**BACHELOR OF COMPUTER SCIENCE**

**SCHOOL OF COMPUTER SCIENCE**

**BINA NUSANTARA UNIVERSITY**

**JAKARTA**

**ASSESSMENT FORM**

**Course:** **COMP6047001 - Algorithm and Programming**

**Method of Assessment:** **Case Study**

**Semester/Academic Year : 1/2022-2023**

**Name of Lecturer : INDRA DWI RIANTO, S. KOM, S.SI., M.T.I**

**Date : Sabtu,21 Januari 2023**

**Class : LC01**

**Topic : Material Review II**

|  |  |
| --- | --- |
| **Group Members :** | 1. Kevyn Aprilyanto -260208973   2\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  3\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  4\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  5\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  6\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  7\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  8\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ |

**Student Outcomes:**

(SO 2) Mampu merancang, mengimplementasikan, dan mengevaluasi solusi berbasis komputasi untuk memenuhi serangkaian persyaratan komputasi dalam konteks ilmu computer

*Able to design, implement, and evaluate a computing-based solution to meet a given set of computing requirements in the context of computer science.*

**Learning Objectives:**

(LObj 2.2) Mampu mengimplementasikan solusi berbasis komputasi untuk memenuhi serangkaian persyaratan komputasi tertentu dalam konteks ilmu komputer

*Able to implement a computing-based solution to meet a given set of computing requirements in the context of computer science*

| **No** | **Assessment criteria** | **Weight** | **Excellent (85 - 100)** | **Good (75-84)** | **Average (65-74)** | **Poor (0 - 64)** | **Score** | **(Score x Weight)** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | Ability to identify the problems and explain the solution | **25%** | The problem is well defined and solution is clearly explained and detailed | The problem is well defined and solution is less clearly explained | The problem is defined and solution is less clearly explained | The problem and solution are badly defined and explained respectively | **100** | **25** |
| 2 | Apply logical thinking | **25%** | Correctly and effectively applying logic thinking to solve the problem. | Correctly applying logic thinking to solve the problem | Partially correct applying logic thinking to solve the problem | Incorrectly applied the logic thinking | **100** | **25** |
| 3 | Ability to construct a C program | **25%** | All the syntax in the program are correctly and effectively applied | All the syntax in the program are correctly applied | Only some of the syntax in the program are correctly applied | None of the syntax in the program are correctly applied | **100** | **25** |
| 4 | Ability to choose the appropriate algorithm | **25%** | Correctly choosing the most effective algorithm to solve the problem | Correctly choosing effective algorithm to solve the problem | Correctly choosing the algorithm to solve the problem | Incorrectly choosing the algorithm to solve the problem | **100** | **25** |
|  | **Total Score:** ∑(Score x Weight) | | | | | | | **100** |

Remarks:

**ASSESSMENT METHOD**

Instructions

1. This is an individual assignment and will be held in review topic session with duration of 1 week, or week 13.
2. You will be given 3 questions. The questions are study case.
3. There will be 4 files that you need to upload (and zipped):
   1. One is a word file. This file consists of your explanation of solution. The explanation should be given using flowchart for better explanation. A correct and detailed flowchart will give higher score.
   2. Another 3 is .c files of your solution which will be checked by the respective lecturers.
4. Don’t cheat.

**Note for Lecturers**:

1. This case study assignment will be held with duration of 1 week in review topic, or week 13.
2. The answer is manually checked by each lecturer (not by system).
3. You may refer to the rubric table given above.

**Questions**

You are required to write a program that can read a file and perform several functions. The file needed can be downloaded from link:

<https://1drv.ms/u/s!AhuAx03LAKWtnOM9O1wlXSAR84Z67g?e=lVmH5x> .

File description:

This file is a csv file; therefore, each column is separated with comma. This file consists of 3939 rows of Housing Data in Malaysia with no missing value for each row. This file also has a **header. Therefore, when your program loads the data, your program should be able to skip this header before passing the data into your record variable**. A glimpse of the data:

Text

Description automatically generated

You are required to perform 3 functions as follows:

1. **(40 Points) Describe**.

This function explains the information from each column. When running this function, **give a prompt to get input from the user** as the name of the column you want to describe. Then, display:

* 1. For column loc1, loc2, room, bathrooms, carparks, type, or furnish, display:
     1. Frequency for each unique value
     2. Maximum frequency
     3. Minimum frequency

For example, if we call describe function followed with loc1 as column name, the program should display like this:

Text

Description automatically generated

* 1. For area and price column, display only:
     1. Minimum value
     2. Maximum value
     3. Average value

**Because area and price value are not discrete, therefore there isn’t a need to perform frequency check for each of it.   
Draw the Flowchart of your solution.**

**Diagram

Description automatically generated**

1. **(40 Points) Search Data.**

To be able to handle search function, ask user to give input with format:

DataX in ColumnName

Your program should be able to parse above input. **It is prohibited to use 3 string input**. Then, display all data that has that DataX.

For example:

Partly in furnish

Above command will result in:

A picture containing text, computer

Description automatically generated

However, your program also should be able to search using only sub-string, for example:

kepong in loc1

will result in:

Calendar

Description automatically generated with low confidence

* **Give information if data searched doesn’t exist in the record.**
* **YOU ONLY HAVE TO DO SEARCH IN THE COLUMN OTHER THAN AREA AND PRICE.**

**Draw the Flowchart of your solution.**

**Diagram

Description automatically generated**

1. **(20 Point)** Please refer to **textbook Paul J. Deitel (2016). C how to program: with an introduction to C++, Chapter 8, Exercise 8.21 (page 404). *Alphabetizing a List of String.***In the original question, you are asked to alphabetize list of string using 10 to 15 name of towns. **For this case, implement what is being asked using column Location 1.**

**Draw the Flowchart of your solution.**

Diagram

Description automatically generated

**Note:**

1. All solutions should be built using C code.
2. As mentioned in each question, you also need to give draw Flowcharts from your solution for each case number. If your solution contains more than 1 sub program, each sub program should also be drawn. For example:

Diagram

Description automatically generated

Each Function A and Function B must have its own flowchart diagram.