The University of the South Pacific

School of IT, Engineering, Mathematics & Physics

CS112: Data Structures & Algorithms

Assignment 1 – Semester II, 2023

Total Marks: 10%

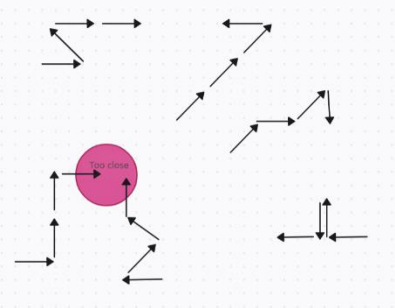
Due Date: as shown on Moodle

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 focus is on Arrays. The usage of struct or class will also be tested. C++ is the only acceptable language for this assignment. This assessment covers the following course learning outcomes:

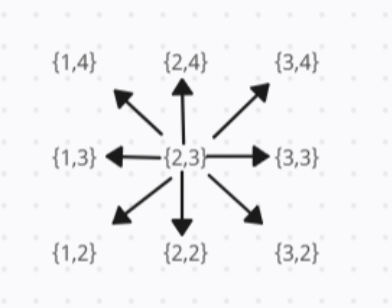
CLO 1 – Demonstrate the usage of various data structures in programming.

Suppose you have been hired by Digital Fiji to prepare a Pandemic Outbreak Tracing App (POTA) to prepare the country and regions better for another COVID-19-like outbreak. A person with the symptoms of the disease should stay in confinement. If he/she moves around then there is a danger of passing the disease to others due to the highly contagious nature of the disease. As a programmer, you are required to do the following:

1. Create a Person class/struct whose movement should be tracked with his position as latitude and longitude. You can simply use int x and int y in a 2D cartesian plane to make a simulation. Also, use some sort of ID to identify a person. [1.5%]
2. Create 100 people and put them in random locations (use values like {2,3} or {8,11} to indicate the location). Suppose distances are in meters. Make the limits [-100, 100] for both X and Y axis. Make random 10% of the population carrier of the disease. [2%]
3. Now create a simulation for n days. Every day a person can move up to ± 1 distance in 2D cartesian plane for simplicity. For example, a person in the current location of {2,3} can stay in {2,3} or move to either {1, 2} or {1, 3} or {1, 4}] or {2, 4}, {3, 2} or {3, 3} or {3, 4} or {2, 2} as shown in the diagram below. Do some display to show that this feature works. [2% - 1 for SD]



Movement of population



Movement of individual

4. During the n days of the movement, a healthy person has a 90% chance of getting infected by a carrier who is within the proximity of at least $\sqrt{2}$ meters. Your system must keep updating the proliferation of new infections as they start infecting other healthy people.
[2 % - 1 for SD]

5. Eventually, you will have to track how many people are infected at the end of n days.
[1 % - 1 for SD]

6. Now try to trace or print the movements of these people (along with the infection indicator). This carries low marks, but you will get +1 bonus marks for doing this part well. You may like to use the `system("cls")` function here. [1.5 + 1 bonus – 1.5 for SD]

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 simple code for a well-defined problem II. Use of basic standard programming practices such as commenting, indentation etc. III. computer program produces correct output.	I. All satisfactory and demonstrate very good programming skills.	**5.5**	
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.	**4.5**	
Sub Total & comments					

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. Completely fill Mark Allocation Sheet and submit it with your assignment. Failing to do so may result in a deduction of 50% marks.
4. This assignment can be submitted in groups of up to 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.
5. Marks are allocated for standard programming, your creativity, ease of use and an error-free application.

Mark Allocation Sheet

After having discussed as 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	Percentage contribution of allocated task

Certification		
ID	Member name	Signature

Due Date – please refer to Moodle.

Assessments mapping with CBOK

Core Body of Knowledge		CS112	Assign1	Assign2	Assign3	Test 1	Test 2
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						