

ELEC362

Project

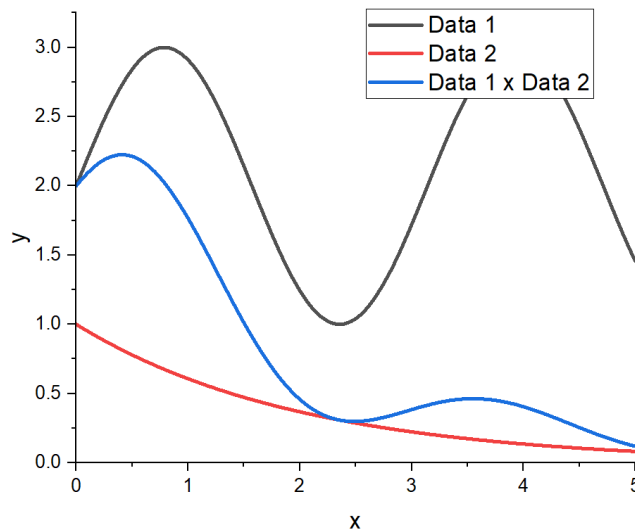
Data Processing application

Module	ELEC362
Coursework name	Project
Component weight	50%
Semester	1
HE Level	5
Lab location	personal computers/laptops, university remote computer
Work	Individual
*Estimated time to finish	40 hours (coding and testing)
Assessment method	Individually
Submission format	Online via VITAL
Submission deadline	23:59 on the 16th of December
Late submission	Standard university penalty applies
Resit opportunity	None except for extenuating circumstance
Marking policy	Marked and moderated independently
Anonymous marking	No (the marker needs to link the code to the report)
Feedback	via CANVAS
Expected release of marks date	10 business days from the deadline
Learning outcomes	LO2: Using C++ to implement GUI-based software. LO3: Using online documentation for self-learning.

***Note: This estimate may vary based on the need to debug your application. Make sure you start working on the project as soon as possible.**

The project

Task 1: Design and implement a Qt-based GUI application which allows the user to import different datasets, plot them in the application, and allow the user to create a function from a combination of the datasets.



To handle the plot, the library [QCustomPlot](#) will be used. Instructions for how to set it up and run it will be made available on a video tutorial.

The application should have the following specifications:

- Should have the ability to load as many datasets as needed.
- Should allow the user to choose which data sets will be plotted.
- Should have a legend for the data plotted.
- Should allow the user to write functions of the input dataset. Specifically; the addition of 2 datasets, their product, or the logarithm of one of them.
- Should allow the user to change the colours of the plotted graphs, using three different means.
- Should be validated, in the report, by using test cases, then comparing the plots to a plot provided by another programme such as Excel or MATLAB (No linking to Excel or MATLAB is required in the application).
- Should be able to deal with a range of potential run-time errors, which the developers should identify.

As a simplification, you may assume that all the data will have the same x axis.

Task 2: Modify the application so that the write function specification can be applied to an arbitrary number of data sets (not only 2 datasets as in task 1).

It is an expectation from everyone to consult Qt documentation to find any functions or classes that might be useful for the implementation of this project. Finally, you should make the programme as professional as possible. Imagine this project as a real-life project given to you, and that in the next years, it will be developed further by other students.

Note: The mark will be based on how your application behaves on a computer in the lab. Check your app on one of the computers in the lab to make sure it works as you expect before submitting your work

Approach to be followed

The following steps will help you to have a good design and implementation of the application:

- Start by conducting a background overview of the available programs for data visualisation and processing and how they are used. This will give you a feeling of what a professional design is and a better understanding of the core functionality. Make sure the practical notes given in the lectures are taken into consideration in your design and implementation.
- Follow the incremental model by breaking the implementation of the application into smaller parts, where every part is designed, implemented and tested before other parts are implemented.
- Make sure your application is working perfectly before moving to task 2.

Note for Mac OS users: You can develop the application and write the report based on Mac version of the app. The application will be tested and marked on a Windows machine. You should verify that your app works as it should on one of the computers in the lab before submission, to avoid any potential cross-platform issue, if you have any problems, please let me know.

The deliverables

Every submission should consist of the project's files in a zip file + brief report. The name of the zip file as well as the report must be your name (MohammadHasan.zip and MohammadHasan.pdf). The report must not be zipped. The report does not need an introduction or a conclusion, it must have a cover page showing the students name and ID number, and includes the following sections:

- How the programme works (explain the idea of the algorithm, include a flow chart).
- User instructions (how should the user use it)
- Testing and verification attempts.
- Overall notes (optional): in this section you can highlight any special features in your design or implementation of the application. Or you can mention any major issue you faced how you overcome it. If you left this section empty you will not lose any marks.
- Appendix: headers+ source files (must be copied-pasted as text, not as an image).

Industrial Relevance

Data visualisation and processing applications are essential tool in all fields of industry and academia, which apply to all fields of science from physics, chemistry and engineering to social sciences. The tasks given in this project are essential part of famous software including:



Mapping the project to modules' material:

Aspect	Where it is covered
Creating child windows or child dialogs.	Lecture 14, Lecture 16
Saving object in vectors in Qt	Week 9 lab
Loading and saving files	Lecture 16
Error handling	week 10 lab
Dealing with string sequences	Week 8 lab
Implementing mouse functionality	Lecture 14
Implementing a functionality using different methods	Lecture 14
Controlling the style, layout and GUI design	Video on CANVAS
Implementing buttons and other input widgets functionality	Lecture 14, week 8 lab
Adding QCustomPlot library to a project	Video on CANVAS
Online documentation	https://doc.qt.io

***Marking Criteria**

Criteria (weight %)	What does it mean?	Indicative characteristics	
		Adequate / pass (40%)	Very good / Excellent
Algorithm and design of the programme (task 1) (20%)	Does the application do what it is designed to do? Does the application handle memory efficiently?	<ul style="list-style-type: none"> • The programme does what it designed to do with clear shortcoming. • The use of memory is completely inefficient but is working. 	<ul style="list-style-type: none"> • The programme does what it designed to do without any flaws and in an efficient way • The memory management is very efficient.
Code and GUI implementation (task 1) (25%)	Is the code well organised? Are the variables named properly? Is the code well commented? Does the code make good use of Object-Oriented Programming style? Is the GUI user friendly?	<ul style="list-style-type: none"> • It is difficult to understand what the code does. • Poorly commented code. • The naming of the variables does not give any idea on what variables are used for. • The GUI is difficult to handle and is unorganised. • Layout of the GUI is not set. • Classes are poorly used or not used at all. 	<ul style="list-style-type: none"> • The code is written in very organised way that is easy to follow. • The code is very well documented. • The variable naming is appropriate and follows professional naming rules. • The GUI is very easy to work with. • The data structure used are the most appropriate for the purpose they are used for. • The code is structured in terms of classes and uses multiple aspects of OOP paradigm.
Verification and Error-Handling (task 1) (20%)	Has the application been validated and verified? What does the programme do if there is a run-time error?	<ul style="list-style-type: none"> • The application was tested for one case only. • The application contains minimal error handling. 	<ul style="list-style-type: none"> • The application has been thoroughly tested. • Error handling has been done professionally for 3 potential run-time errors and user input errors as well. • Error messages are meaningful.
Task 2 (15%)	Can the application handle functions involving multiple datasets?	<ul style="list-style-type: none"> • The app can handle multi-datasets function but has a lot of shortcoming 	<ul style="list-style-type: none"> • The app works can handle formulas involving multiple datasets without any flaw.
Report (20%)	Is the report clear and descriptive? Does it have all sections? Are the given user instructions clear?	<ul style="list-style-type: none"> • The report is understandable but contains many errors. • The length of some sections is inappropriate – either too short or too long. • The source codes are given in the appendix as an image. 	<ul style="list-style-type: none"> • The report is well written, and all the relevant information it should contain are clearly stated. • The report is concise.

***If you have a feedback query after the marks are released make sure you contact me within 2 weeks, after that the marks are finalised and sent to the Student Support Office.**