

# Fundamentals of Programming Mini Project

2023

## 1 Instructions

- Form a group of 3 to 4 students per team.
- Include your team members' Names and Student ID as comments in the program.
- Include in the report a list of responsibilities and tasks of each team member in the peer review evaluation form.
- Submit the deliverables to Blackboard before the deadline, otherwise late penalties will apply.
- The group assignment will contribute to your overall coursework marks.

## 2 Deliverables

Your team is required to submit the following items:

- Written report
- Peer review evaluation form
- Source code
- Video recording of a presentation (5–10 minutes at resolution between 480–720 px in mp4, mkv, or mov format)

## **2.1 Written Report**

Submit a written report in English, between 1000 to 2000 words, which contains the following items:

- Introduction
- Discussion on the program, code design, and software testing
- Conclusion
- Appendix
  - Flowcharts
  - Pseudocode
  - Source code

## **2.2 Source Code**

Submit the source code in files with the suitable file extensions.

## **2.3 Marking Scheme**

- Written report, presentation, and peer review evaluation form = 10%
- Source code = 10%

## **Academic Misconduct Notice**

PSB Academy takes cases of academic misconduct involving plagiarism and cheating very seriously. Plagiarism is taking work made by someone else as your own. Academic cheating is subverting the coursework or examination process for your own benefit.

Students are required to conduct themselves honourably in their academic work. Any evidence of plagiarism will be penalised, which is not limited to the student receiving a failing grade. Refer to the latest Student Handbook regarding the issue of Academic Misconduct.

### 3 Problem

In this assignment, your team is required to develop a basic Customer Relationship Management (CRM) software program using C language. The CRM program should have the following features:

1. Contact Management
2. Sales Data
3. Sales Analytics
4. Sales Reports

#### Specifications

- The user interface (UI) for the program can be text-based. Graphics is not required.
- The **Contact Management** feature should allow the CRM user to enter and store the customer's information in an external text file. The information should be in the format:

```
char customer_id[8]
char customer_name[20]
char telephone[10]
char email[15]
```

- The **Sales Data** feature should allow the CRM user to write data to an external text file with the following data categories in the format below. (Note that total sales is the product of quantity and price.)

```
char customer_id[8]
char salesperson_name[20]
char item[20]
char year[4]
char month[2]
float quantity
float price
float total_sales
```

- The **Sales Analytics** feature should allow the CRM user to analyse the sales data with two data categories using any suitable statistical data analysis methods, such as linear regression, quadratic regression, and many more.
  - For example, there may be a need to analyse the effectiveness of each sales representatives against price and total sales achieved.
  - Note that when providing results from data analytics, the CRM program does not have to explain the meaning behind the results. However, the results presented in the Sales Report must be organised so that it is easy to interpret and understand.
- The **Sales Reports** feature should allow the CRM user to view the Sales Analytics results on the terminal screen or text file. Any other report on different aspects of the sales data may be presented as well. The design and format is up to your team to decide.

END OF GROUP ASSIGNMENT

---

Last updated: 2023-01-04

## Marking Scheme

Table 1: Marking scheme for written report

No	Item	Poor	Average	Good
1	Report contained sections and page numbers	0.0	0.5	1.0
2	Discussion on software design (functions, variables, etc.)	0.0	0.5	1.0
3	Discussion on algorithm applied in software	0.0	0.5	1.0
4	Design documents was provided (pseudocode, flowchart, etc.)	0.0	0.5	1.0
5	All functions were tested with proof of correctness	0.0	0.5	1.0
6	Report written in English with no spelling and grammatical errors	0.0	0.5	1.0
7	Appendices contained supporting materials	0.0	0.0	0.5
8	Peer review evaluation form	0.0	0.0	0.5
9	Presentation (with/without slides included)	0.0	2.0	3.0
Total				10.0

Table 2: Marking scheme for source code

No	Item	Poor	Average	Good
1	Source code compiled without any problems	0.0	0.5	1.0
2	Source code ran without any errors	0.0	0.5	1.0
3	Source code displayed expected result	0.0	0.5	1.0
4	Source code contained suitable comments	0.0	1.0	1.5
5	Important tasks were organised as functions	0.0	0.5	1.0
6	Function calls contained appropriate input and/or output parameters	0.0	0.5	1.5
7	Functions calls were declared from main()	0.0	0.5	1.0
8	Most variables were declared as local variables	0.0	0.5	1.0
9	Appropriate library functions or user-defined functions were used	0.0	0.5	1.0
Total				10.0