CZ2002

Object Oriented Programming and Design

Lab Assignment Report

Student Course Registration and Mark Entry Application (SCRAME)
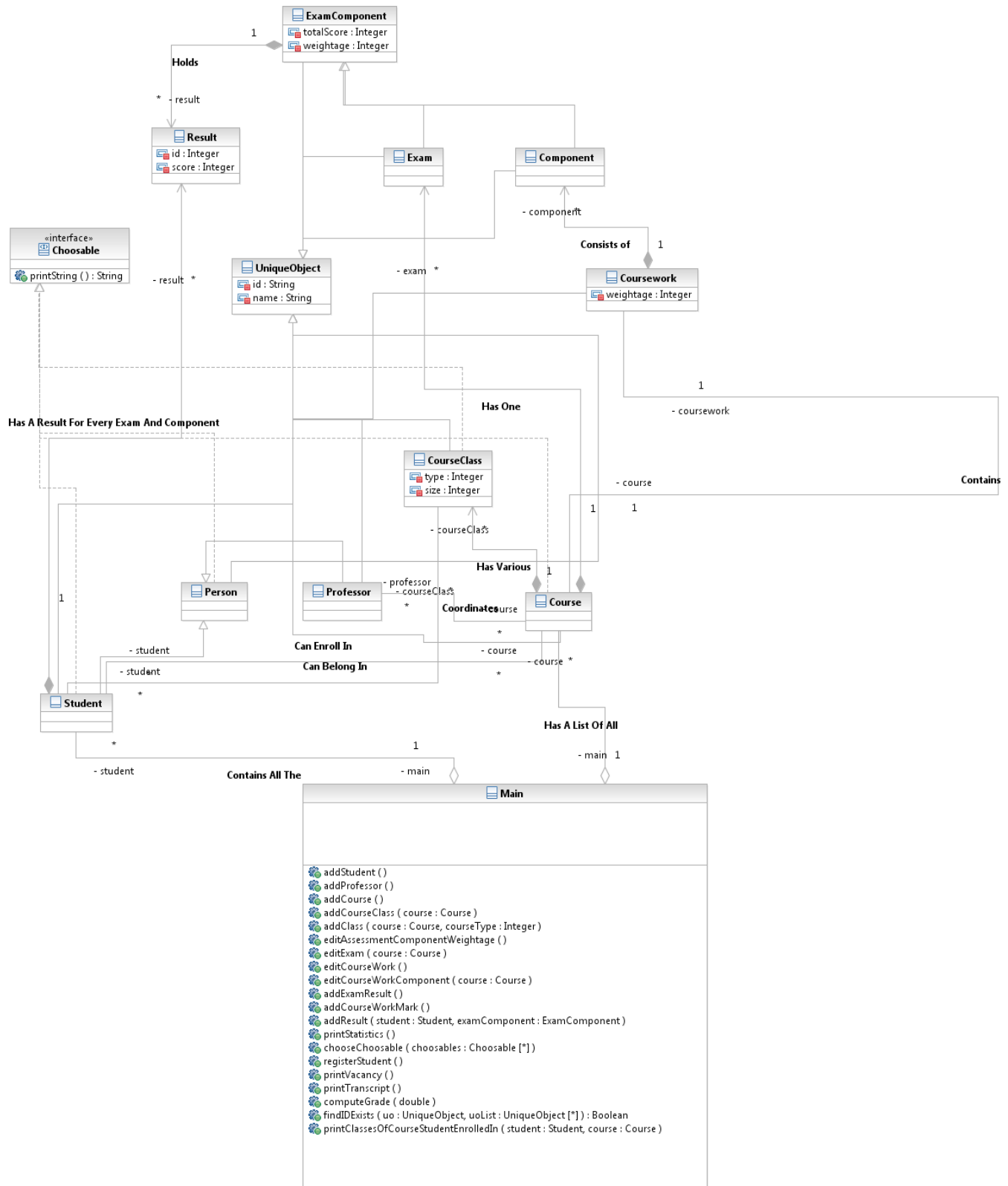
Name:	Matriculation No.:
Cheok Jia De	U1121561B
June Quak Ren Feng	U1121476E
Sri Hartati	U1123011H

Submission Date: 12 November 2012

Table of Contents

1.   Class Diagram

ExamComponent
- totalScore : Integer
- weightage : Integer

1

Holds

* - result

Result
- id : Integer
- score : Integer

Exam

Component

- component

Consists of   1

«interface»
Choosable
- printString ( ) : String

- result   *

UniqueObject
- id : String
- name : String

- exam   *

Coursework
- weightage : Integer

1

- coursework

Has One

Has A Result For Every Exam And Component

CourseClass
- type : Integer
- size : Integer

- course

Contains

1   1

- courseClass

Has Various   1

1

Person

Professor

- professor
- courseClass

*

- course

Coordinates course

Course

Can Enroll In

- student

- student

Can Belong In

- course   *

- course   *

*

1

Student

*

Has A List Of All

*   1

- student   **Contains All The**   - main

- main   1

Main

- addStudent ( )
- addProfessor ( )
- addCourse ( )
- addCourseClass (course : Course)
- addClass (course : Course, courseType : Integer)
- editAssessmentComponentWeightage ( )
- editExam (course : Course)
- editCourseWork ( )
- editCourseWorkComponent (course : Course)
- addExamResult ( )
- addCourseWorkMark ( )
- addResult (student : Student, examComponent : ExamComponent)
- printStatistics ( )
- chooseChoosable (choosables : Choosable [*])
- registerStudent ( )
- printVacancy ( )
- printTranscript ( )
- computeGrade (double)
- findIDExists (uo : UniqueObject, uoList : UniqueObject [*]) : Boolean
- printClassesOfCourseStudentEnrolledIn (student : Student, course : Course)

1

2.   Sequence Diagram for 'Print Student Transcript'
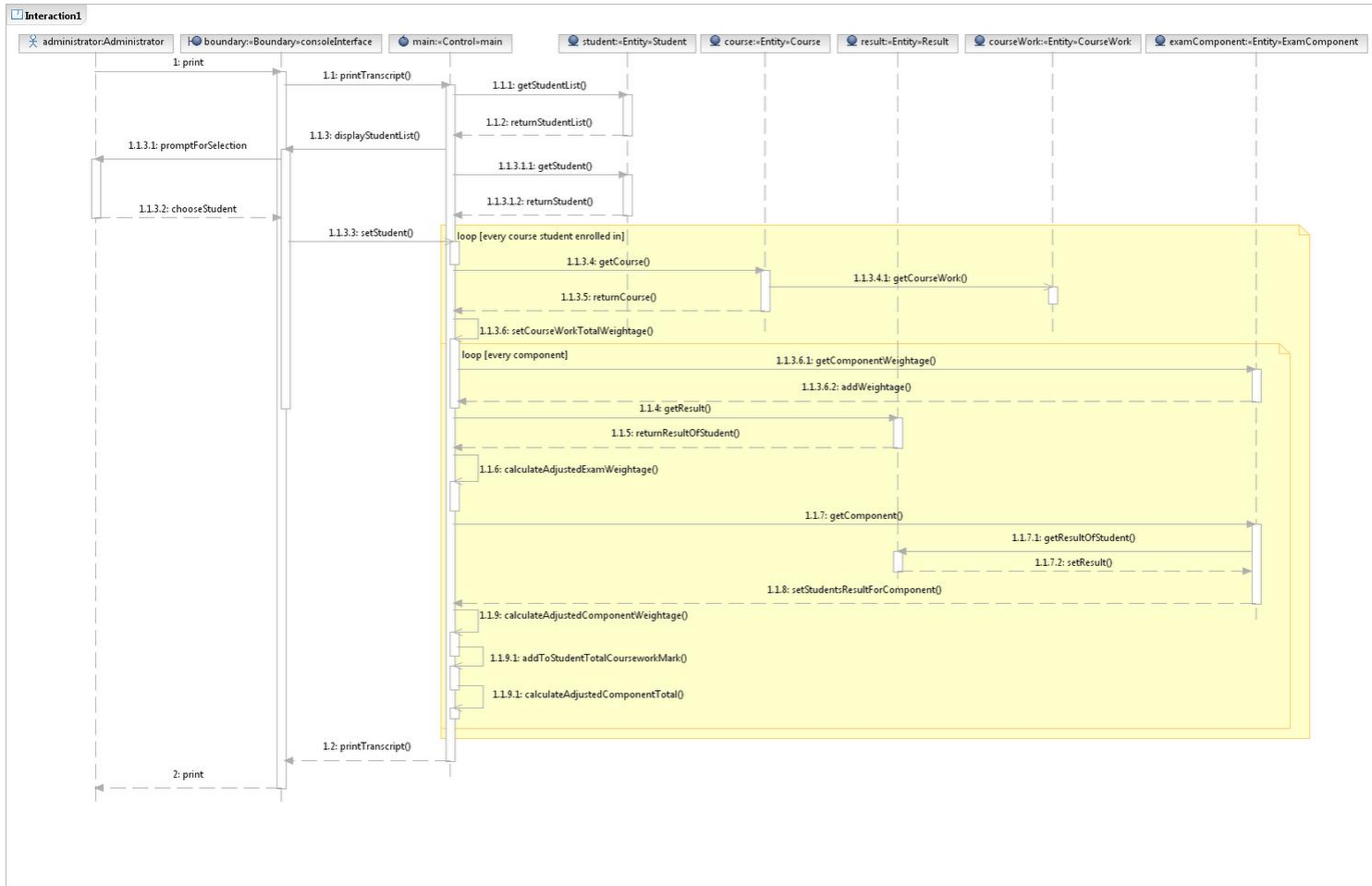
3.   Design considerations & use of OO concepts

As the focus of this course is object oriented programming, we have decided to use an object database(db4o) for persistence of our objects as this would preserve the uniqueness of our objects without needing a unique key attribute for each object. Using an object database has also allowed us to focus on the logic of the program as opposed to wasting time on data management. Also this would allow us to not be accused of plagiarism as any code using a file reader would likely trigger turnitin's plagiarism algorithms.

Due to this course also focusing on OOP and not on data entry, we have decided to minimized the amount of data needed to create an object and instead just have a name an ID. Requiring less attributes in an object will also allow us to demonstrate the program in a timely fashion and still showing the capabilities. Furthermore, adding additional attributes would be easy to implement in the future if required as there is full separation between entity and business logic.

Although the recommended data structure for an aggregation is a set due to having only unique elements, we have used a list as the set interface does not provide a method for getting an element by position due to the nature of a set. Thus, we have included constraints to prevent duplicate objects.

For our course classes we chose to use a single class to represent any class with a type attribute defining the class type. This allows us to easily add or remove class types as required instead of using Java reflection or creating 3 sub classes that have the only difference in the value of a type attribute and class name.

For validation and invalid input, we have tried hard to disallow invalid input as opposed to handling invalid input. For example, when choosing a single unique object, the user can select the object from an easy-to-use list or search for the object by predefined attributes.

In the first case, a single method lets the user select a single object from any array of objects that implement a <choosable> interface. If the user decides not to select any object, the method returns null and the user is taken back to the main menu.

For validity of input like limiting names to [aA-zZ], we have decided that as NTU is a globalized university, we will not restrict anything to an Anglicized preconceived notion of naming but instead allow any input supported by Unicode other than a carriage return. The only exception to this is NTU's internal identification like course code and student ID.

For exam and coursework components, due to them having mostly similar attributes and functions, we have extended them from an abstract <ExamComponent> class. The only difference in the 2 classes is that a single exam is owned by a course while many coursework components are owned by the coursework.

Furthermore, for exam and coursework, we have chosen to use a weighted percentage system to compute the final percentage as opposed to using a hard percentage of 100. We feel this be more intuitive for users as in test cases suggested, if a percentage of 50% was entered for the coursework, the user had to make sure the coursework components added up to 50%. Whereas in a weighted percentage, the user just needs to make sure they are in the correct ratios. Our program will take care of making sure they add up correctly in terms of the weights given.

For our result class, it has a one-to-many relationship with both the the student and ExamComponent. Therefore, the uniqueness of a result object is defined its student and ExamComponent objects (or foreign keys if in a relational database). Our system validates and maintains this uniqueness at data creation time.

For the use of OO concepts, we have tried to the SOLID principles where possible. All our objects have a Single responsibility – that of storing data and connections to other objects. This leads to us to the Open/closed principle where the entities can be extended but modifying their attributes would lead to the program crashing. For the Liskov substitution principle, our child objects can easily replace their parent objects. In following the Interface segregation principle, we created an interface that specific objects use instead of using general interfaces already implemented.

4.   Test Case

1.1 Adding a new student	1.2 Adding an existing student
Expected outcome: Program should allow adding of student and print out list of students after added successfully.	Expected outcome: Program should prompt error message that the student with matriculation number already existed.
| ```
Choice made:1
Add a Student : Input Student's Name and Matriculation No. - Edwin Sim
U1125439R
New Student : U1125439R - Edwin Sim has been added successfully!

List of Students Registered:
U1121476E - June Quak Ren Feng
U1123011H - Sri Hartati
U1121561B - Cheok Jia De
U1125678J - Peh Weileng
U1129907Y - Xu Ai
U1120922T - Lim Guan
U1128823U - Tiffany Chen
U1122345E - Gn Cher Teck
U1120034G - Choo Zhi Cheng
U1126653W - Teo Soon Kiang
U1127633J - Chung Qiwen
U1127744T - James Liao
U1125987H - Xiao Ming
U1126432Y - Alex Tan
U1127364M - Simon Says
U1125439R - Edwin Sim
``` | ```
*******************************************************************
* STUDENT COURSE REGISTRATION AND MARK ENTRY Application (SCRAME) *
*******************************************************************
                     Please select a choice:
(1) Add Student
(2) Add Course
(3) Register Student for Course
(4) Check Vacancy in a Class
(5) Print Student List by Class Type
(6) Enter Course Assessment Component Weightage
(7) Enter Coursework Marks inclusive of its Components
(8) Enter Exam Marks
(9) Print Course Statistic
(10) Print Transcript

Choice made:1
Add a Student : Input Student's Name and Matriculation No. - Amy Tan
U1123011H
Student not added! Matriculation No. already exists!
``` |
| 2.1 Adding a new course | 2.2 Adding to an existing course |
| Expected outcome: Program should allow adding of new course together with the professor in-charge as well as the number of classes for each class type. | Expected outcome: Program should prompt error message that the course index already existed thus the course is not added. |
| ```
Choice made:2
Add a Course : Input Course Name and Course Index - Software Engineering
CZ2006
Please select Professor-in-Charge for this course:
Select a professor or cancel to add a new professor:
(1) ID: 10001 Name: Tan Kheng Leong
(2) ID: 10003 Name: Cai Wentong
(3) ID: 10005 Name: Huang Shell Ying
(4) ID: 10007 Name: Hsu Wen Jing
(5) ID: 10009 Name: Zhang Wei
End of Professor List. Enter 0 to Restart the List or -1 to Quit
-1
Add a Professor : Input Professor's Name and Staff No. - Chang Kuiyu
10011
Professor Chang Kuiyu, Staff No. 10011 has been added successfully!

Please select Professor-in-Charge for this course:
Select a professor or cancel to add a new professor:
(1) ID: 10001 Name: Tan Kheng Leong
(2) ID: 10003 Name: Cai Wentong
(3) ID: 10005 Name: Huang Shell Ying
(4) ID: 10007 Name: Hsu Wen Jing
(5) ID: 10009 Name: Zhang Wei
(6) ID: 10011 Name: Chang Kuiyu
End of Professor List. Enter 0 to Restart the List or -1 to Quit
6
New Course : CZ2006 - Software Engineering     Professor-in-Charge : Chang Kuiyu has been added successfully!

Add Course Class Types :
(1) Lectures, Laboratory, Tutorials
(2) Lectures & Tutorials ONLY
(3) Lectures ONLY

Choice made :
3
Adding lectures...
How many classes for Software Engineering - lecture ?
1
Add a Course Class : Input Class Name, Class Index, Class Intake Size.
LEC 20055 30
New Course Class : 1 lecture has been added successfully!

List of Courses Added:
CZ2006 Software Engineering
20055 - LEC
``` | ```
*******************************************************************
* STUDENT COURSE REGISTRATION AND MARK ENTRY Application (SCRAME) *
*******************************************************************
                     Please select a choice:
(1) Add Student
(2) Add Course
(3) Register Student for Course
(4) Check Vacancy in a Class
(5) Print Student List by Class Type
(6) Enter Course Assessment Component Weightage
(7) Enter Coursework Marks inclusive of its Components
(8) Enter Exam Marks
(9) Print Course Statistic
(10) Print Transcript

Choice made:2
Add a Course : Input Course Name and Course Index - Astronomy
CZ2002
Course not added! Course Index already exists!
``` |

3.1 Add a student to course with vacancies in Tut / Lab	3.2 Add a student to course with no vacancies in Tut / Lab
Expected outcome: Program should allow student to be enrolled in the courses together with its corresponding tutorial and laboratory classes.	Expected outcome: Program should prompt that the particular class selected already full and prompt to select another class that has vacancy available.

Column 3.1:

```
Choice made:3
Select Student to Register:
(1) ID: U1121476E Name: June Quak Ren Feng
(2) ID: U1123011H Name: Sri Hartati
(3) ID: U1121561B Name: Cheok Jia De
(4) ID: U1125678J Name: Peh Weileng
(5) ID: U1129907Y Name: Xu Ai
(6) ID: U1120922T Name: Lim Guan
(7) ID: U1128823U Name: Tiffany Chen
(8) ID: U1122345E Name: Gn Cher Teck
(9) ID: U1120034G Name: Choo Zhi Cheng
(10) ID: U1126653W Name: Teo Soon Kiang
Enter 11 to see the Next 10 Student
11
(1) ID: U1127633J Name: Chung Qiwen
(2) ID: U1127744T Name: James Liao
(3) ID: U1125987H Name: Xiao Ming
(4) ID: U1126432Y Name: Alex Tan
(5) ID: U1127364M Name: Simon Says
(6) ID: U1125439R Name: Edwin Sim
End of Student List. Enter 0 to Restart the List or -1 to Quit
3
(2) ID: U1123011H Name: Sri Hartati
(3) ID: U1121561B Name: Cheok Jia De
(4) ID: U1125678J Name: Peh Weileng
(5) ID: U1129907Y Name: Xu Ai
(6) ID: U1120922T Name: Lim Guan
(7) ID: U1128823U Name: Tiffany Chen
(8) ID: U1122345E Name: Gn Cher Teck
(9) ID: U1120034G Name: Choo Zhi Cheng
(10) ID: U1126653W Name: Teo Soon Kiang
Enter 11 to see the Next 10 Student
11
(1) ID: U1127633J Name: Chung Qiwen
(2) ID: U1127744T Name: James Liao
(3) ID: U1125987H Name: Xiao Ming
(4) ID: U1126432Y Name: Alex Tan
(5) ID: U1127364M Name: Simon Says
(6) ID: U1125439R Name: Edwin Sim
End of Student List. Enter 0 to Restart the List or -1 to Quit
3
Select Course to Register Student into:
(1) ID: CZ2002 Name: Object-Oriented Programming and Design
(2) ID: CZ2005 Name: Operating System
(3) ID: CZ2001 Name: Algorithms
(4) ID: CZ3001 Name: Advanced Computer Architecture
(5) ID: CZ2006 Name: Software Engineering
End of Course List. Enter 0 to Restart the List or -1 to Quit
1

Please Select the Class to Register For :
(1) ID: 20012 Name: LEC Vacancies: 19
End of CourseClass List. Enter 0 to Restart the List or -1 to Quit
1

Please Select the Class to Register For :
(1) ID: 20013 Name: TUT Vacancies: 10
(2) ID: 20014 Name: TUT Vacancies: 0
End of CourseClass List. Enter 0 to Restart the List or -1 to Quit
2
Course CZ2002 - Object-Oriented Programming and Design  - TUT 20014 Has No More Vacancy. Please Select Another Class!

(1) ID: 20013 Name: TUT Vacancies: 10
(2) ID: 20014 Name: TUT Vacancies: 0
End of CourseClass List. Enter 0 to Restart the List or -1 to Quit
1

Please Select the Class to Register For :
(1) ID: 20015 Name: LAB Vacancies: 4
(2) ID: 20016 Name: LAB Vacancies: 5
End of CourseClass List. Enter 0 to Restart the List or -1 to Quit
2
Student Xiao Ming, Matriculation No. U1125987H Has Successfully Been Enrolled to the Following Course:
CZ2002 Object-Oriented Programming and Design  - Lecture LEC - 20012
CZ2002 Object-Oriented Programming and Design  - Tutorial TUT - 20013
CZ2002 Object-Oriented Programming and Design  - Laboratory LAB - 20016
```

Column 3.2:

```
Choice made:3
Select Student to Register:
(1) ID: U1121476E Name: June Quak Ren Feng
(2) ID: U1123011H Name: Sri Hartati
(3) ID: U1121561B Name: Cheok Jia De
(4) ID: U1125678J Name: Peh Weileng
(5) ID: U1129907Y Name: Xu Ai
(6) ID: U1120922T Name: Lim Guan
(7) ID: U1128823U Name: Tiffany Chen
(8) ID: U1122345E Name: Gn Cher Teck
(9) ID: U1120034G Name: Choo Zhi Cheng
(10) ID: U1126653W Name: Teo Soon Kiang
Enter 11 to see the Next 10 Student
11
(1) ID: U1127633J Name: Chung Qiwen
(2) ID: U1127744T Name: James Liao
(3) ID: U1125987H Name: Xiao Ming
(4) ID: U1126432Y Name: Alex Tan
(5) ID: U1127364M Name: Simon Says
(6) ID: U1125439R Name: Edwin Sim
End of Student List. Enter 0 to Restart the List or -1 to Quit
2
Select Course to Register Student into:
(1) ID: CZ2002 Name: Object-Oriented Programming and Design
(2) ID: CZ2005 Name: Operating System
(3) ID: CZ2001 Name: Algorithms
(4) ID: CZ3001 Name: Advanced Computer Architecture
(5) ID: CZ2006 Name: Software Engineering
End of Course List. Enter 0 to Restart the List or -1 to Quit
1

Please Select the Class to Register For :
(1) ID: 20012 Name: LEC Vacancies: 19
End of CourseClass List. Enter 0 to Restart the List or -1 to Quit
1

Please Select the Class to Register For :
(1) ID: 20013 Name: TUT Vacancies: 10
(2) ID: 20014 Name: TUT Vacancies: 0
End of CourseClass List. Enter 0 to Restart the List or -1 to Quit
2
Course CZ2002 - Object-Oriented Programming and Design  - TUT 20014 Has No More Vacancy. Please Select Another Class!

(1) ID: 20013 Name: TUT Vacancies: 10
(2) ID: 20014 Name: TUT Vacancies: 0
End of CourseClass List. Enter 0 to Restart the List or -1 to Quit
1

Please Select the Class to Register For :
(1) ID: 20015 Name: LAB Vacancies: 4
(2) ID: 20016 Name: LAB Vacancies: 5
End of CourseClass List. Enter 0 to Restart the List or -1 to Quit
2
Student James Liao, Matriculation No. U1127744T Has Successfully Been Enrolled to the Following Course:
CZ2002 Object-Oriented Programming and Design  - Lecture LEC - 20012
CZ2002 Object-Oriented Programming and Design  - Tutorial TUT - 20013
CZ2002 Object-Oriented Programming and Design  - Laboratory LAB - 20016
```

4.1 Check vacancies for classes	5.1 Print Student List by Class Type (Lecture)
Expected outcome: Program should display the vacancies available with the total maximum vacancies allowed in the particular class selected.	Expected outcome: Program should display the student list with name and matric no. who has enrolled in the particular lecture class selected.

Left column (4.1):

```
****************************************************************
* STUDENT COURSE REGISTRATION AND MARK ENTRY Application (SCRAME) *
****************************************************************
                  Please select a choice:
(1) Add Student
(2) Add Course
(3) Register Student for Course
(4) Check Vacancy in a Class
(5) Print Student List by Class Type
(6) Enter Course Assessment Component Weightage
(7) Enter Coursework Marks inclusive of its Components
(8) Enter Exam Marks
(9) Print Course Statistic
(10) Print Transcript

Choice made:4
Select Course To View Vacancy :
(1) ID: CZ2002 Name: Object-Oriented Programming and Design
(2) ID: CZ2005 Name: Operating System
(3) ID: CZ2001 Name: Algorithms
(4) ID: CZ3001 Name: Advanced Computer Architecture
(5) ID: CZ2006 Name: Software Engineering
End of Course List. Enter 0 to Restart the List or -1 to Quit
1
Lecture :       Index : 20012   Name : LEC     Vacancy :19/30
Tutorial :      Index : 20013   Name : TUT     Vacancy :10/10
Tutorial :      Index : 20014   Name : TUT     Vacancy :0/10
Laboratory :    Index : 20015   Name : LAB     Vacancy :4/10
Laboratory :    Index : 20016   Name : LAB     Vacancy :5/10
```

Right column (5.1):

```
****************************************************************
* STUDENT COURSE REGISTRATION AND MARK ENTRY Application (SCRAME) *
****************************************************************
                  Please select a choice:
(1) Add Student
(2) Add Course
(3) Register Student for Course
(4) Check Vacancy in a Class
(5) Print Student List by Class Type
(6) Enter Course Assessment Component Weightage
(7) Enter Coursework Marks inclusive of its Components
(8) Enter Exam Marks
(9) Print Course Statistic
(10) Print Transcript

Choice made:5
Please Select the Course to Display:
(1) ID: CZ2002 Name: Object-Oriented Programming and Design
(2) ID: CZ2005 Name: Operating System
(3) ID: CZ2001 Name: Algorithms
(4) ID: CZ3001 Name: Advanced Computer Architecture
(5) ID: CZ2006 Name: Software Engineering
End of Course List. Enter 0 to Restart the List or -1 to Quit
1
Please Select the Course Class to Display:
(1) ID: 20012 Name: LEC Vacancies: 19
(2) ID: 20013 Name: TUT Vacancies: 10
(3) ID: 20014 Name: TUT Vacancies: 0
(4) ID: 20015 Name: LAB Vacancies: 4
(5) ID: 20016 Name: LAB Vacancies: 5
End of CourseClass List. Enter 0 to Restart the List or -1 to Quit
1
Printing Course List of Students in 20012 LEC lecture
Student's Matriculation No. : U1121476E Student's Name : June Quak Ren Feng
Student's Matriculation No. : U1123011H Student's Name : Sri Hartati
Student's Matriculation No. : U1121561B Student's Name : Cheok Jia De
Student's Matriculation No. : U1125678J Student's Name : Peh Weileng
Student's Matriculation No. : U1129907Y Student's Name : Xu Ai
Student's Matriculation No. : U1120922T Student's Name : Lim Guan
Student's Matriculation No. : U1128823U Student's Name : Tiffany Chen
Student's Matriculation No. : U1122345E Student's Name : Gn Cher Teck
Student's Matriculation No. : U1120034G Student's Name : Choo Zhi Cheng
Student's Matriculation No. : U1125439R Student's Name : Edwin Sim
Student's Matriculation No. : U1127364M Student's Name : Simon Says
```

6.1. Enter Course Component Weightage (Exam + 1 main coursework component w/o sub-components)	6.2. Enter Course Component Weightage (Exam + coursework with 2 sub-components)
Expected outcome: Program should allow adding of exam and coursework weightage	Expected outcome: Program should allow adding of exam and coursework weightage together with the coursework components

Left column (6.1):

```
****************************************************************
* STUDENT COURSE REGISTRATION AND MARK ENTRY Application (SCRAME) *
****************************************************************
                  Please select a choice:
(1) Add Student
(2) Add Course
(3) Register Student for Course
(4) Check Vacancy in a Class
(5) Print Student List by Class Type
(6) Enter Course Assessment Component Weightage
(7) Enter Coursework Marks inclusive of its Components
(8) Enter Exam Marks
(9) Print Course Statistic
(10) Print Transcript

Choice made:6
Choose Course to Update Assessment Component Weightage :
(1) ID: CZ2002 Name: Object-Oriented Programming and Design
(2) ID: CZ2005 Name: Operating System
(3) ID: CZ2001 Name: Algorithms
(4) ID: CZ3001 Name: Advanced Computer Architecture
(5) ID: CZ2006 Name: Software Engineering
End of Course List. Enter 0 to Restart the List or -1 to Quit
2
Exam total and weightage has not been set.
Coursework total and weightage has not been set.
Press 1 to Update Exam and 2 to Update Coursework. Press Any Other Number to Quit.
1
Please Input Total Score of the Exam and its Weighted Percentage : 100 60
Total Score of the Exam is 100 and its Weighted Percentage is 60
Press 1 to Update Exam and 2 to Update Coursework. Press Any Other Number to Quit.
2
Please Input Total Weighted Percentage of the Coursework : 60
Weighted Percentage is 60
There are currently 0 Components.
Do you want to 1. Add a new component?
4
Press 1 to Update Exam and 2 to Update Coursework. Press Any Other Number to Quit.
5
```

Right column (6.2):

```
****************************************************************
* STUDENT COURSE REGISTRATION AND MARK ENTRY Application (SCRAME) *
****************************************************************
                  Please select a choice:
(1) Add Student
(2) Add Course
(3) Register Student for Course
(4) Check Vacancy in a Class
(5) Print Student List by Class Type
(6) Enter Course Assessment Component Weightage
(7) Enter Coursework Marks inclusive of its Components
(8) Enter Exam Marks
(9) Print Course Statistic
(10) Print Transcript

Choice made:6
Choose Course to Update Assessment Component Weightage :
(1) ID: CZ2002 Name: Object-Oriented Programming and Design
(2) ID: CZ2005 Name: Operating System
(3) ID: CZ2001 Name: Algorithms
(4) ID: CZ3001 Name: Advanced Computer Architecture
(5) ID: CZ2006 Name: Software Engineering
End of Course List. Enter 0 to Restart the List or -1 to Quit
1
Exam total and weightage has not been set.
Coursework total and weightage has not been set.
Press 1 to Update Exam and 2 to Update Coursework. Press Any Other Number to Quit.
1
Please Input Total Score of the Exam and its Weighted Percentage : 50 60
Total Score of the Exam is 50 and its Weighted Percentage is 60
Press 1 to Update Exam and 2 to Update Coursework. Press Any Other Number to Quit.
2
Please Input Total Weighted Percentage of the Coursework : 50
Weighted Percentage is 50
There are currently 0 Components.
Do you want to 1. Add a new component?
1
Please Input Component's Name, Total Marks and Weighted Percentage. Assignment
50 50
There are currently 1 Components.
Do you want to 1. Add a new component or 2. Update a component?
1
Please Input Component's Name, Total Marks and Weighted Percentage. Quiz
20 50
There are currently 2 Components.
Do you want to 1. Add a new component or 2. Update a component?
3
Press 1 to Update Exam and 2 to Update Coursework. Press Any Other Number to Quit.
3
```

7.1. Enter Coursework Marks inclusive of its components

Expected outcome: Program should allow adding of student's coursework marks together with the marks of the corresponding coursework components

```
*****************************************************************
* STUDENT COURSE REGISTRATION AND MARK ENTRY Application (SCRAME) *
*****************************************************************
                    Please select a choice:
(1) Add Student
(2) Add Course
(3) Register Student for Course
(4) Check Vacancy in a Class
(5) Print Student List by Class Type
(6) Enter Course Assessment Component Weightage
(7) Enter Coursework Marks inclusive of its Components
(8) Enter Exam Marks
(9) Print Course Statistic
(10) Print Transcript

Choice made:7
Select Course to Add Coursework Marks For :
(1) ID: CZ2002 Name: Object-Oriented Programming and Design
(2) ID: CZ2005 Name: Operating System
(3) ID: CZ2001 Name: Algorithms
(4) ID: CZ3001 Name: Advanced Computer Architecture
(5) ID: CZ2006 Name: Software Engineering
End of Course List. Enter 0 to Restart the List or -1 to Quit
1
Select Student to Add Coursework Results For :
(1) ID: U1121476E Name: June Quak Ren Feng
(2) ID: U1123011H Name: Sri Hartati
(3) ID: U1121561B Name: Cheok Jia De
(4) ID: U1125678J Name: Peh Weileng
(5) ID: U1129907Y Name: Xu Ai
(6) ID: U1120922T Name: Lim Guan
(7) ID: U1128823U Name: Tiffany Chen
(8) ID: U1122345E Name: Gn Cher Teck
(9) ID: U1120034G Name: Choo Zhi Cheng
(10) ID: U1125439R Name: Edwin Sim
Enter 11 to see the Next 10 Student
1
Select Component to Add Marks For :
(1) Component name: Assignment  Total Score: 50      Weightage: 50
(2) Component name: Quiz       Total Score: 20      Weightage: 50
End of Component List. Enter 0 to Restart the List or -1 to Quit
1
Maximum Grade for Assignment is 50. Please Input Grade for June Quak Ren Feng : 45
```

7.2. Enter Coursework Marks with 2 components

Expected outcome: Program should allow adding of student's coursework marks together with the marks of the corresponding coursework components

```
*****************************************************************
* STUDENT COURSE REGISTRATION AND MARK ENTRY Application (SCRAME) *
*****************************************************************
                    Please select a choice:
(1) Add Student
(2) Add Course
(3) Register Student for Course
(4) Check Vacancy in a Class
(5) Print Student List by Class Type
(6) Enter Course Assessment Component Weightage
(7) Enter Coursework Marks inclusive of its Components
(8) Enter Exam Marks
(9) Print Course Statistic
(10) Print Transcript

Choice made:7
Select Course to Add Coursework Marks For :
(1) ID: CZ2002 Name: Object-Oriented Programming and Design
(2) ID: CZ2005 Name: Operating System
(3) ID: CZ2001 Name: Algorithms
(4) ID: CZ3001 Name: Advanced Computer Architecture
(5) ID: CZ2006 Name: Software Engineering
End of Course List. Enter 0 to Restart the List or -1 to Quit
1
Select Student to Add Coursework Results For :
(1) ID: U1121476E Name: June Quak Ren Feng
(2) ID: U1123011H Name: Sri Hartati
(3) ID: U1121561B Name: Cheok Jia De
(4) ID: U1125678J Name: Peh Weileng
(5) ID: U1129907Y Name: Xu Ai
(6) ID: U1120922T Name: Lim Guan
(7) ID: U1128823U Name: Tiffany Chen
(8) ID: U1122345E Name: Gn Cher Teck
(9) ID: U1120034G Name: Choo Zhi Cheng
(10) ID: U1125439R Name: Edwin Sim
Enter 11 to see the Next 10 Student
1
Select Component to Add Marks For :
(1) Component name: Assignment  Total Score: 50      Weightage: 50
(2) Component name: Quiz       Total Score: 20      Weightage: 50
End of Component List. Enter 0 to Restart the List or -1 to Quit
2
Maximum Grade for Quiz is 20. Please Input Grade for June Quak Ren Feng : 10
```

8.1. Enter Exam Marks

Expected outcome: Program should allow adding of exam marks for a particular course

```
*****************************************************************
* STUDENT COURSE REGISTRATION AND MARK ENTRY Application (SCRAME) *
*****************************************************************
                    Please select a choice:
(1) Add Student
(2) Add Course
(3) Register Student for Course
(4) Check Vacancy in a Class
(5) Print Student List by Class Type
(6) Enter Course Assessment Component Weightage
(7) Enter Coursework Marks inclusive of its Components
(8) Enter Exam Marks
(9) Print Course Statistic
(10) Print Transcript

Choice made:8
Select Course to Add Exam Results For :
(1) ID: CZ2002 Name: Object-Oriented Programming and Design
(2) ID: CZ2005 Name: Operating System
(3) ID: CZ2001 Name: Algorithms
(4) ID: CZ3001 Name: Advanced Computer Architecture
(5) ID: CZ2006 Name: Software Engineering
End of Course List. Enter 0 to Restart the List or -1 to Quit
1
Select Student to Add Exam Results For :
(1) ID: U1121476E Name: June Quak Ren Feng
(2) ID: U1123011H Name: Sri Hartati
(3) ID: U1121561B Name: Cheok Jia De
(4) ID: U1125678J Name: Peh Weileng
(5) ID: U1129907Y Name: Xu Ai
(6) ID: U1120922T Name: Lim Guan
(7) ID: U1128823U Name: Tiffany Chen
(8) ID: U1122345E Name: Gn Cher Teck
(9) ID: U1120034G Name: Choo Zhi Cheng
(10) ID: U1125439R Name: Edwin Sim
Enter 11 to see the Next 10 Student
1
Maximum Grade for Object-Oriented Programming and Design  is 50. Please Input Grade for June Quak Ren Feng : 40
```

8.2 Enter Exam Marks with a course with no student enrolled in

Expected outcome: Program should prompt that the course selected has no student enrolled in.

```
*****************************************************************
* STUDENT COURSE REGISTRATION AND MARK ENTRY Application (SCRAME) *
*****************************************************************
              Please select a choice:
(1) Add Student
(2) Add Course
(3) Register Student for Course
(4) Check Vacancy in a Class
(5) Print Student List by Class Type
(6) Enter Course Assessment Component Weightage
(7) Enter Coursework Marks inclusive of its Components
(8) Enter Exam Marks
(9) Print Course Statistic
(10) Print Transcript

Choice made:8
Select Course to Add Exam Results For :
(1) ID: CZ2002 Name: Object-Oriented Programming and Design
(2) ID: CZ2005 Name: Operating System
(3) ID: CZ2001 Name: Algorithms
(4) ID: CZ3001 Name: Advanced Computer Architecture
(5) ID: CZ2006 Name: Software Engineering
End of Course List. Enter 0 to Restart the List or -1 to Quit
2
Select Student to Add Exam Results For :
Empty list! Please try again!
```
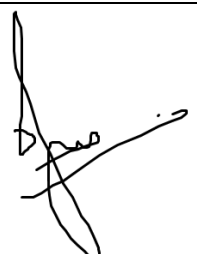
8.3. Enter Exam Marks with a mark that exceeded the total exam mark and will be set to the highest marks	**9.1 Print Course Statistics (with overall grade percentage for exam and coursework and its corresponding exam and coursework only)**
Expected outcome: Program should automatically set the exam marks to the highest marks allowed.	Expected outcome: Program should display the overall grade percentage for exam + coursework for a course together with each of the exams n coursework components

```
************************************************************
* STUDENT COURSE REGISTRATION AND MARK ENTRY Application (SCRAME) *
************************************************************
                Please select a choice:
(1) Add Student
(2) Add Course
(3) Register Student for Course
(4) Check Vacancy in a Class
(5) Print Student List by Class Type
(6) Enter Course Assessment Component Weightage
(7) Enter Coursework Marks inclusive of its Components
(8) Enter Exam Marks
(9) Print Course Statistic
(10) Print Transcript

Choice made:8
Select Course to Add Exam Results For :
(1) ID: CZ2002 Name: Object-Oriented Programming and Design
(2) ID: CZ2005 Name: Operating System
(3) ID: CZ2001 Name: Algorithms
(4) ID: CZ3001 Name: Advanced Computer Architecture
(5) ID: CZ2006 Name: Software Engineering
End of Course List. Enter 0 to Restart the List or -1 to Quit
1
Select Student to Add Exam Results For :
(1) ID: U1121476E Name: June Quak Ren Feng
(2) ID: U1123011H Name: Sri Hartati
(3) ID: U1121561B Name: Cheok Jia De
(4) ID: U1125678J Name: Peh Weileng
(5) ID: U1129907Y Name: Xu Ai
(6) ID: U1120922T Name: Lim Guan
(7) ID: U1128823U Name: Tiffany Chen
(8) ID: U1122345E Name: Gn Cher Teck
(9) ID: U1120034G Name: Choo Zhi Cheng
(10) ID: U1125439R Name: Edwin Sim
Enter 11 to see the Next 10 Student
3
Maximum Grade for Object-Oriented Programming and Design  is 50. Please Input Grade for Cheok Jia De : 60
Mark entered exceeded maximum grade. Mark has been set to maximum.
```

```
************************************************************
* STUDENT COURSE REGISTRATION AND MARK ENTRY Application (SCRAME) *
************************************************************
                Please select a choice:
(1) Add Student
(2) Add Course
(3) Register Student for Course
(4) Check Vacancy in a Class
(5) Print Student List by Class Type
(6) Enter Course Assessment Component Weightage
(7) Enter Coursework Marks inclusive of its Components
(8) Enter Exam Marks
(9) Print Course Statistic
(10) Print Transcript

Choice made:9
Select Course to View Statistics :
(1) ID: CZ2002 Name: Object-Oriented Programming and Design
(2) ID: CZ2005 Name: Operating System
(3) ID: CZ2001 Name: Algorithms
(4) ID: CZ3001 Name: Advanced Computer Architecture
(5) ID: CZ2006 Name: Software Engineering
End of Course List. Enter 0 to Restart the List or -1 to Quit
1
Average Actual Mark for Exam is 40/50. Average Percentage is 80. Average Weighted Percentage is 43 .
Average Actual Mark for Assignment is 33/50. Average Percentage is 66. Average Weighted Percentage is 15
Average Actual Mark for Quiz is 12/20. Average Percentage is 60. Average Weighted Percentage is 13 .
Overall grade percentage for Object-Oriented Programming and Design  is 71
```

10.1. Print transcript (Select Student to print the results)	**10.2. Print transcript that shows results for all courses registered by a particular students (enrolled in 4 courses)**
Expected outcome: Program should allow to select a student to show the student's results.	Expected outcome: Program should display the results for all courses the student has enrolled in, together with the grade of that student for that course.

```
************************************************************
* STUDENT COURSE REGISTRATION AND MARK ENTRY Application (SCRAME) *
************************************************************
                Please select a choice:
(1) Add Student
(2) Add Course
(3) Register Student for Course
(4) Check Vacancy in a Class
(5) Print Student List by Class Type
(6) Enter Course Assessment Component Weightage
(7) Enter Coursework Marks inclusive of its Components
(8) Enter Exam Marks
(9) Print Course Statistic
(10) Print Transcript

Choice made:10
Please Select Student Matriculation No. to Display Transcript :
(1) ID: U1121476E Name: June Quak Ren Feng
(2) ID: U1123011H Name: Sri Hartati
(3) ID: U1121561B Name: Cheok Jia De
(4) ID: U1125678J Name: Peh Weileng
(5) ID: U1129907Y Name: Xu Ai
(6) ID: U1120922T Name: Lim Guan
(7) ID: U1128823U Name: Tiffany Chen
(8) ID: U1122345E Name: Gn Cher Teck
(9) ID: U1120034G Name: Choo Zhi Cheng
(10) ID: U1126653W Name: Teo Soon Kiang
Enter 11 to see the Next 10 Student
11
(1) ID: U1127633J Name: Chung Qiwen
(2) ID: U1127744T Name: James Liao
(3) ID: U1125987H Name: Xiao Ming
(4) ID: U1126432Y Name: Alex Tan
(5) ID: U1127364M Name: Simon Says
(6) ID: U1125439R Name: Edwin Sim
End of Student List. Enter 0 to Restart the List or -1 to Quit
5
```

```
Results for Course: Object-Oriented Programming and Design ...
Exam:            Actual: 30/50   Adjusted:32/54

Coursework Breakdown:
Assignment:          Actual: 10/50   Adjusted:4/22
Quiz:            Actual: 20/20   Adjusted:22/22
Final Mark: 58/100 Final Grade: C
Results for Course: Operating System...
Exam:            Actual: 80/100  Adjusted:40/50

Coursework Breakdown:
Quiz1:            Actual: 80/100  Adjusted:35/44
class participation:          Actual: 20/20   Adjusted:5/5
Final Mark: 80/100 Final Grade: D

Results for Course: Algorithms...
Exam:            Actual: 40/80   Adjusted:33/66

Coursework Breakdown:
Quiz:            Actual: 60/100  Adjusted:20/33
Final Mark: 53/100 Final Grade: C

Results for Course: Advanced Computer Architecture...
Exam:            Actual: 40/50   Adjusted:40/50

Coursework Breakdown:
Assignment:          Actual: 60/100  Adjusted:30/50
Final Mark: 70/100 Final Grade: B
```

5. Declaration of Original Work for CE/CZ2002 Assignment

We hereby declare that the attached group assignment has been researched, undertaken, completed and submitted as a collective effort by the group members listed below.

We have honoured the principles of academic integrity and have upheld Student Code of Academic Conduct in the completion of this work.

We understand that if plagiarism is found in the assignment, then lower marks or no marks will be awarded for the assessed work.

Name:	Course	Lab Group	Signature/Date
Cheok Jia De	CZ2002	SSP6	
June Quak Ren Feng	CZ2002	SSP6	
Sri Hartati	CZ2002	SSP6	