Assignment Guidance and Front Sheet

This front sheet for assignments is designed to contain the brief, the submission instructions, and the actual student submission for any WMG assignment. As a result the sheet is completed by several people over time, and is therefore split up into sections explaining who completes what information and when. Yellow highlighted text indicates examples or further explanation of what is requested, and the highlight and instructions should be removed as you populate 'your' section.

This sheet is only to be used for components of assessment worth more than 3 CATS (e.g. for a 15 credit module, weighted more than 20%; or for a 10 credit module, weighted more than 30%).

To be completed by the student(s) prior to final submission:

Your actual submission should be written at the end of this cover sheet file, or attached with the cover sheet at the front if drafted in a separate file, program or application.

Student ID or IDs for group work	**e.g. 1234567**

To be completed (highlighted parts only) by the programme administration after approval and prior to issuing of the assessment; to be consulted by the student(s) so that you know how and when to submit:

Date set	09/12/2022
Submission date (excluding extensions)	10/02/2023
Submission guidance	Via Tabula
Marks return date (excluding extensions)	10/03/2023
Late submission policy	If work is submitted late, penalties will be applied at the rate of **5 marks per University working day** after the due date, up to a **maximum of 10 working days** late. After this period the mark for the work will be reduced to 0 (which is the maximum penalty). "Late" means **after the submission deadline time as well as the date** – work submitted after the given time even on the same day is counted as 1 day late. For **Postgraduate** students only, who started their **current course before 1 August 2019**, the daily penalty is **3 marks** rather than 5.
Resubmission policy	If you fail this assignment or module, please be aware that the University allows students to remedy such failure (within certain limits). Decisions to authorise such resubmissions are made by Exam Boards. Normally these will be issued at specific times of the year, depending on your programme of

	study. More information can be found from your programme office if you are concerned.

To be <u>completed</u> by the <u>module owner/tutor</u> prior to approval and issuing of the assessment; to be <u>consulted</u> by the <u>student(s)</u> so that you understand the assignment brief, its context within the module, and any specific criteria and advice from the tutor:

Module title & code	WM245 Programming Languages for Cyber Security
Module owner	HS Lallie
Module tutor	Dr Nikki Williams
Assessment type	Reflective Report
Weighting of mark	40%

Assessment brief
Please see below

Word count	Described below
Module learning outcomes (numbered)	1. Compare different programming paradigms used to create software. 2. Reflect on how software vulnerabilities can be minimised during software creation. 3. Incorporate security features in small-scale programs. 4. Develop small-scale programs that employ the idioms of a programming paradigm in a conventional manner.
Learning outcomes assessed in this assessment (numbered)	1. Compare different programming paradigms used to create software. 2. Reflect on how software vulnerabilities can be minimised during software creation.
Marking guidelines	Generally indicated within specification
Academic guidance resources	All queries to be directed to the tutor's email, with responses posted via moodle or workshop sessions.

1 Context

During this module we are exploring different programming languages, and how your decisions, for example in terms of language choice, paradigm choice, and library choice can affect the security of the resulting program. This need to make decisions with security as a key factor, is an added dimension compared to your goals and aims when first learning how to program.

2 The task

In this reflective report you need to discuss where you are in your **programming learning journey** (to provide context to the later discussions), discuss how you can bring **cyber security knowledge into your programming activities** and the types of mitigations you may put in place to develop more secure software, and use a **specific example** to show either how you could improve your code to better consider cyber security, or the specific design choices or functionality you included to reflect good cyber security. how a greater understanding of the security of your choices influences the code that your write. The final section should explore how you intend to **consider cyber security within your programming activities in the future**.

Your description should include a specific programming example you have made, allowing specific examples of code to be explained and explored. You may find it useful to include code snippets. Code snippets do not count towards the word count. You should explain what the code shows, as the marker may not know the language you have used. The code described does not have to have been written during this module.

There are a number of models or frameworks available to help your structure a reflective piece of writing, and some of these are covered in the module. You should follow one of these models when structuring the text. Reflective writing is almost always written in the first-person.

Elements to cover:

1. My programming journey
2. Programming to consider cyber security
3. Specific example of applying cyber security in my programming
4. How I will incorporate cyber security in my future programming activities

The **word count** for this assessment is **1,500 words** +/-10%.

3 Style and substance

Notes on style and substance:

Style represents the effort taken by the author to make things accessible to the reader; substance is the concept being expressed. Poor style will hide good substance. Good style cannot hide poor substance.
Style includes (but is not limited to):

- physical layout, typography, use of white space, pagination, headers and footers, consistency, colour,
- logical layout, structure via (sub-)sections, section numbering, section headings, organisation of sentences and paragraphs, sequencing of material,
- language, lucidity, economy (padding / waffle removal), appropriate assumptions,
- supporting material as necessary (diagrams, tables),
- conventions, referencing

Substance is the collection of cyber-security and programming concepts you are seeking to communicate to the reader. These must be correct, relevant and necessary to the problem being addressed.

4 Miscellaneous Important Constraints

a) All activity must be conducted legally and ethically.
b) Use conventional Harvard academic references.
c) You are advised to use diagrams wherever they help your explanation. Original sources of diagrams must be referenced where they are not of your own creation.
d) The overall word limit is 1,500 words +/-10%.
e) The text visible on the printed page must be consistent with the text accessible to pdf document text analysis tools. Inconsistency will be treated as academic misconduct.
f) Your name should not appear on any page.
g) This is an individual assignment.

Criteria	Weighting	Criteria				
		Excellent 1st (>70%)	**Very Good 2:1 (60-69%)**	**Good 2:2 (50-59%)**	**Satisfactory 3rd (40-49%)**	**Poor Minor Fail (0-39%)**
Programming knowledge - Correct use of key terminology and concepts within programming	35	Demonstrated a comprehensive knowledge and understanding of the subject and application to the topic.	Demonstrated an extensive knowledge and understanding of the subject and the application to the topic.	Demonstrated a good knowledge and understanding of the subject with some application to the topic.	Demonstrated a satisfactory knowledge and understanding of the subject with little application to the topic.	Below satisfactory attainment.
Cyber security knowledge - Correct use of key terminology and concepts within cyber security	35	Demonstrated a comprehensive knowledge and understanding of the subject with many security examples to illustrate points made.	Demonstrated an extensive knowledge and understanding of the subject with many security examples to illustrate points made.	Demonstrated a good knowledge and understanding of the subject with many security examples to illustrate points made.	Demonstrated a satisfactory knowledge and understanding of the subject with many security examples to illustrate points made.	Below satisfactory attainment.

Criteria	Weighting	Criteria				
		Excellent 1st (>70%)	**Very Good 2:1 (60-69%)**	**Good 2:2 (50-59%)**	**Satisfactory 3rd (40-49%)**	**Poor Minor Fail (0-39%)**
Quality of reflection (use of model and evidence of learning or changed actions)	30	Excellent application of the selected model to the reflections and a compelling narrative including key learning points. Excellent description of the lessons learned from the experience with detailed personal reflection and opportunities for further growth outlined.	A model is used and applied well to the reflective writing, conveying the key messages. Full description of the lessons learned from the experience with detailed personal reflection and opportunities for further growth outlined.	A model is used and applied broadly accurately to the reflection Critical description of the lessons learned from the experience with detailed personal reflection or opportunities for further growth outlined.	Reflection covers key elements but does not follow a formal model. Limited description of the lessons learned from the experience in the reflection.	Below satisfactory attainment.