COMPUTER GRAPHICS PROGRAMMING
MOD006127 2020-21 TRI2
MULTIPHASED COURSEWORK 010

PORTFOLIO OF IN-CLASS EXERCISES (15%)

During the trimester, weekly practical classes will re-enforce principles of the associated lecture topic. Of these, there will be a number of programming exercises whose solutions should be archived into a portfolio and submitted along with the main assignment essay at the end of the trimester. These exercises should be worked on inside and outside of scheduled practical class time, and *students can support/help each other* provided that no actual code is exchanged between students, whether embedded in an email or attached as a separate file – however verbal discussion and/or communication by free text grammatical social media is encouraged.

MAIN COURSEWORK ASSIGNMENT (85%)

Extending a 2D drawing application

Program requirements
Many off-the-shelf packages utilise graphics drawing functionality (e.g. interior design, 3D-modelling, computer gaming). This assignment requires you to build a basic shape-drawing application from first principles. The file *GrafPack.cs* is provided for you, which contains the partial code for a WinForms application that currently just creates and displays only one square. You are required to develop the application so that various different shapes can be drawn with these shapes referenced in an array. The different shapes should all be defined as classes which inherit from a class Shape. Shapes should be created by discrete mouse clicks or rubber-banding (better). The main menu might contain the following options (you may change these if you wish):

Create: Produces a sub-menu for creation of Square, Circle and Triangle shapes.

Select: Allows the user to cycle through the shapes using mouse clicks or keyboard, highlighting the selected shape. Alternatively the shape may be selected by clicking on a vertex.

Transform: If a shape has been selected, this should produce a sub-menu with offerings such as Move and Rotate. The amounts of movement and rotation may be input from the keyboard or via the mouse, including 'drag and drop' operations.

Delete: If a shape has been selected, this will delete it from the screen and from memory.

Exit: Terminate the program cleanly.

The given classes may be modified, but only modifications for which documentation is supplied will be taken into account for marking. The language of implementation must be C# (ensures background equivalence – all students will have studied C#). Wherever possible you

should not use the drawing and processing features of the WinForms GDI or WPF but perform the operations from first principles using code you have written utilising some of the techniques outlined in this module. You may enhance the functionality beyond the above specification to provide added value (see mark scheme).

ASSESSMENT

Assessment of Portfolio
Each exercise will score 2 for functionality, 2 for code clarity, and 2 marks for description. Of 7 set exercises where only 4 are attempted/submitted, then max mark would be 4x6 = 24/42 = 8.6/15.

Assessment of Grafpack Assignment
The work will be assessed according to:

a) Functionality - program usability and spectrum of operations.
b) Shape and drawing development - class design, drawing methods.
c) Transformations, selections and deletions - use of data structure to store and access shapes (add, delete) and implement move, reflect, rotate etc.
d) Report evaluating what functionality you have been able to implement and the strengths and limitations of the implementation (400 words). Include instructions for running the program in an appendix (not included in word count) and a copy of the source code embedded in the report as a second appendix.

See the mark sheet for details of grade criteria.

Coursework submission
Structure your submission with a folder named after your sid e.g. 'sid1234567' Within this folder there should be two further folders, one called 'Portfolio' and the other called 'Grafpack' or similar.

The portfolio folder should itself contain one folder per assessed exercise, each containing the C# source code files (.cs) and the C# executable (.exe) file. There should also be a word doc/pdf description of how each solution was achieved for each exercise (50 words max per exercise) – this can be reported either all in one word doc/pdf file in the root of the portfolio folder or in individual short 1-page text files associated with each exercise folder.

The Grafpack folder should contain the report (see d) above), the C# source code (.cs) files and the C# executable (.exe). The submission, 'sid1234567' should then be compressed into a single zip file e.g. 'sid1234567.zip' and submitted through TurnitIn by 2pm on Friday 23rd April 2021. Note TurnitIn submissions have a file size limit of 100Mb; there should be no difficulty if you avoid submitting VS project files.

REASSESSMENT DETAILS

If a re-assessment is required you should re-submit the main coursework assignment (the Grafpack development) by the standard due date for Trimester 2 2020-21 resubmissions. Your mark out of 100% will be calculated on this main assignment submission only (usually the system will cap this at 40%). Note there will be NO portfolio exercises taken into account irrespective of how well they were undertaken at the first attempt. You are encouraged to explore a different approach to writing the main assignment compared to the first attempt.

MOD006127 Coursework feedback sheet 2020-21 Tri2									
	Practical exercises	**Grafpack development**							
Student	**/15** (not used in resubmission)	**Functionality**	**Shape and drawing development**	**Transformations, selection and deletions**	**Grafpack report**	**/85**	**Comment**	**Overall 010 mark %**	**Overall module grade**
	7 exercises set each with a score of 6 (2 for functionality, 2 for code clarity, 2 for description).	A: Intuitive, interface for many successful actions. B: Limited actions which are successful. C: Limited actions, some display errors. D: Unclear interface, wrong output. F: Cannot compile or multiple run time errors occur.	A: Multiple shapes defined and drawn by user-defined classes. B: Mix of user-defined vs library shape/drawing actions. C: A few shapes drawn by GDI or other libraries. D: A few shape definitions. F: No shape or drawing methods.	A: Many transforms actioned via coded matrix arithmetic. B: Many transforms actioned by GDI or 1 or 2 by coded matrix arithmetic. C: Effective use of array/list to manage selection. D: Uses array/list to store shapes. F: No use of data structure, shapes cannot delete.	A: Clear, accurate evaluation of pros/cons. B: Evaluation has weak grammatical style/errors. C: Accurately describes what has been implemented. D: Insufficient detail, poorly written. F: Substantial elements missing.		Personalized comments specific to student submission.	Grades are mapped to marks for both items	F (<40); D (40-49); C (50-59); B (60 - 69); A (70+)
Sid number	1.Circle=x 2.Triangle=x 3.XOR=x 4.Bounce=x 5.Points=x 6.Multiply=x 7.Rotate=x **XX.X**	**Grade**	**Grade**	**Grade**	**Grade**	**XX.X**	**Comment**	**Final Mark**	**Final Grade**