

# The University of the South Pacific

## School of IT, Engineering, Mathematics & Physics

---

### CS112: Data Structures & Algorithms

#### Assignment 2 – Semester II, 2023

---

*Total Marks: 15%*

*Due Date: as shown on Moodle*

This assignment covers mostly the practical aspects of this course. The marking rubrics is heavily based on *Programming CBoK*. Rubrics have been taken from ACS-SCIMS rubrics V1.0. The main focus is on the usage of user-defined linked list and classes. C++ is the only acceptable language and Visual Studio is the only acceptable IDE for this assignment. This assessment covers the following course learning outcomes:

CLO 2 – Develop data structures using object-oriented programming to solve well-defined computing problems.

#### Problem statement

The immigration department is responsible for granting different kinds of visas. Suppose you have been hired by the Australian immigration department to make a software for their visa processes. Visa application details are given below:

visaType: tourist visa, student visa or business visa.

invoiceNo: invoice number of visa application

surname: surname of the principal applicant

firstName: first name of the principal applicant

contact: phone number of principal applicant

status: current status of the visa application (waiting client's response – WCR, in the queue (not processed) – NP, in process – IP, completed – C)

result: the outcome of visa application (success, failure, no decision by default)

#### Requirements

1. Download the linked list program from Moodle. You must use this given program for your assignment. DO NOT CHANGE any given code. [-10% if the given code is changed]
2. Make a class for visa application which **MUST** inherit from *Data* class. [Abs: 2%]
3. Make a List of visa application data given in the sample file. [D&IM: 3%]
4. Write the following functions in the main: [P: 2% each]

a) ***void print:*** Print all applications of the sample data to a file or standard output as indicated through argument.

b) ***void print\_success\_list(?) :*** print only successful applications.

[Hint: You may use casting to get the data:

Base \*b = new Derived; // or pointing to an existing Derived object.

Derived \*d = (Base\*)b;]

c) ***void print\_failure\_list(?) :*** print only unsuccessful applications.

d) ***Void remove\_success\_failure(?) :*** remove all successful and unsuccessful applications from the linked list.

5. Provide an appropriate menu for your client.

[SD: 2%]

CBOK	Unsatisfactory (0%-49%)	Satisfactory (50% - 75%)	Good (76% - 100%)	Marks Allocated	% Marks Attained
Programming	I. Code has compile/run/logic errors. Poorly written code. II. Plagiarism III. Poor indentation, hard to read and follow the code IV. Lots of bugs and/or errors V. Program produces unexpected output VI. Inappropriate use/definition of functions. VII. Inappropriate use of variables and parameters. VIII. No input validation IX. Hard coding of data in the program. X. Program is not well structured.	I. Able to write a complex code for a loosely defined problem IV. Appropriate encapsulation, inheritance or polymorphism for object-oriented programming. V. computer program produces expected output	I. All satisfactory and demonstrate very good programming skills.	8	
System Development/ Acquisition	XII. Many important features do not work as expected.	XII. All the required functionalities work correctly.	XII. All the required functionalities work correctly.	2	
Sub Total & comments					

<b>CBOK</b>	<b>Unsatisfactory (0%-49%)</b>	<b>Satisfactory (50% - 75%)</b>	<b>Good (76% - 100%)</b>	<b>Marks Allocated</b>	<b>% Marks Attained</b>
<b>Data and Information Management</b>	VII. Very limited or undesirable data has been collected  VIII. Data is not stored and managed using inappropriate tools.  IX. Poor or incorrect inference is derived from the collected/given data	VII. Appropriate data has been collected  VIII. data is store and managed using a standard tool  IX. logical inference is derived from the collected/given data	I. All satisfactory and  III. Extensive analysis is done on the collected/given data.  IV. Exceptional Inference is derived from the data.	<b>3</b>	
<b>Sub Total &amp; comments</b>					

<b>CBOK</b>	<b>Unsatisfactory (0%-49%)</b>	<b>Satisfactory (50% - 75%)</b>	<b>Good (76% - 100%)</b>	<b>Marks Allocated</b>	<b>% Marks Attained</b>
<b>Abstraction</b>	V. Incorrect flowchart or UML/ERD class diagrams.  VIII. Poor architecture and design makes future updates hard to incorporate.	IV. Correct flowchart or UML/ERD class diagrams using an appropriate tool  VI. Appropriate choice of the architecture and design	IV. Correct flowchart or UML/ERD class diagrams using an appropriate tool  VI. Appropriate choice of the architecture and design	<b>2</b>	
<b>Sub Total &amp; comments</b>					

### **Submission instructions**

1. Write a README file for detailed notes regarding the functionality of the corresponding code, and a set of instructions on how to run them.
2. It is your responsibility to ensure your software works in the lab PCs and it is ready to run without bugs/errors. NO marks will be given if the program does not execute.
3. **This assignment must be done in a group of 2 members.** Assign a group leader and submit the assignment through the group leader's moodle account. You have to submit just one zipped file of your project. The submission filename should read A1\_Sxxx\_Syyy.zip or A1\_Sxxx\_Syyy.rar where Sxxx and Syyy are student ids of the group members. For example A1\_S11003232\_S01004488.zip. Incorrect/late submission will result in a high penalty.
4. Completely fill Mark Allocation Sheet and submit it with your assignment. Failing to do so may result in a deduction of 50% marks.
5. Marks are allocated for standard programming, your creativity, ease of use and an error-free application.

### **Mark Allocation Sheet**

After having discussed this as a group, we recommend the following mark allocation to each group member based on contribution or lack of it throughout the assignment.

**Group Name** \_\_\_\_\_

**Project manager** \_\_\_\_\_

Member ID	Allocated task and percentage contribution

Certification		
ID	Member name	Signature

**Due Date – please refer to Moodle.**

## Assessments mapping with CBOK

Core Body of Knowledge		CS112	Assign1	Assign2	Assign3	Mid Sem Test	Online Act
	Complex Computing	B	✓	✓			
ICT Professional Knowledge	Ethics	B	✓	✓			
	Professional expectations						
	Teamwork concepts/issues	B		✓			
	Communication						
	Societal Issues/Legal issues/Privacy						
	Understanding the ICT profession						
ICT Problem Solving:	Abstraction	B		✓			
	Design						
Technology Resources	Hardware and Software Fundamentals						
	Data and Information Management	B		✓			
	Networking						
Technology Building	Human Factors						
	Programming	B	✓	✓			
	Systems Development / Acquisition	B	✓	✓			
ICT Management	IT Governance and organisational issues						
	IT Project management						
	Service management						
	Security management						