ASSESSMENT BRIEF

COURSE: Bachelor of Information Technology	
Unit:	Object Oriented Design and Programming
Unit Code:	OODP101
Type of Assessment:	Assessment 3 – Solution to programming problem by group of 3-4 students
Length/Duration:	20 Hours
Unit Learning Outcomes addressed:	Upon successful completion of this unit students should be able to: 1. Demonstrate basic knowledge of object-oriented programming concepts and programming problems 2. Analyse and dissect simple design and programming problem 3. Implement a well-designed modularized solution to small programming problems 4. Develop and/or implement testing schedules
Submission Date:	Week 8
Assessment Task:	A group of 3-4 students will work together to provide a quality solution to programming problem using JAVA programming language,
Total Mark:	20 marks
Weighting:	Converted to 20% of the unit total marks

Students are advised that **submission of an Assessment Task past the due date without a formally signed approved Assignment Extension Form** (Kent Website *MyKent Student Link> FORM – Assignment Extension Application Form – Student Login Required)* **or previously approved application for other extenuating circumstances impacting course of study, incurs a 5% penalty per calendar day**, calculated by deduction from the total mark.
For example. An Assessment Task marked out of 40 will incur a 2 mark penalty for each calendar day.

More information, please refer to *(*Kent Website *MyKent Student Link> POLICY – Assessment Policy & Procedures – Student Login Required)*

Kent Institute Australia Pty. Ltd.
ABN 49 003 577 302 CRICOS Code: 00161E RTO Code: 90458
TEQSA Provider Number: PRV12051

ASSESSMENT DESCRIPTION:

Your task is to design, develop and test a small application which will allow a mobile phone user to compare the cost of their phone usage on particular day under plans from three different phone providers and find the most expensive and cheapest from them.

Task 1- Design

This stage requires you to prepare documentation that describes the function of the program and how it is to be tested. There is no coding or code testing involved in this stage.

Requirements:

1) Read through *Task 2: Program Development* to obtain details of the requirements of this program.
2) Write pseudocode that describes how the program will operate.
    a. All program requirements must be included, even if you do not end up including all these requirements in your program code.
    b. The pseudocode must be structured logically so that the program would function correctly.
3) Prepare and document test cases that can be used to check that the program works correctly once it has been coded. You do NOT need to actually run the test cases in this stage; this will occur in *Task 3: Testing*.
    a. Test cases should be documented using a template which is week 6 lecture and tutorial. You may include extra information if you wish. At this stage, the Actual Result column will be left blank. Two test cases per group member are required to gain full marks in this task.

Task 2: Program Development

Using the Design Documentation to assist you, develop a Java program that allows the user to enter details of their phone usage and then compare the bill which would result from this usage under different billing plans.

All requirements require that you follow coding conventions, such as proper layout of code, using naming conventions and writing meaningful comments throughout your program.

Requirement 1:

Display a welcome message when the program starts

- The welcome message should have a row of "*" at the top and the bottom, just long enough to extend over the text. *Hint: Use a loop for this.*
- The first line of the message should read "WELCOME TO PHONE BILL COMPARISON SYSTEM"
- The second line of the message should be blank.
- The third line should read "Developed by" followed by your names and a comma, then "student ID", then your student ids of all group members.
- The fourth line should display "OODP101 Object Oriented Design and Programming"
- The fifth line should display the current date and time of system. You are expected to do a research to complete this task.
- The sixth line should be blank, and the seventh line should be another row of "*"

```
**********************************************************************************************************
WELCOME TO PHONE BILL COMPARISON SYSTEM

Developed by groupMemberNames, student Ids: YourIDs
OODP101 Object Oriented Design and Programming
2020/08/17 10:43:47

**********************************************************************************************************
```

Requirement 2

Provide a menu from which the user can select to Enter Usage Details, Display Cost Under Provider 1, Display Cost Under Provider 2, Display Cost Under Provider 3, Clear Usage, or Exit System. This menu should be repeated each time after the user has chosen and completed an option until the user chooses to Exit. The user selects an option by entering the number next to it. If an invalid number is selected, the user is advised to make another selection.

```
MAIN MENU

Please select from the menu:
1. Enter Usage Details
2. Display Cost Under Provider 1
3. Display Cost Under Provider 2
4. Display Cost Under Provider 3
5. Clear Usage Details
6. Exit System

Enter your selection: 7

Value must be between 1 and 6. Please try again.
Enter your selection: 0

Value must be between 1 and 6. Please try again.
Enter your selection:
```

Requirement 3

When the user selects the Enter Usage Details option, provide another menu from which the user can select Phone Call, SMS, Data Usage, or Return to Main Menu. The user selects an option by entering the number next to it. If an invalid number is selected, the user is told to make another selection.

```
ENTER USAGE DETAILS MENU

Please select an option from the menu:
1. Phone Call
2. SMS
3. Data Usage
4. Return to main menu

Enter your selection:
```

Requirement 3.1

If the user selects Phone Call, they are prompted to enter the length of the call in seconds. If user selects this option more than once then it means that there are more than one calls that user had made on particular day so your program should be able to consider all calls in billing system. The value entered must be positive – if not, the user should be prompted to re-enter a new value. After entering a valid call length, number of calls should be displayed and the user is returned to the Enter Usage Details Menu so that they may choose to enter additional usage details.

```
ENTER USAGE DETAILS MENU

Please select an option from the menu:
1. Phone Call
2. SMS
3. Data Usage
4. Return to main menu

Enter your selection: 1
Enter call length in seconds:
34
Total number of calls so far=2


ENTER USAGE DETAILS MENU

Please select an option from the menu:
1. Phone Call
2. SMS
3. Data Usage
4. Return to main menu
```

Requirement 3.2

If the user selects SMS, the program should simply increment the count of the number of SMS messages and number of messages. No further information is required so the program should simply display the total number of SMS messages recorded so far, and then return to the Enter Usage Details Menu.

```
ENTER USAGE DETAILS MENU

Please select an option from the menu:
1. Phone Call
2. SMS
3. Data Usage
4. Return to main menu

Enter your selection: 2
Total number of SMS so far=1
```

Requirement 3.3

If the user selects Data Usage, they should be prompted to enter the amount of data used in MB. The value entered must be positive – if not, the user should be prompted to re-enter a new value. After entering a valid value, the user is returned to the Enter Usage Details Menu so that they may choose to enter additional usage details.

```
ENTER USAGE DETAILS MENU

Please select an option from the menu:
1. Phone Call
2. SMS
3. Data Usage
4. Return to main menu

Enter your selection: 3
Enter the amount of data in MB:
34
Data amount so far 34MB
```

Requirement 4

When the user selects the Display Cost Under Provider 1 option, the program should display a summary of the usage details which have been entered, and their cost under Provider 1, along with the total cost, formatted as shown in the screenshot below. The cost structure for Provider 1 is listed in the following table. Once the bill summary has been displayed, it's total value should be saved into an array and the program should return to the Main Menu.

Usage Item	Item Cost – Provider 1
Per phone call (flag fall charge)	$0.20
Per second of total time over all phone calls	$0.03
Per SMS	$0.10
Per MB of data usage	$0.02

```
Enter your selection: 2
Cost under Provider 1
**********************************************
Number of calls =2              $0.4
Total call time(secs) =34       $1.02
Number of SMS =2                $0.2
Data Usage (MB) =23             $0.46
**********************************************
TOTAL COST                      $2.08
```

Requirement 5

When the user selects the Display Cost Under Provider 2 option, the program should do the same as in Step 4, but using Provider 2's cost structure instead, which is listed in the following table,and then return to the Main Menu.

Usage Item	Item Cost – Provider 2
Per phone call (flag fall charge)	$0.15
Per second of total time over all phone calls	$0.04
Per SMS	$0.12
Per MB of data usage	$0.04

Requirement 6

When the user selects the Display Cost Under Provider 3 option, the program should do the same as in Step 4, but using cost structure that will be developed by you and then return to the Main Menu.

Usage Item	Item Cost – Provider 3
Per phone call (flag fall charge)	---
Per second of total time over all phone calls	---
Per SMS	---
Per MB of data usage	---

Requirement 7

When the user selects Clear Usage Details the value of all variables related to the usage (number of calls, total length of calls, number of SMS, total data usage) should all be reset to 0. A message reporting this should be displayed, and the program should return to the Main Menu.

```
Please select from the menu:
1. Enter Usage Details
2. Display Cost Under Provider 1
3. Display Cost Under Provider 2
4. Display Cost Under Provider 3
5. Clear Usage Details
6. Exit System

Enter your selection: 5
**************************************************************
ALL USAGE DETAILS HAVE BEEN RESET TO 0
**************************************************************

MAIN MENU

Please select from the menu:
1. Enter Usage Details
2. Display Cost Under Provider 1
3. Display Cost Under Provider 2
4. Display Cost Under Provider 3
5. Clear Usage Details
6. Exit System
```

Requirement 8

When the user selects Exit System, quit the program with a message to the user which will show which is the expensive and cheapest provider by using the values stored in the array.

Requirement 9

Modularize the code, correctly using method calls and passing data between methods as parameters.

Task 3: Testing

After finishing the development, test your program with the help of test cases developed task 1. In this task, you will be giving the actual output of your program. Make sure you provide screenshots in your report of all actual outcome of all test cases. You don't need to rewrite the test cases in this task but you definitely need to provide the proper numbers so your teacher can identify the relevant test cases from your screenshots.

ASSESSMENT SUBMISSION:

This submission will have one word/pdf and one java file.
This assignment should be submitted online in Moodle **.**

The assignment MUST be submitted electronically in Microsoft Word format. Other formats may not be readable by markers. Please be aware that any assessments submitted in other formats will be considered LATE and will lose marks until it is presented in Word.

For assistance please speak to our Academic Learning Skills Coordinators, in Sydney (als_syd@kent.edu.au) or in Melbourne (als_mel@kent.edu.au). They can help you with understanding the task, draft checking, structure, referencing and other assignment-related matters.

GENERAL NOTES FOR ASSESSMENT TASKS

Content for Assessment Task papers should incorporate a formal introduction, main points and conclusion.

Appropriate academic writing and referencing are inevitable academic skills that you must develop and demonstrate in work being presented for assessment. The content of high quality work presented by a student must be fully referenced within-text citations and a Reference List at the end. Kent strongly recommends you refer to the Academic Learning Support Workshop materials available on the Kent Learning Management System (Moodle). For details please click the link http://moodle.kent.edu.au/kentmoodle/mod/folder/view.php?id=3606 and download the file titled "Harvard Referencing Workbook". This Moodle Site is the location for Workbooks and information that are presented to Kent Students in the ALS Workshops conducted at the beginning of each Trimester.

Kent recommends a minimum of **FIVE (5)** references in work being presented for assessment. Unless otherwise specifically instructed by your Lecturer or as detailed in the Unit Outline for the specific Assessment Task, any paper with less than five (5) references may be deemed not meeting a satisfactory standard and possibly be failed.

Content in Assessment tasks that includes sources that are not properly referenced according to the "*Harvard Referencing Workbook*" will be penalised.

Marks will be deducted for failure to adhere to the word count if this is specifically stated for the Assessment Task in the Unit Outline. As a general rule there is an allowable discretionary variance to the word count in that it is generally accepted that a student may go over or under by 10% than the stated length.

GENERAL NOTES FOR REFERENCING

References are assessed for their quality. Students should draw on quality academic sources, such as books, chapters from edited books, journals etc. The textbook for the Unit of study can be used as a reference, but not the Lecturer Notes. The Assessor will want to see evidence that a student is capable of conducting their own research. Also, in order to help Assessors determine a student's understanding of the work they cite, all in-text references (not just direct quotes) must include the specific page number(s) if shown in the original. Before preparing your Assessment Task or own contribution, please review this 'YouTube' video (Avoiding Plagiarism through Referencing) by clicking on the following link: link: http://moodle.kent.edu.au/kentmoodle/mod/folder/view.php?id=3606

A search for peer-reviewed journal articles may also assist students. These type of journal articles can be located in the online journal databases and can be accessed from the Kent Library homepage. Wikipedia, online dictionaries and online encyclopaedias are acceptable as a starting point to gain knowledge about a topic, but should not be over-used – these should constitute no more than 10% of your total list of references/sources. Additional information and literature can be used where these are produced by legitimate sources, such as government departments, research institutes such as the National Health and Medical Research Council (NHMRC), or international organisations such as the World Health Organisation (WHO). Legitimate organisations and government departments produce peer reviewed reports and articles and are therefore very useful and mostly very current. The content of the following link explains why it is not acceptable to use non-peer reviewed websites (Why can't I just Google?): https://www.youtube.com/watch?v=N39mnu1Pkgw
(Thank you to La Trobe University for access to this video).

MARKING GUIDE (RUBRIC):

Your answers for the final examination questions will be assessed as per the following marking criteria. Please read carefully each section/level and marks weightage.

Marking Criteria	Marks
Task 1 Design	
Pseudocode (Well written following all points discussed in class and including all requirements from task 2)	2
Test Cases (two per team member, clearly indicate the test case, test data, expected output)	2
Task 2 Development	
Requirement 1	
Welcome message displays all necessary details as given in screenshot	1
Requirement 2	
Main Menu is displayed, and user is prompted to enter valid value if invalid value is entered. Proper loop has been used to do this task.	1.5
Requirement 3	
User details menu is displayed, and user is prompted to enter valid value if invalid value is entered. Proper loop has been used to do this task.	1.5
Requirement 3.1	
Value of length of call is saved and number of calls so far is displayed correctly. Invalid values are handled properly with error message	1
Requirement 3.2	
Value of number of SMS is saved and number of SMS so far is displayed correctly. Invalid values are handled properly with error message	1
Requirement 3.3	
Value of amount of data is saved and number of SMS so far is displayed correctly. Invalid values are handled properly with error message	1

Requirement 4 Total cost is calculated, saved in array and displayed properly using the values given in cost under provider 1.	1
Requirement 5 Total cost is calculated, saved in array and displayed properly using the values given in cost under provider 2.	1
Requirement 6 Total cost is calculated, saved in array and displayed properly using your own values of provider 3.	1
Requirement 7 Setting all values to zero and displaying the message as given screenshot.	1
Requirement 8 Program display the exit message having details about expensive and cheapest provider and exit the program.	1.5
Requirement 9 Code should have four modules (should demonstrate the use of parameters, arguments and return values)	1.5
Task 3 Testing All actual output screenshots are provided with the test cases.	2
Total	**20**