L5M3Ict01, L5M3Ict02, L5M3Ict03,
335 L5M4Ict01, L5M4Ict02, L5M4Ict03, L5M5Ict01, L5M5Ict02, L5M5Ict03, L5M6Ict01, L5M6Ict02, L5M6Ict03, Ict01Re,
336 Ict02Re, Ict03Re;
337
338 @FXML
339 private Label L5blM1, L5blM2, L5blM3, L5blM4, L5blM5, L5blM6, lblRe, L5blP1, L5blP2, L5blP3, L5blP4,
340 L5blP5, L5blP6, lblPRE, L5blcredit, L5blmessage, L5blmessage1, L5blce, L5blmin, L5bltotal;
341
342 public void getinputs1(ActionEvent event) {
343 // Module-01
344 int marks1 = Integer.parseInt(L5M1Ict01.getText());
345 int marks2 = Integer.parseInt(L5M1Ict02.getText());
346 int marks3 = Integer.parseInt(L5M1Ict03.getText());
347 int total1 = 0, avg1 = 0;
348 if (marks1<=100&&marks2<=100&&marks3<=100){
349 total1 = (marks1 + marks2 + marks3);
350 avg1 = total1 / 3;
351 L5blM1.setText(Integer.toString(avg1));
352 if (avg1 >= 40 && (marks1 < 40 || marks2 < 40 || marks3 < 40)) {
353 L5blP1.setText("RESIT");
354 }else if (avg1 >= 40) {
355 L5blP1.setText("PASS");
356 } else if (avg1 >= 30) {
357 L5blP1.setText("RESITE");
358 } else if (avg1 < 30) {
359 L5blP1.setText("RETAKE");
360 }
361 }else{
362 L5blP1.setText("Invalid");
363 }
}

496 int[] marks = new int[6];
497 marks[0] = avg1;
498 marks[1] = avg2;
499 marks[2] = avg3;
500 marks[3] = avg4;
501 marks[4] = avg5;
502 marks[5] = avg6;
503 int credit = 0;
504 totalL5 = 0;
505 for (int i = 0; i < marks.length; i++) {
506 if (marks[i] >= 40) {
507 credit = credit + 20;
508 L5blce.setText(Integer.toString(credit));
509 }
510 if (marks[i] >= 40) {
511 if (marks[i] <= min5) {
512 min5 = marks[i];
513 L5blmin.setText(Integer.toString(min5));
514 }
515 totalL5 = totalL5 + marks[i];
516 L5bltotal.setText(Integer.toString(totalL5));
517 }
518 }
519 }
520 }
521 }

522 public void finish1(ActionEvent event) {
523 int credit = Integer.parseInt(L5blce.getText());
524 int marks01 = Integer.parseInt(Ict01Re.getText());
525 int marks02 = Integer.parseInt(Ict02Re.getText());
526 int marks03 = Integer.parseInt(Ict03Re.getText());
527 int total0 = 0; // avg0 = 0;
528 if (marks01<=100&&marks02<=100&&marks03<=100){
529 total0 = (marks01 + marks02 + marks03);
530 avg0 = total0 / 3;
531 lblRe.setText(Integer.toString(avg0));
532 if (avg0 >= 40) {
533 lblPRE.setText("PASS");
534 lblRe.setText(Integer.toString(40));
535 } else if (avg0 >= 30) {
536 lblPRE.setText("RESITE");
537 } else if (avg0 < 30) {}
538 lblPRE.setText("RETAKE");
539 }
540 if (avg0 >= 40) {
541 finalCredit5 = credit + 20;
542 } else {
543 finalCredit5 = credit;
544 }
545 L5blcredit.setText(Integer.toString(finalCredit5));

546 if (finalCredit5 == 120 && avg0 < 40) {
547 L5blmessage.setText("You are eligible for move to the Level-06");
548 L5blmessage1.setText("");
549 } else if (finalCredit5 == 120) {
550 L5blmessage.setText("You are eligible for move to the Level-06");
551 L5blmessage1.setText("but you have to RETAKE your failure Module");
552 } else if (finalCredit5 == 100 && avg0 < 40) {
553 L5blmessage.setText("You are eligible for move to the Level-06");
554 L5blmessage1.setText("but you have to RETAKE your failure Module");
555 } else if (finalCredit5 < 100) {
556 L5blmessage.setText("You are not eligible for move to the Level-06");
557 L5blmessage1.setText("");
558 } else if (finalCredit5 == 140) {
559 L5blmessage.setText("You are eligible for move to the Level-06");
560 L5blmessage1.setText("and you passed the RETAKE module of Level-04");
561 }
562 }else{
563 lblPRE.setText("Invalid");
564 }
565 }
566 }

```

❖ Exit with DipHE

- 1) Read Level05 credits.
- 2) IF(credits=120 OR credits=140)
 Display CerHE details.
ENDIF

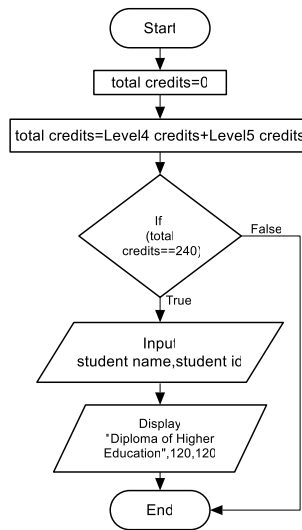


Figure 1.6

```

568 // diphe
569 @FXML
570 private Label lblDipTopic, lblDiphe1, lblDiphe2, lblDiphe3, lblDiphe4, lblDiphe5, lblDiphe6, lblNamediphe, lblDiphe;
571 public void diphe(ActionEvent event) throws Exception {
572     if (L5lblmessage1.getText().equals("You are eligible for move to the Level-06")
573         && L5lblmessage1.getText().equals(""))
574         || L5lblmessage1.getText().equals("and you passed the RETAKE module of Level-04")) {
575         Stage primaryStage = new Stage();
576         Parent root = FXMLLoader.load(getClass().getResource("/application/diphe.fxml"));
577         Scene scene = new Scene(root);
578         scene.getStylesheets().add(getClass().getResource("application.css").toExternalForm());
579         primaryStage.setScene(scene);
580         primaryStage.show();
581     }
582 }
583
584 public void Proceed2(ActionEvent event) throws Exception {
585     lblDipTopic.setText("Diploma of Higher Education");
586     lblDiphe1.setText("120");
587     lblDiphe2.setText("120");
588     lblDiphe3.setText("--");
589     lblDiphe4.setText("No Degree");
590     lblDiphe5.setText("--");
591     lblDiphe6.setText("YOU HAVE THE DIPLOMA OF HIGHER EDUCATION");
592     lblNamediphe.setText(name);
593     lblDiphe.setText(Integer.toString(id));
594 }
  
```

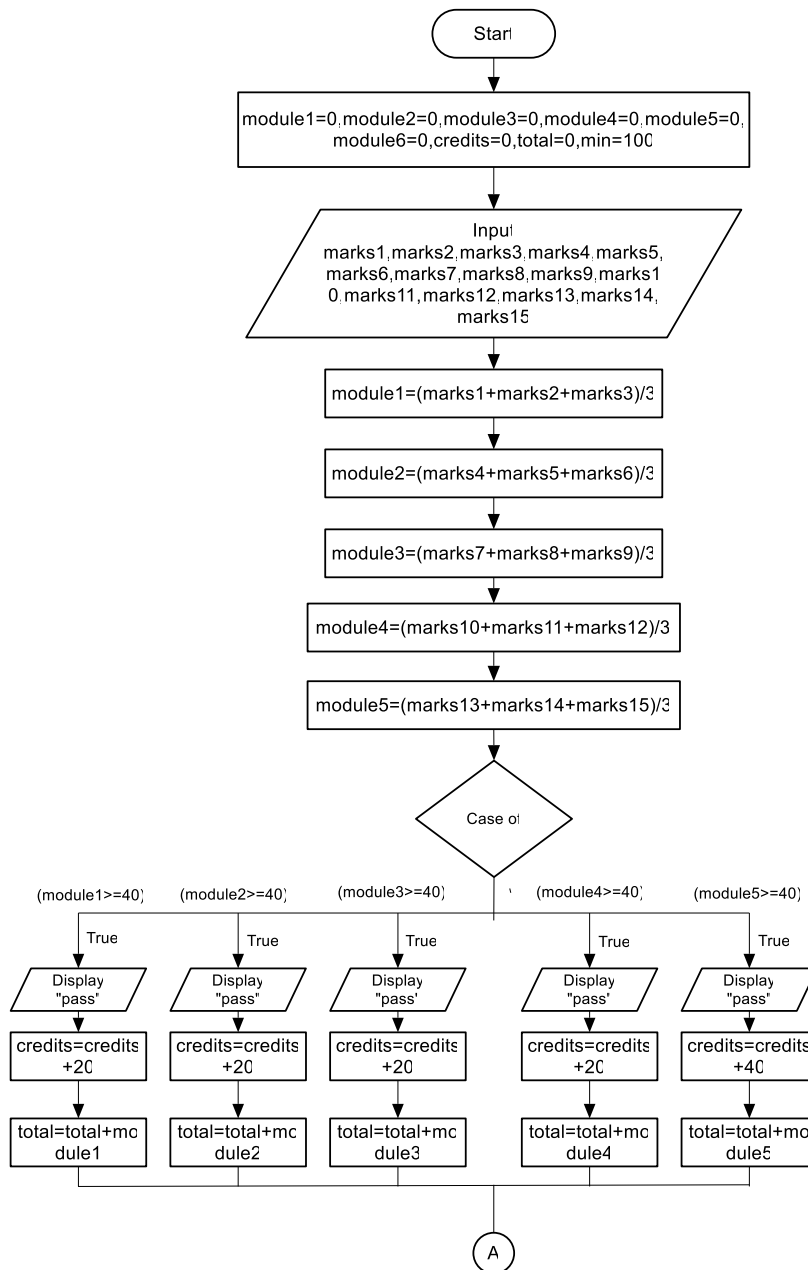
❖ Calculate Level06 credits and marks.

- 1) Prompt user for marks of each ICT exams.
- 2) Calculate Total marks of each module separately.
- 3) Calculate Average marks of each module separately.
- 4) IF(Average>=40)
 Display "pass"
ELSEIF(Average>=30
 Display "resit"
ELSE

Display "retake"

ENDIF

- 5) If have a failure module of Level04 Prompt user for retake marks.
- 6) Calculate retake module credits.
- 7) Calculate credits = (Number of pass single credits modules*20+number of double credits module*40+retake credits).
- 8) If get enough credits move to the Award calculation.



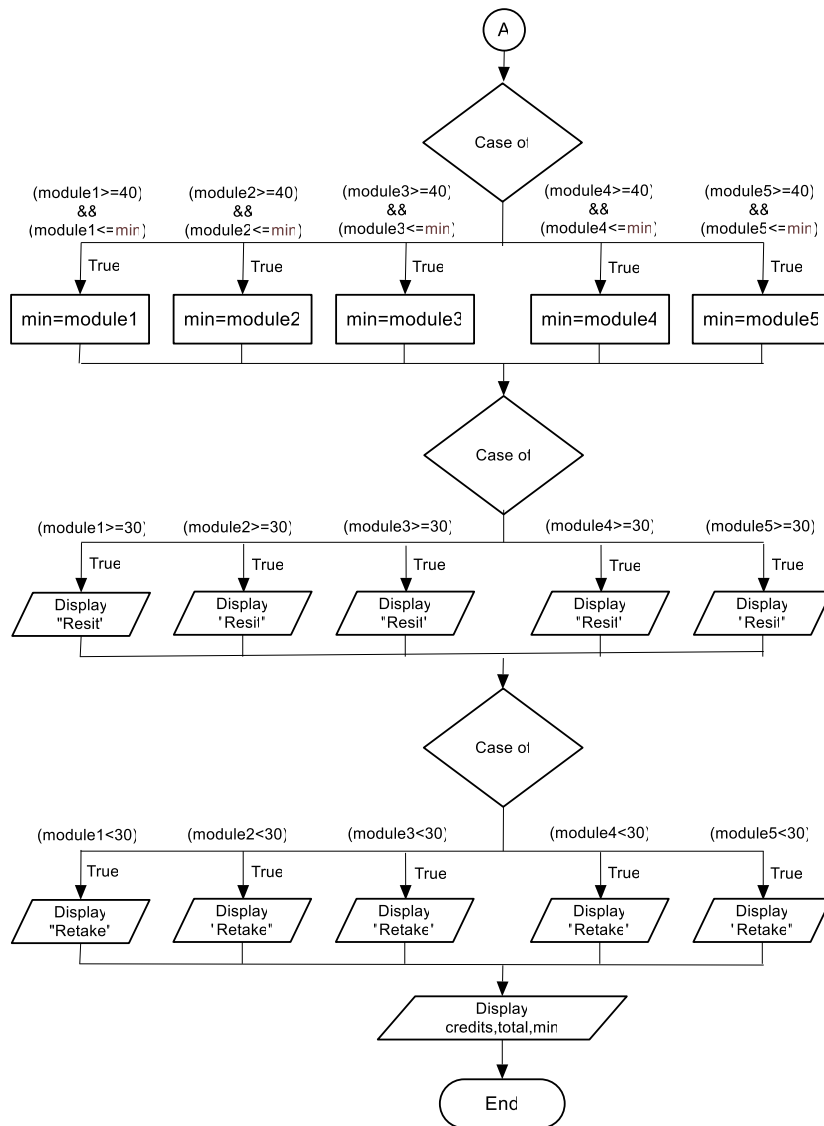


Figure 1.7

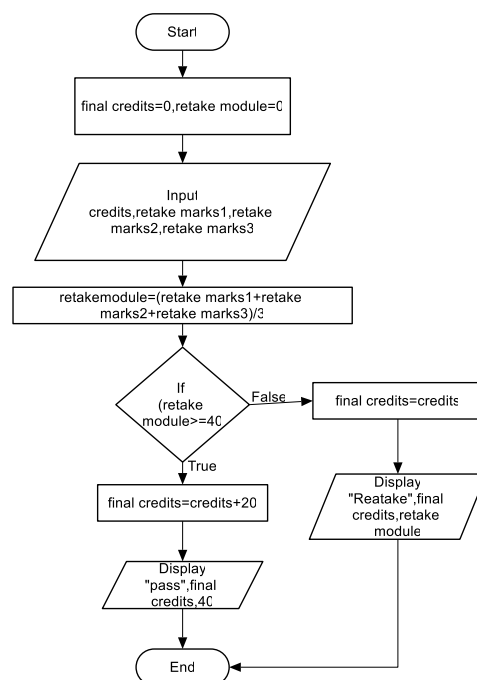


Figure 1.8

```

588 // Level 06 Modules Window
589 @FXML
590 private TextField L6M1Ict01, L6M1Ict02, L6M1Ict03, L6M2Ict01, L6M2Ict02, L6M2Ict03, L6M3Ict01, L6M3Ict02, L6M3Ict03,
591 L6M4Ict01, L6M4Ict02, L6M4Ict03, L6M5Ict01, L6M5Ict02, L6M5Ict03, L6M6Ict01, L6M6Ict02, L6M6Ict03,
592 L6Ict01Re, L6Ict02Re, L6Ict03Re;
593
594 @FXML
595 private Label L6lblM1, L6lblM2, L6lblM3, L6lblM4, L6lblM5, L6lblM6, L6lblRe, L6lblP1, L6lblP2, L6lblP3, L6lblP4,
596 L6lblP5, L6lblP6, L6lblPRe, L6lblcredit, L6lblmessage, L6lblmessage1, L6lblce, L6lblmin, L6lbltotal;
597
598 public void getinputs2(ActionEvent event) {
599     // Module-01
600     int marks1 = Integer.parseInt(L6M1Ict01.getText());
601     int marks2 = Integer.parseInt(L6M1Ict02.getText());
602     int marks3 = Integer.parseInt(L6M1Ict03.getText());
603     int total1 = 0, avg1 = 0;
604     if (marks1 <= 100 && marks2 <= 100 && marks3 <= 100) {
605         total1 = (marks1 + marks2 + marks3);
606         avg1 = total1 / 3;
607         L6lblM1.setText(Integer.toString(avg1));
608         if (avg1 >= 40 && (marks1 < 40 || marks2 < 40 || marks3 < 40)) {
609             L6lblP1.setText("RESITE");
610         } else if (avg1 >= 40) {
611             L6lblP1.setText("PASS");
612         } else if (avg1 >= 30) {
613             L6lblP1.setText("RESITE");
614         } else if (avg1 < 30) {
615             L6lblP1.setText("RETAKE");
616         }
617     } else {
618         L6lblP1.setText("Invalid");
619     }
620
621     int[] marks = new int[4];
622     marks[0] = avg1;
623     marks[1] = avg2;
624     marks[2] = avg3;
625     marks[3] = avg4;
626
627     int credit = 0;
628     totalL6 = 0;
629     for (int i = 0; i < marks.length; i++) {
630         if (marks[i] >= 40) {
631             credit = credit + 20;
632             L6lblce.setText(Integer.toString(credit));
633         }
634         if (marks[i] >= 40) {
635             if (marks[i] <= min6) {
636                 min6 = marks[i];
637                 L6lblmin.setText(Integer.toString(min6));
638             }
639
640             totalL6 = totalL6 + marks[i];
641             L6lbltotal.setText(Integer.toString(totalL6));
642         }
643     }
644     if (avg5 >= 40) {
645         credit = credit + 40;
646         L6lblce.setText(Integer.toString(credit));
647     }
648     if (avg5 <= min6 && avg5 >= 40) {
649         min6 = avg5;
650         L6lblmin.setText(Integer.toString(min6));
651     }
652     if (avg5 >= 30) {
653         totalL6 = totalL6 + avg5;
654         L6lbltotal.setText(Integer.toString(totalL6));
655     }
656 }
657
658 public void finish2(ActionEvent event) {
659     int credit = Integer.parseInt(L6lblce.getText());
660     int marks001 = Integer.parseInt(L6Ict01Re.getText());
661     int marks002 = Integer.parseInt(L6Ict02Re.getText());
662     int marks003 = Integer.parseInt(L6Ict03Re.getText());
663     int total00 = 0; // avg00 = 0;
664     if (marks001 <= 100 && marks002 <= 100 && marks003 <= 100) {
665         total00 = (marks001 + marks002 + marks003);
666         avg00 = total00 / 3;
667         L6lblRe.setText(Integer.toString(avg00));
668         if (avg00 >= 40) {
669             L6lblPRe.setText("PASS");
670             L6lblRe.setText(Integer.toString(40));
671         } else if (avg00 >= 30) {
672             L6lblPRe.setText("RESITE");
673         } else if (avg00 < 30) {
674             L6lblPRe.setText("RETAKE");
675         }
676     }
677     if (avg00 >= 40) {
678         finalCredit6 = credit + 20;
679     } else {
680         finalCredit6 = credit;
681     }
682     L6lblcredit.setText(Integer.toString(finalCredit6));
683 }

```

```

787         if (finalCredit6 == 120 && avg00 < 40) {
788             L6lblmessage.setText("You are eligible for the HONORS DEGREE");
789             L6lblmessage1.setText("");
790         } else if (finalCredit6 == 120) {
791             L6lblmessage.setText("You are eligible for the HONORS DEGREE");
792             L6lblmessage1.setText("nd you passed the RETAKE module of Level-05");
793         } else if (finalCredit6 >= 60 && avg00 < 40) {
794             L6lblmessage.setText("You are eligible for the NON-HONORS DEGREE");
795             L6lblmessage1.setText("you can RETAKE your failure modules");
796         } else if (finalCredit6 < 60) {
797             L6lblmessage.setText("You have not passed the Level-06");
798             L6lblmessage1.setText("");
799         } else if (finalCredit6 == 140) {
800             L6lblmessage.setText("You are eligible for the HONORS DEGREE");
801             L6lblmessage1.setText("and you passed the RETAKE module of Level-05 ");
802         }
803     }
804 } else {
805     L6lblPre.setText("Invalid");
806 }
807 }

694 public void next3(ActionEvent event) throws Exception {
695     if (L6lblmessage.getText().equals("You are eligible for the HONOURS DEGREE")
696         || L6lblmessage.getText().equals("You are eligible for the NON-HONOURS DEGREE")) {
697         Stage primaryStage = new Stage();
698         Parent root = FXMLLoader.load(getClass().getResource("/application/Award.fxml"));
699         Scene scene = new Scene(root);
700         scene.getStylesheets().add(getClass().getResource("application.css").toExternalForm());
701         primaryStage.setScene(scene);
702         primaryStage.show();
703     }
704 }

```

❖ Final Award calculation

- 1) Read final credits of each Level ,total, average, retake marks.
- 2) Calculate total credits
- 3) Calculate total marks.
- 4) Reduce the minimum module marks.
- 5) Calculate the average.
- 6) IF(Credits>360)

Display "Invalid"

ELSEIF(Credits=360)

Display "B.Sc.Honors"

IF(average>=70)

Display "1st class Honors"

ELSEIF(average>=60)

Display "2nd class upper division"

ELSEIF(average>=50)

Display "2nd class lower division"

ELSEIF(average>=40)

Display "3rd class Honors"

ELSEIF(Credits>=300 AND total<360)

Display "B.Sc"

ENDIF

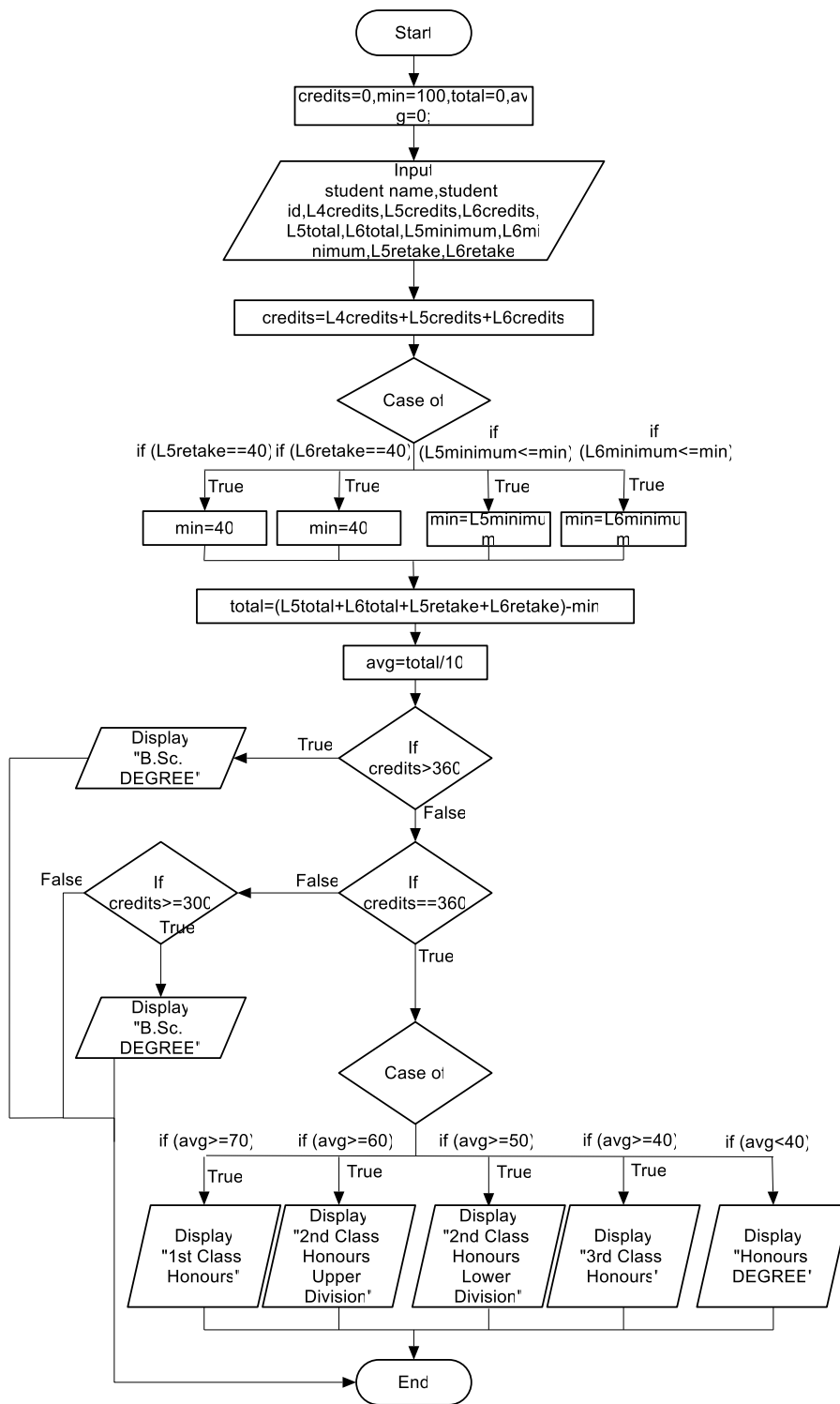


Figure 1.9

```

822 @FXML
823 private Label lblawName, lblawId, lblawcredit, lblawavg, lblawL4c, lblawL5c, lblawL6c, lblawL5t, lblawL6t, lblawL5m, lblawL6m,
824 lblawL5r, lblawL6r, lblawmin, lblaw1, lblaw2, lblaw3;
825
826 public void proceed(ActionEvent event) throws Exception {
827     if (avg0 >= 40) {
828         avgRe1=40;
829     }else{
830         avgRe1=0;
831     }
832     if (avg00 >= 40) {
833         avgRe2=40;
834     }else{
835         avgRe2=0;
836     }
837     lblawName.setText(name);
838     lblawId.setText(Integer.toString(id));
839     lblawL4c.setText(Integer.toString(finalCredit4));
840     lblawL5c.setText(Integer.toString(finalCredit5));
841     lblawL6c.setText(Integer.toString(finalCredit6));
842     lblawL5t.setText(Integer.toString(totalL5));
843     lblawL6t.setText(Integer.toString(totalL6));
844     lblawL5m.setText(Integer.toString(min5));
845     lblawL6m.setText(Integer.toString(min6));
846     lblawL5r.setText(Integer.toString(avgRe1));
847     lblawL6r.setText(Integer.toString(avgRe2));
848
849     int credits = 0, min = 100, total = 0, avg = 0;
850     credits = finalCredit4 + finalCredit5 + finalCredit6;
851     lblawcredit.setText(Integer.toString(credits));
852     int[] marks = new int[2];
853     marks[0] = min5;
854     marks[1] = min6;
855     if (avgRe1 == 40) {
856         min = 40;
857         lblawmin.setText(Integer.toString(min));
858     }
859
860     if (avgRe2 == 40) {
861         min = 40;
862         lblawmin.setText(Integer.toString(min));
863     }
864
865     for (int i = 0; i < marks.length; i++) {
866         if (marks[i] <= min) {
867             min = marks[i];
868             lblawmin.setText(Integer.toString(min));
869         }
870     }
871     total = (totalL5 + totalL6 + avgRe1 + avgRe2) - min;
872     avg = total / 10;
873     lblawavg.setText(Integer.toString(avg));
874     if (credits > 360) {
875         lblaw1.setText("Invalid");
876         lblaw2.setText("Invalid");
877         lblaw3.setText("Invalid");
878     }else if (credits == 360) {
879         lblaw1.setText("B.Sc. Honors");
880         if (avg >= 70) {
881             lblaw2.setText("1st Class Honors");
882             lblaw3.setText("B.Sc. 1st Class Honors DEGREE");
883         } else if (avg >= 60) {
884             lblaw2.setText("2nd Class Honors Upper Division");
885             lblaw3.setText("B.Sc. 2nd Class Honors Upper Division DEGREE");
886         } else if (avg >= 50) {
887             lblaw2.setText("2nd Class Honors Lower Division");
888             lblaw3.setText("B.Sc. 2nd Class Honors Lower Division DEGREE");
889         } else if (avg >= 40) {
890             lblaw2.setText("3rd Class Honors");
891             lblaw3.setText("B.Sc. 3rd Class Honors DEGREE");
892         } else {
893             lblaw2.setText("No Class");
894             lblaw3.setText("B.Sc. Honors DEGREE");
895         }
896     } else if (credits >= 300) {
897         lblaw1.setText("B.Sc.");
898         lblaw3.setText("B.Sc. DEGREE");
899         lblaw2.setText("--");
900     }

```

Test Cases

Black-Box Testing

INPUT	EXPECTED RESULT	ACTUAL RESULT
Enter Marks For Each Module	It will be able to enter only Integers.	It will be able to enter only Integers.
Press Ok	Calculate average of each module and print whether Pass, Resit or Retake and Print current credits.	Calculate average of each module and print whether Pass, Resit or Retake and Print current credits.

Enter Condoned	If module has condoned enter condoned credits. Else enter 0.	If module has condoned enter condoned credits. Else enter 0.
Press Finish	Calculate final credits, adding condoned and display states.	Calculate final credits, adding condoned and display states.
Press CerHE/DipHE	Exit for the CerHE/DipHE	Exit for the CerHEDipHE.
Press Next Level	Move to the next level	Move to the next level
Press Ok Button in the Level05 and Level06	Calculate average of each module and print whether Pass, Resit or Retake and Print current credits. Calculate total of averages and minimum average,	Calculate average of each module and print whether Pass, Resit or Retake and Print current credits. Calculate total of averages and minimum average,
Press Finish Button in the Level05 and Level06.	Calculate final credits, adding Retake module credits and display states.	Calculate final credits, adding Retake module credits and display states.
Press Proceed	Calculate final Award	-.

Table 1.0

INPUT	EXPECTED RESULT	ACTUAL RESULT
45 855 799	invalid	invalid
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 80 80 20 10 10 10 40 40 40 39 39 39 100 100 100	L4-Credits=60, Con=40,Finla credits=100 L5-Total = 269, Minimum = 40, Credits=80, Final credits=100 L6- Total = 208, Minimum = 40, Credits=60, Final credits=80	L4-Credits=60, Con=40,Finla credits=100 L5-Total = 269, Minimum = 40, Credits=80, Final credits=100 L6- Total = 208, Minimum = 40, Credits=60, Final credits=80
Credits=L4-100 L5-120 L6-140 Total=L5-346 L6-296 Minimum=L5-40 L6-47 Retake=L5-40 L6-40	Credits=360, Average=68, Minimum=40,Degree=B.Sc.Honors, Degree Class=2 nd class Honors upper division, Display "B.Sc 2 nd class Honors upper division DEGREE.	Credits=360, Average=68, Minimum=40,Degree=B.Sc.Honors, Degree Class=2 nd class Honors upper division, Display "B.Sc 2 nd class Honors upper division DEGREE.

Table 1.1

University of Gugsí

Level-4 Modules

Module	Mark 1	Mark 2	Mark 3	Overall Marks	Status
Module-01	45	855	799	%	Invalid
Module-02	0	0	0	0 %	RETAKE
Module-03	0	0	0	0 %	RETAKE
Module-04	0	0	0	0 %	RETAKE
Module-05	0	0	0	0 %	RETAKE
Module-06	0	0	0	0 %	RETAKE

Credits:-
Condoned Credits:-
Final Credits:-

Buttons: Ok, Finish, CerHE, Next Level

Figure 2.0

University of Gugsí

Level-4 Modules

Module	Mark 1	Mark 2	Mark 3	Overall Marks	Status
Module-01	45	85	79	69 %	PASS
Module-02	80	80	20	60 %	PASS
Module-03	10	10	10	10 %	RETAKE
Module-04	40	40	40	40 %	PASS
Module-05	39	39	39	39 %	RESITE
Module-06	100	100	100	100 %	PASS

Credits:- 60
Condoned Credits:- 40
Final Credits:- 100

You are eligible for move to the Level-05
but you have to RETAKE your failure Module

Buttons: Ok, Finish, CerHE, Next Level

Figure 2.1

University of Gugsí

Level-5 Modules

Module	Mark 1	Mark 2	Mark 3	Overall Marks	Status
Module-01	45	85	79	69 %	PASS
Module-02	80	80	20	60 %	RESITE
Module-03	10	10	10	10 %	RETAKE
Module-04	40	40	40	40 %	PASS
Module-05	39	39	39	39 %	RESITE
Module-06	100	100	100	100 %	PASS

Minimum:- 40 Total:- 269
Credits:- 80
Retake module of Level-4: 85 99 78 40 % PASS
Final Credits:- 100

Buttons: Ok, Finish, DipHE, Next Level

Figure 2.2

University of Gugsí

Level-6 Modules

Module	Mark 1	Mark 2	Mark 3	Overall Marks	Status
Module-01	45	85	79	69 %	PASS
Module-02	80	80	20	60 %	RESITE
Module-03	10	10	10	10 %	RETAKE
Module-04	40	40	40	40 %	PASS
Module-05	39	39	39	39 %	RESITE

Minimum:- 40 Total:- 208
Credits:- 60
Retake module of Level-5: 85 99 78 40 % PASS
Final Credits:- 80

Buttons: Ok, Finish, Next

Figure 2.3

University of Gugsí

Final Awards

NAME:- chamod ID:- 2015300

	CREDITS	TOTAL	MINIMUM	RETAKE TOTAL
LEVEL-04:-	100			
LEVEL-05:-	120	346	40	40
LEVEL-06:-	140	296	47	40

CREDITS:- 360 AVERAGE:- 68 MINIMUM:- 40

DEGREE:- B.Sc. Honors
DEGREE CLASS:- 2nd Class Honors Upper Division

B.Sc. 2nd Class Honors Upper Division DEGREE

Buttons: Proceed

Figure 2.5

White Box Testing

CONDITION	YES	NO
If user name=="chamod"&& password=="2015300"	Login to the program.	"Login failed "error shown .
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 module average>=30	Credits= Credits+20	Move to the next condition.
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 "1 st class honors degree"	Move to the next condition.
If total average marks>=60	Display "2 nd class honors upper division degree"	Move to the next condition.
If total average marks>=50	Display "2 nd class honors lower division degree"	Move to the next condition.
If total average marks>=40	Display "3 rd class honors degree"	Display "No class "

Table 1.2

Conclusion

I learnt what is the javaFX and I knew about how to design a GUI based program. I got some knowledge about how to do a course work properly.