INTI International University & Colleges™ UNIVERSITY OF WOLLONGONG AUSTRALIA

FACULTY OF ENGINEERING AND INFORMATION SCIENCES

SUBJECT'S INFORMATION:			
Subject:	CSIT121 Object Oriented Design and Programming		
Session:	Autumn 2019 (February)		
Programme / Section:	Computer Science and IT		
Lecturer:	Ms. Siti Hawa		
Coursework Type *(tick appropriate box)*	❑ Individual Assignment ❑ Group Assignment ❑ Project ✓ Lab Task ❑ Seminar / Tutorial Paper ❑ Others		
Coursework Title:	**Lab Task 6**	Coursework Percentage:	3%
ASSESSMENT CRITERIA:			
Correctness	All programs should produce the correct result as stated in the specification.		
Coding and GUI design	Creative use of GUI components and correct event handling performed.		
Readability	Appropriate comments are included. Meaningful identifiers used. Proper indentation and line spacing used.		
Well formatted output	Output should be well formatted with appropriate messages displayed. Numbers are shown with appropriate precision.		

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:	**WEEK 16**

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.

COURSEWORK SPECIFICATION

OBJECTIVES:

Following completion of this task, students should be able to:

- Write Java applications using lambdas and streams.

Question 1

Write a Java application that declares an array of 20 elements to store the prices of items sold at a grocery store. Initialize the array with 20 prices. Then, using lambda and streams perform the following operations on the array created:

a) Display all prices sorted in ascending order
b) Display the total prices.
c) Display the most expensive price, the cheapest price, and the average price.
d) Count and display all prices that are above RM100 sorted in ascending order
e) Increase all prices to 10% of its original price, sort, and display the new prices.
f) Display all prices that are in the range of RM100 and RM500 inclusive.

Question 2

Declare and implement a class called Invoice with the following fields:

- part number (an integer value)
- description (such as hammer, wrench, sugar, flour, etc)
- type (such as hardware, grocery, etc)
- quantity (an integer)
- price (a double value)

Implement the following methods for the class Invoice:

- a non-default constructor that takes parameters to set all of its fields
- set methods for each of the fields
- get methods to retrieve each of the fields

Use the class Invoice above to create an array of Invoice objects. Use the data given in the data.txt file to set the values for each of the objects in the array. Then, perform the following queries on the array of Invoice objects and display the result.

a) Use lambdas and streams to sort the Invoice objects by description, then display the results.

b) Use lambdas and streams to sort the Invoice objects by price, then display the results.

c) Use lambdas and streams to display the Invoice objects group by type.

d) Use lambdas and streams to map each Invoice to its description and the value of the invoice (quantity * price). Order the results by Invoice value.

e) Re-write (d) to select the Invoice values in the range of RM200 to RM500.

Question 3

Write a recursive method called power(), that takes as parameters two integers x and y such that x is non-zero and returns x^y. Use the following recursive definition to calculate x^y.

If $y \geq 0$:

$$x^y = \begin{cases} 1 & \text{if } y = 0 \\ x & \text{if } y = 1 \\ x \times x^{y-1} & \text{if } y \neq 0 \end{cases}$$

If $y < 0$:

$$x^y = \frac{1}{x^y} .$$

Write a main method to test your recursive method above. Let the user enters the value for x and y. Call the recursive method and display the result.