Computer Science and Creative Technologies

Coursework or Assessment Specification

Module Details

Module Code	UFCFQN-30-0
Module Title	Computational Thinking and Practice
Module Leader	Zaheer Khan
Module Tutors	Raj Ramachandran, David Coward, John Hunt, Jacob Baker
Year	2019-20
Component/Element Number	B / CW1
Total number of assessments for this module	3 – please refer to module specification
Weighting	50%
Element Description	Portfolio of Group Outputs

Dates

Date issued to students	23 September 2019
Submission Date	31 October 2019
Submission Place	Blackboard
Submission Time	14:00
Date to be returned to students	December 2019
Submission Notes	This is group submission. All parts must be collated into one final portfolio - as per the template in Appendix – B and submitted via Blackboard. As it is group submission via Blackboard, the final portfolio should be submitted by any one member of group before submission deadline. Group should nominate a group leader who will take the responsibility of portfolio submission. This should ideally be agreed between group members, at the latest, by the end of Week 5.

Feedback

Feedback provision will be	Formative interim feedbacks during sessions; Summative written feedback via blackboard

Contents

Section 1: Overview of Assessment

In addition to generally supporting your learning on this module, this assignment assesses the following module learning outcomes (taken from the module specification):

- *Formulate and model problems in various task domains, identifying significant features and how to apply an appropriate strategy to finding solutions.*
- *Translate solution to a problem into a computable form, and evaluate its effectiveness.*
- *Explore methods to represent solutions as a formal process, such as instructions, algorithms or pseudocode.*

The assessment is designed to allow you to play to your strengths and to use the expertise within your group to gain the best possible mark. You will find that you need to do a number of different tasks, individually as well in group, and to write them up in different ways.

You are expected to form a group of up to **four (04) students** during first week of study. Group members will self-organise meetings. As a good practice all group members should define team charter/contract very early on in the project. For example, group members will share their progress and issues faced and solutions found with group on daily basis and meet at least twice a week to discuss tasks completed and planning for next tasks. You should seek help of your tutors as early as possible and before deadlines if disputes or problems arise. You can always seek advice and/or interim feedback from your tutors during practical sessions.

In brief, you will create and submit a group portfolio of work in which your group has documented the details of a range of problems and their solutions for a selected case study. As depicted in Figure 1, the group members will identify set of problems for a case study. Your documentation will record not just the problems but also the process and techniques that your group has used to think about the problems. In addition you are also asked, individually, to select one (or more) identified problems and find solution(s) by applying computational thinking. Please record solutions as well as the process and computational concepts that you have used to find the solution. Finally, you are asked as a group to evaluate and reflect on how those solutions to individual problems will contribute collectively to selected case study. Please be concise and provide examples. You can use MS Word and other suitable tools to develop the portfolio and will submit via Blackboard submission system.

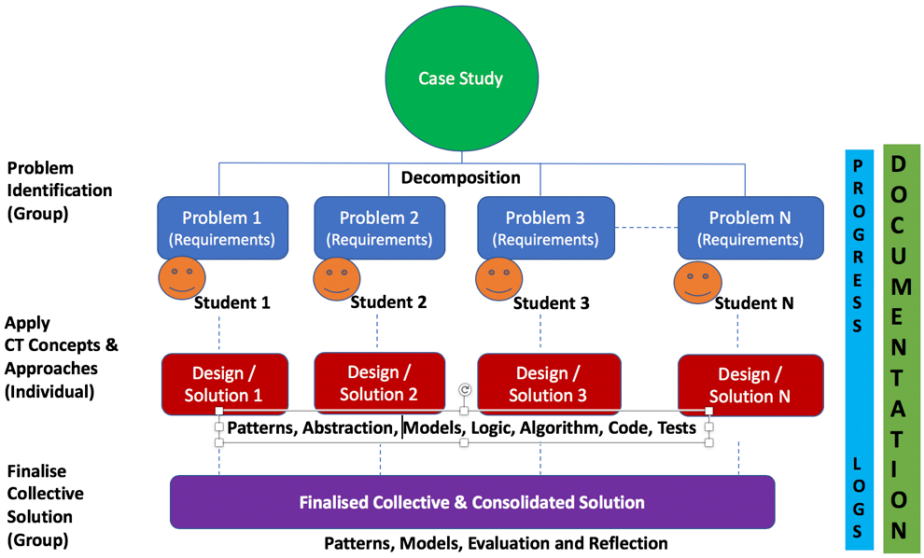

Figure 1: Case study – Problem Identification to finding and analysing solutions

The assignment is described in more detail in **section 2**.
The assignment is worth **50*%*** of the overall mark for the module and is a ***group*** assignment designed to ensure that all group members can equally contribute to individual as well as group parts.
Working on this assignment will help you to develop effective and systematic computational thinking skills. It will also help you to establish ways in which you can work as part of a group and to develop your reporting and writing skills. If you have questions about this assignment, please post them to the discussion board on Blackboard. You can find the discussion board from the front page of the module Blackboard pages under the title "Portfolio discussion".

Section 2: Task Specification

The portfolio consists of 4 parts. You **must** include all of the required elements in your submission. The group will just submit one portfolio. Group should nominate a group leader who will take the responsibility of portfolio submission. This should ideally be agreed between group members, at the latest, by the end of Week 5. Please read the assessment criteria carefully as there are marks for group activity as well as for individual work. All members of the group will get same marks for part 1 and part 3. For part 2 and part 4 marks will be allocated based on work completed by individual members of the group.
Note: It is expected that all group members will contribute equally to group parts as well as complete individual parts with all necessary inputs and it will be reflected in the final group portfolio to gain same marks.

Part 1 – Problem Identification and Requirements
In this part of the portfolio your group will select one of the case studies from Appendix - A at the latest by the **end of first week** and analyse it. Your group will agree on the contents and will write up:
i) a list of possible problems,
ii) the process and techniques that your group has used to think about the problems, and
iii) analysis of the problems and derive requirements which seem necessary to solve those problems.

This part of the portfolio is worth 30% of the marks for the portfolio. This amounts to 15 marks in terms of the overall module assessment. Each member of the group will get the same mark. [1]

Part 2 – Designing Solution

In this part each group member will select one or more problems identified in the part 1 and design a solution by applying suitable computational concepts and techniques. Individually, you should consider decomposition, pattern recognition, abstraction or modelling, algorithm/pseudocode design that would help solve the problem. Your design should be written in such a way that the pseudocode could be translated more or less line by line into Python code. You must also specify how you would test that your design is correct and give your test data. To gain top marks in this part, you will evaluate and reflect on the solution by suggesting at least 2 strengths and 2 weaknesses and how these can be improved.
Part 2 is worth 40% of the marks for the portfolio (20 marks in the overall module assessment). You will be awarded an individual mark for this part.

Part 3 – Consolidated solution

In this part you will work as group and identify common patterns from the solutions to individual problems (part 2) and model a consolidated solution to the case study. To gain top marks in this part, you should suggest a way that you could extend consolidated solution so that it is could be applied more widely/scaled.
This part of the portfolio is worth 20% of the marks for the portfolio. This amounts to 10 marks in terms of the overall module assessment. Each member of the group will get the same mark.

Part 4 – Learning logs

In this part you will submit your learning log. The portfolio should contain the learning logs for each member of the group. This part is worth 10% of the marks for the portfolio (5 marks in the overall module assessment). You will be awarded an individual mark for your learning log.
You should keep a learning log for each week that the module is running. It will be a log of what you have learned that week, how you have interacted with your group, any particular stumbling blocks that you have met and how you plan to overcome them.
Alongside the learning log each member of the group should refer to the marking criteria below and submit their own view of what mark their learning log should be awarded.
Please note that the actual mark will be based on your tutor's judgement of the log but the tutor mark will take your assessment into account.
Note: Please note that a template for this write up is provided in Appendix - B.

Section 3: Deliverables

You will create your portfolio using the template provided in Appendix - B and submit the complete portfolio through Blackboard. The template provides you with the headings that must be part of the portfolio. In addition to these please feel free to add to these if you feel you need to for your group's submission.

[1] The rule that each group member will get the same mark will only be varied if every member of the group agrees in writing that the marks should be apportioned differently. This will be on the discretion of module tutors.

Section 4: Marks Distribution and Marking Criteria

In common with all UWE standard undergraduate assignments, the pass mark for this assignment is 40%. The marking criteria for this portfolio is a little complicated and will be discussed further in class. Please speak to your tutor if your group is unsure about how many marks are attached to which part of the work or if you are unsure about how your work will be assessed. Detailed **assessment criteria** is in Appendix – C that will clarify how to get top marks. To simplify it, the marks breakdown is as follows:

Part 1 – Problem Identification and requirements. 30%

All group members will get same marks for this part.

Background	Case study context Problems identified and rationale provided	Up to 8 marks
Decomposition	Criteria used for decomposition of case study	Up to 6 marks
Requirements	Requirements analysis	Up to 10 marks
Commentary	Reflection on process and future improvements	Up to 6 marks
		Total 30 marks

Part 2. Designing Solution. 40%

Each member of the group will be have an individual mark.

Patterns	Similar Characteristics identified	Up to 6 marks
Modelling	System diagram(s)	Up to 8 marks
Design	Design of partial solution i.e. pseudocode/algorithm	Up to 8 marks
Coding	Working Python code	Up to 6 marks
Testing	Testing strategy and data	Up to 6 marks
Evaluation	Strenghts and weaknesses and suggested improvements	Up to 6 marks
		Total 40 marks

Part 3. Consolidated solution. 20%

All group members will get same marks for this part.

Patterns	Patterns identified	Up to 6 marks
Modelling	Cosolidated system diagrams	Up to 8 marks
Extension	Possibilities of extending consolidated solution	Up to 6 marks
		Total 20 marks

Part 4. Learning Log. 10%

Each member of the group will be have an individual mark. A maximum of 10 marks is attributed to this part.

Section 5: Feedback mechanisms

There will be opportunity for you to get formative interim feedbacks during teaching or practical sessions; Detailed summative written feedback will be provided via blackboard.

Appendices

APPENDIX A. Case Studies

You must select one of the following case studies and investigate it either by desktop research (i.e. online library resource, google search etc.) or meeting with stakeholders in a case study domain, for example UWE Chess Society for Case study 1; or you may visit 'Pepper' at the Bristol Robotics Lab (BRL) for Case Study 4 (entry to BRL is restricted. So you may need to contact your tutor to arrange a visit).

Case Study 1 - Chess Competition: A local organisation has expressed interest to host an U-20 regional Chess tournament. The tournament must follow the Swiss Pairing and conform with the FIDE regulations. The basic Swiss laws are as follows:
(a) a player must not be paired with any other player more than once; (b) players with equal scores must be paired if it is possible to do so; (c) if it is impossible to pair all players with equal scores, every player who is not paired with an opponent whose score is the same as his own must be paired with an opponent whose score is as close to his own as possible.
The Arbiter and Chief arbiter are involved in the process of pairing and conduct of the tournament. Using computational methods, propose an algorithm that would help the arbiter and chief arbiter decide the best pairing method(s) based on the number of registered participants for the regional tournament. The hosting organisation is also interested to find a solution for interested players to participate in the tournament through an online app and hence would appreciate a feasibility study by applying computational thinking.

Case Study 2 - Ensemble Mapping: The university's music society plans to organise an ensemble of two classical traditions – the Western Classical and the Carnatic Classical. Both the genres have similarities and subtle differences. You are a part of the organising committee and have the responsibility to submit a proposal that would satisfy a diverse audience. As a part of the proposal, the organising committee has asked you to propose a software that would help them achieve the objective. The software must be able to suggest instruments from both traditions that complement each other including suggestions for melodies to play in the ensemble that might satisfy a diverse audience.

Case Study 3 – Carbon Netural City: Bristol City council aims to be cabon neutral by 2030 by applying various interventions including reducing transport carbon footprint. They plan to encourage use of healthy activities (i.e. cycling, walking) and public transport for travelling, discourage diesel & petrol vehicles and promote electrical vehicles for private transport. They are looking for a technology solution that can measure current traffic trends i.e. public (buses) and private (cars) transport in the city and their carbon footprint on different days and times of the year. The system then should calculate expected impact (financial, social, health, environmental) to achieve the carbon neutral goal by 2030. Using the computational methods, you need to propose an algorithm that would help Bristol City Council to achieve this goal.

Cast Study 4 – Assisted Living: 'Pepper' is an English speaking robot that hopes to assist people at home. It could move about, recognise objects, remind what to do/ what not to do and when to do. For example, it could remind a person to take medicines at the right time. If the person takes the medicine either before or after the scheduled time, Pepper would appropriately prompt. Using computational thinking explore if Pepper could be a home tutor and the impact it could have on the wider society. As a part of the solution, you must also identify appropriate audience. You must consider the social, legal ethical and professional issues while proposing ways in which this might be achieved.

Case Study X: To promote creativity you may propose a new case study – subject to moderation and prior approval by your tutor.

APPENDIX B. Template for Portfolio

Selected Case Study:
Part 1: Problem Identification and Requirements

a) **Background, Context and Problems Identified: (up to 300 words)**
b) **Decomposition criteria: (up to 300 words)**
c) **Requirements analysis: (up to 300 words)**
d) **Commentary: (up to 200 words)**

Part 2: Designing Solution

a) **Student 1:**
   i) Patterns: (up to 200 words)
   ii) Modelling: (up to 250 words)
   iii) Design: (up to 250 words)
   iv) Coding: (code snippets) – actual source code files to be submitted with the portfolio
   v) Testing: (up to 200 words)
   vi) Evaluation: (up to 200 words)
b) **Student N:**
   i) Patterns: (up to 200 words)
   ii) Modelling: (up to 250 words)
   iii) Design: (up to 250 words)
   iv) Coding: (code snippets) – actual source code files to be submitted with the portfolio. You can either copy pate code in an Appendix of the portfolio. OR you can upload .py files with portfolio and write one sentence in the portfolio about the purpose of each .py file.
   v) Testing: (up to 200 words)
   vi) Evaluation: (up to 200 words)

Part 3: Consolidated Solution

a) **Patterns: (up to 200 words)**
b) **Modelling: (up to 250 words)**
c) **Extension: (up to 200 words)**

Part 4:

a) **Student 1:** (up to 500 words of text in table)

Week (from – to)	Learning	Group interaction	Stumbling Blocks or Issues	Solutions to Issues
Date from – Date To				
Date from – Date To				
Date from – Date To				
Date from – Date To				
Date from – Date To				
Date from – Date To				

Student 1's view of what mark for his/her learning logs should be awarded:

b) Student N: (up to 500 words of text in table)

Week (from – to)	Learning	Group interaction	Stumbling Blocks or Issues	Solutions to Issues
Date from – Date To				
Date from – Date To				
Date from – Date To				
Date from – Date To				
Date from – Date To				
Date from – Date To				

Student 1's view of what mark for his/her learning logs should be awarded:

Note that word limits are indicative and do not include any diagrams or images you may like to include in your portfolio.

APPENDIX C. Marking Criteria

Note that for all elements of assessment, the work must be well presented and well-written, properly spelt in grammatically correct English to achieve top marks.

Part 1: *Problem Identification and requirements*	**0-2**	**3-4**	**5-6**	**7-8**	**9-10**	**Mark & Advice for Improvement**
Background (8)	Little or no meanful background given	Problems identified and described but not put into a context	The context in which the problems were identified is explored and the problems clearly identified	Clear evidence of insightful thinking, exploring the context of the problems and their wider implications		
Decomposition (6)	No evidence of decomposition and no criteria explained	Limited decomposition and problems identified and criteria not fully explained	Criteria explained and correctly applied to identify problems, covering full scope of the case study			
Requirements (10)	No evidence of meaningful requirements gathering	Some evidence of meaningful requirements gathering with reference to techniques discussed in the course.	Good evidence of meaningful requirements gathering with reference to techniques discussed in the course	Good evidence of requirements gathering with reference to techniques and analysis performed	The process and results of requirements gathering and analysis are fully discussed	
Commentary (6)	Little or no reflection on the process	Some reflection on parts of the process.	Thoughtful reflection on all of the process, including improvements for the future.			
Part 2: *Problem Solution*	**0-2**	**3-4**	**5-6**	**7-8**	**9-10**	**Mark & Advice for Improvement**
Patterns (6)	No patterns or Patterns do not encompass the scope of the problem	Though fully scoped, significant characteristics or similarities are missing from patterns	Patterns are clearly presented with number of characteristics or similarities and examples are provided showing usefulness of patterns			
Modelling (8)	Model(s) do not encompass the scope of the problem	Though fully scoped, significant elements of the model(s) are missing	The system under consideration is modelled using a recognised notation/diagramming technique	The system is fully modelled from a number of perspectives, using a recognised notation/diagramming technique.		
Design (8)	The design is incomplete and/or wholly incorrect	Following the design (e.g. pseudocode, algorithm) will	The design (e.g. pseudocode, algorithm) will lead to the required results.	The design (e.g. pseudocode, algorithm) is wholly correct and complete		

		mostly lead to the required results		and presented to a professional standard		
Coding (6)	No code or incorrect code not mapping to design	Working code with some functionalties and partially mapped onto design	A fully functional code and fully mapped onto design			
Testing (6)	Testing is missing or only given minimal treatment	Some discussion of testing and test data is present but there are significant flaws.	A recognised testing strategy has been adopted and test data is chosen in accordance with the strategy and is more or less complete.			
Evaluation (6)	No reflection of very limited incoherent statement	Reflection with 2 strenghts and 2 weaknesses of the solution	Refelection with suggested improvements			
Part 3. Consolidated Solution	**0-2**	**3-4**	**5-6**	**7-8**	**9-10**	**Mark & Advice for Improvement**
Patterns (6)	No patterns or Patterns do not encompass the scope of the case study	Though fully scoped, significant characteristics or similarities are missing from patterns	Patterns are clearly presented with number of characteristics or similarities and examples are provided showing usefulness of patterns			
Modelling (8)	Model(s) do not encompass the scope of the case study	Though fully scoped, significant elements of the model(s) are missing	The system under consideration is modelled using a recognised notation/diagramming technique	The system is fully modelled from a number of perspectives, using a recognised notation/diagramming technique.		
Extension idea (6)	No plausible extension idea.	Extension idea leads to the design having limited additional application.	Idea significantly extends the applicability of the program design.			
Part 4. Learning log	**0-3**	**4-5**	**6-8**	**9-10**		**Mark & Advice for Improvement**
Learning log (10)	Significant missing elements or simply copies from slides or other course material.	No more than 1 missing element. There may be some copying from course materials but clear evidence of having thought about how the course topics can be applied.	No missing elements and no copying of course materials. Clear evidence of thought about how the course topics can be applied.	Nothing missing, no copying. Evidence of deep reflection and personal learning. Topics are discussed with reference to relevant sources in the course materials. Problems faced and solutions presented.		