INTI International University & Colleges

UNIVERSITY OF WOLLONGONG AUSTRALIA

FACULTY OF ENGINEERING AND INFORMATION SCIENCES

SUBJECT'S INFORMATION:			
Subject:	CSCI251 Advanced Programming		
Session:	Spring 2019 (July session)		
Programme / Section:	Computer Science		
Lecturer:	Ms. Siti Hawa		
Coursework Type *(tick appropriate box)*	✓ Individual Assignment □ Lab Task	□ Group Assignment □ Seminar / Tutorial Paper	□ Project □ Others
Coursework Title:	**Assignment 3**	Coursework Percentage:	10%
ASSESSMENT CRITERIA:			
Correctness	All programs should produce the correct result as stated in the specification.	2 marks	
Use of STL containers	Use correct STL containers with suitable functions and iterators in the program.	3 marks	
Use of STL generic algorithm	Correct and suitable use of a minimum of two generic algorithms	1 mark	
Main Function	Good design of the main function with complete functionalities included. Well-structured and used modular approached. Necessary data validation is implemented.	2 marks	
Readability	Appropriate comments are included. Meaningful identifiers used. Proper indentation and line spacing used.	1 mark	
Well formatted output	Output should be well formatted with appropriate messages displayed. Numbers are shown with appropriate precision.	1 mark	
SUBMISSION:			
All completed work should be submitted online through Moodle before or on the due date provided. **SUBMIT AS EARLY AS POSSIBLE. ONLY ONE SUBMISSION IS ALLOWED. IF RE-SUBMISSION IS NECESSARY, YOU ARE REQUIRED TO REMOVE THE EARLIER SUBMISSION AND THIS MUST BE DONE BEFORE THE DUE DATE. OTHERWISE YOU WILL BE PENALIZED FOR LATE SUBMISSON.**			
DUE DATE:	**Monday, 18th November 2019 (11:55 pm)**		
PENALTIES FOR LATE SUBMISSION:			
Penalties apply to all late work, except if student academic consideration has been granted. Late submissions will attract a penalty of 25% of the assessment mark per day including the weekend. Work more than (3) days late will be awarded a mark of zero.			
PLAGIARISM:			
When you submit an assessment task, you are declaring the following 1. It is your own work and you did not collaborate with or copy from others. 2. You have read and understand your responsibilities under the University of Wollongong's policy on plagiarism. 3. You have not plagiarised from published work (including the internet). Where you have used the work from others, you have referenced it in the text and provided a reference list at the end ot the assignment.			

Plagiarism will not be tolerated. Students are responsible for submitting original work for assessment, without plagiarising or cheating, abiding by the University's policies on Plagiarism as set out in the University Handbook under University Policy Directory and in Faculty handbooks and subject guides. under University Policy Directory and in Faculty handbooks and subject guides.

COURSEWORK SPECIFICATION

OBJECTIVES

This assignment aims to provide you with some experience in writing codes using C++ programming language that covers the following topics:

- STL Containers, Iterators, and Generic Algorithm

Remember that:

1. **All programs should be able to run on the lab's computers.**
2. **You must put the following information on the header of each text and source file you will be submitting in this assignment:**
   Student's full name:
   Student's ID:
   Modification Date:
   Purpose of this file (or program):
3. **Assignments that are not able to be compiled will result in zero mark given to the assignment.**
4. **You must only use the C++ features that have already been covered in the lectures**

TASKS:

For this task, you are required to write a program using STL to simulate the process of managing sports facility available at a sport center. A sport center may provide a variety of sport facility such as futsal court, badminton court, softball batting cages, etc. The sport center may have more than one place for each sport facility. Each member who would like to use the facility at the sport center will have to register for booking. A facility can be assigned to the member upon availability of the facility. Assignment is done on a first come first serve basis.

Design and implement a class called SportFacility. Each object from this class should keep the information of the facility number/id (this could represent the court number or batting cage number), the description (such as "Futsal Court", "Badminton Court", etc), the capacity (how many member can use the facility at a time), an indicator to mark whether the facility is occupied or not, a duration (number of hours the court is occupied), the start time (the starting time for the facility to be occupied), and rental fee for the facility.

Use the STL multiset to keep the SportFacility objects that are sorted according to the description. You will need to implement a suitable function object to describe how to order the SportFacility objects. At the start of the program, populate the set using data read from a text file. Design a suitable file format for this purpose.

Next, design and implement a Member class holding data that represents the name and id number of a member of the sport center. When a member arrives to book a facility a new object from this class should be instantiated and push into a STL Adapter class queue. Each member will be assigned to a facility of their choice on a first come first serve basis. You may need to create multiple queues for different types of facility available at the sport center. Think of how this can be done.

Lastly, create a STL map container that will be used to keep the records of booking. The Member object should be the key and the SportFacility id should be the value stored in each pair of the map items. A map item will be added when a booking is made and removed when the facility is no longer used by the member.

Use a menu driven program to simulate the arrival of members, booking of sport facility, and leaving the sport facility. When a member arrives, dynamically create a member object and push it to the queue containing the members waiting to book a facility.

When booking a sport facility, search for the sport facility record in the STL multiset and if the facility is not occupied remove the member's record from the queue and add a new booking record to the STL map. There may be multiple instances of the same type of facility. Your program should be able to check all facility of the same type before indicating that the facility is not available.

When a member leaves the sport facility, the related object should be updated to indicate it is now no longer occupied and remove the record from the STL map. In addition to that, your program should also instantly check the queue if there are any member waiting to use the facility. If there are, perform the booking operation for this member.

Additional functions such as displaying all available sport facility and all occupied sport facility should be implemented. Think of suitable operators to be overloaded for your classes to allow operations to be performed correctly when you store your objects in the STL containers.