University of South Wales
Prifysgol De Cymru

Faculty of Computing, Engineering and Science

Assessment Cover Sheet 2020-21

Module Code:	Module Title:	Module Team:
CS2S566	Tool Development for Computer Games	Gaius Mulley
Assessment Title:		**Assessment No.:**
Python/Pygame and Missile command		1
Date Set:	**Submission Date:**	**Return Date:**
28-Sep-2020 00:00	08-Jan-2021 23:55	05-Feb-2021 23:55

IT IS YOUR RESPONSIBILITY TO KEEP RECORDS OF ALL WORK SUBMITTED.

Marking and Assessment

This assignment will be marked out of **100%**.

This assignment contributes to **50%** of the total module marks.

Learning Outcomes to be assessed

As specified in the validated module descriptor https://icis.southwales.ac.uk

- 1) To identify the functional and non-functional requirements of a game engine / game design
- 2) Apply relevant software engineering techniques to develop applications to generate data for use in a game engine

Awarded mark is only provisional: subject to change and / or confirmation by the Assessment Board.

Assessment Task

Your task is to write an implementation of missile command suitable for running on a desktop computer using Python/Pygame and mouse as its primary input.

You should ignore gui based menus as these are covered next term.

Your task is to concentrate on making the game work using Pygame and Python. You should document the controls chosen and justify the design decisions. You must also provide a line by line commentary of all code you write. Finally you should give an analysis of the effectiveness of Python/Pygame when implementing this game.

Marking Scheme

	Fail (0/29)	Narrow Fail (30/39)	3rd Class / Pass (40/49)	Lower 2nd Class / Pass (50/59)	Upper 2nd Class / Merit (60/69)	1st Class / Distinction (70/100)
Analyse and discuss the effectiveness of Python/Pygame in producing such a game (30%)	☐ Very poor Analyse and discuss the effectiveness of Python/Pygame in producing such a game. The discussion barely scratched the surface of the topics	☐ Poor Analyse and discuss the effectiveness of Python/Pygame in producing such a game. It is likely that major topics were ignored or were highly marginalised	☐ Satisfactory Analyse and discuss the effectiveness of Python/Pygame in producing such a game. Both topics were addressed sensibly but no depth was given	☐ Good Analyse and discuss the effectiveness of Python/Pygame in producing such a game. Both topics were sensibly addressed with depth in one or more areas	☐ Very good Analyse and discuss the effectiveness of Python/Pygame in producing such a game. Topics were addressed with reference to your own work and cited sources	☐ Excellent Analyse and discuss the effectiveness of Python/Pygame in producing such a game. Topics were addressed with reference to your own work and cited sources. Independent thought shown
line by line commentary (20%)	☐ Very poor line by line commentary. Few lines of code are commented	☐ Poor line by line commentary. Many lines of code are not commented	☐ Satisfactory line by line commentary. Comments could be better formed, some lines might be missing comments	☐ Good line by line commentary. Majority of the comments well written	☐ Very good line by line commentary. Commentary is well written	☐ Excellent line by line commentary. Independent thought shown
use of PyGame libraries and general code (20%)	☐ Very poor use of PyGame libraries and general code. Code presented appears to show little time spent and little engagement with the topics	☐ Poor use of PyGame libraries and general code. Little understanding shown in the code and the code which exists looks like a cut n paste of lecture material without much thought	☐ Satisfactory use of PyGame libraries and general code. A good start and with more work the implementation could be made to work correctly	☐ Good use of PyGame libraries and general code. Mostly functioning - some bugs but fundamentally on the right track	☐ Very good use of PyGame libraries and general code. Fully functioning game produced, implementation does not deviate much from material given	☐ Excellent use of PyGame libraries and general code. Fully functioning game produced, independent thought shown
various inputs handled and controls chosen (30%)	☐ Very poor various inputs handled and controls chosen. Poor code and poor choice of controls	☐ Poor various inputs handled and controls chosen. Some obvious features might be missing or/and the code was poor	☐ Satisfactory various inputs handled and controls chosen. Quality of code handing the controls might be improved and the controls chosen are limited	☐ Good various inputs handled and controls chosen. Good choice of controls but the code could be improved. Alternatively excellent code but the choice of controls could be improved	☐ Very good various inputs handled and controls chosen. Very good choice of controls and the code which implements them is clean	☐ Excellent various inputs handled and controls chosen. Independent thought shown in addition to the previous box
Global:						