4COSC006C: Software Development I – Coursework specification (2020/21)	
Module leader	W Purdy
Weighting:	50%
Qualifying mark:	30%
Description:	Coursework
Learning Outcomes Covered in this Assignment:	The coursework rationale is: LO1 Analyse specific problems and design their solutions by applying appropriate algorithmic techniques; LO2 Apply programming concepts to implement solutions in the taught programming language; LO3 Implement and manipulate simple data structures; LO4 Use an integrated development environment to create programs to satisfy a simple specification.
Handed Out:	Tuesday 3rd November 2020
Due Date:	**Friday 11th December at 1:00 pm (LK Time)**
Expected deliverables:	a) Submit your Python program code - **Important:** Submit your python code file created in IDLE using the name convention: "student_id.py", e.g. w1234567.py - **DO NOT** submit your code as word, notepad or a PDF document. b) Submit your test case results c) Online Demo (during a scheduled tutorial)
Method of Submission:	Submitted online via Blackboard
Type of Feedback and Due Date:	Written feedback and marks 15 working days (3 weeks) after the submission deadline. All marks will remain provisional until formally agreed by an Assessment Board.

Assessment regulations
Refer to section 4 of the "How you study" guide for undergraduate students for a clarification of how you are assessed, penalties and late submissions, what constitutes plagiarism etc.

Penalty for Late Submission
If you submit your coursework late but within 24 hours or one working day of the specified deadline, 10 marks will be deducted from the final mark, as a penalty for late submission, except for work which obtains a mark in the range 40 – 49%, in which case the mark will be capped at the pass mark (40%). If you submit your coursework more than 24 hours or more than one working day after the specified deadline you will be given a mark of zero for the work in question unless a claim of Mitigating Circumstances has been submitted and accepted as valid.

It is recognised that on occasion, illness or a personal crisis can mean that you fail to submit a piece of work on time. In such cases you must inform the Campus Office in writing on a mitigating circumstances form, giving the reason for your late or non-submission. You must provide relevant documentary evidence with the form. This information will be reported to the relevant Assessment Board that will decide whether the mark of zero shall stand. For more detailed information regarding University Assessment Regulations, please refer to the following website: **http://www.westminster.ac.uk/study/current-students/resources/academic-regulations**

Coursework Description

The University requires a program to predict progression outcomes at the end of each academic year. You should write this program in Python using the data shown in Table 1.

Table 1: Progression outcomes as defined by the University regulations.

	Volume of Credit at Each Level			Progression Outcome
	Pass	Defer	Fail	
1	120	0	0	Progress
2	100	20	0	Progress (module trailer)
3	100	0	20	Progress (module trailer)
4	80	40	0	Do not Progress – module retriever
5	80	20	20	Do not Progress – module retriever
6	80	0	40	Do not Progress – module retriever
7	60	60	0	Do not progress – module retriever
8	60	40	20	Do not progress – module retriever
9	60	20	40	Do not progress – module retriever
10	60	0	60	Do not progress – module retriever
11	40	80	0	Do not progress – module retriever
12	40	60	20	Do not progress – module retriever
13	40	40	40	Do not progress – module retriever
14	40	20	60	Do not progress – module retriever
15	40	0	80	Exclude
16	20	100	0	Do not progress – module retriever
17	20	80	20	Do not progress – module retriever
18	20	60	40	Do not progress – module retriever
19	20	40	60	Do not progress – module retriever
20	20	20	80	Exclude
21	20	0	100	Exclude
22	0	120	0	Do not progress – module retriever
23	0	100	20	Do not progress – module retriever
24	0	80	40	Do not progress – module retriever
25	0	60	60	Do not progress – module retriever
26	0	40	80	Exclude
27	0	20	100	Exclude
28	0	0	120	Exclude

Part 1 - Student Version

1. The program should allow students to predict their progression outcome at the end of each academic year.
2. The program should prompt for the number of credits at pass, defer and fail and then display the appropriate progression outcome for an individual student (i.e., progress, trailing, module retriever or exclude).

Part 1 - An example of the program running with user input (shown in bold) :

```
------------------------------------------------------------
Please enter your credits at pass: 100
Please enter your credit at defer: 0
Please enter your credit at fail: 20
Progress (module trailer)
------------------------------------------------------------
```

- Marks will be allocated for the efficient use of conditional statements.
- Submit the completed part 1 test plan (in appendix) with your final part 1 solution.

Part 2 – Student Version (Validation)

Extend the Student Version to add the following validation.

1. The program should display '**Integer required**' if a credit input is the wrong data type.
2. The program should display '**Out of range**' if credits entered are not in the range 0, 20, 40, 60, 80, 100 and 120.
3. The program should display '**Total incorrect**' if the total of the pass, defer and fail credits is not 120.
4. The program could loop until acceptable inputs are entered. However, this is optional and will not be allocated a mark.

Part 2 - An example of the program running with user input (shown in bold):

```
------------------------------------------------------------
Please enter your credits at pass: p
Integer required

Please enter your credits at pass: 140
Out of range.

Please enter your credits at pass: 100
Please enter your credit at defer: 40
Please enter your credit at fail: 20
Total incorrect.

Please enter your credits at pass: 100
Please enter your credit at defer: 20
Please enter your credit at fail: 0
Progress (module trailer)
------------------------------------------------------------
```

- Use user-defined functions in your solution as appropriate.
- Submit the completed part 2 test plan (in appendix) with your final part 2 solution.

Part 3 - Staff Version with Histogram

This extension should meet the requirements specified for Part 1 and 2 but also allow a staff member to predict progression outcomes for multiple students.

1. The program should prompt for credits at pass, defer and fail and display the appropriate progression for each individual student until the staff member user enters 'q' to quit. Optionally you can use an input of **'y' to continue.**
2. When 'q' is entered, the program should produce a 'histogram' where each star represents a student who achieved a progress outcome in the category range: progress, trailing, module retriever and exclude. **The histogram should relate to the data input entered by the staff member during the program run and work for any number of outcomes**.
3. Display the number of students for each progression category and the total number of students.

Part 3 - Example of a program run and input (in bold). Note: program should exit on 'q' to quit. 'y' to continue shown in the example is optional and depends on your program structure.

```
------------------------------------------------------------
Staff Version with Histogram

Enter your total PASS credits: 120
Enter your total DEFER credits: 0
Enter your total FAIL credits: 0
Progress
```

```
Would you like to enter another set of data?
Enter 'y' for yes or 'q' to quit and view results: y

Enter your total PASS credits: 100
Enter your total DEFER credits: 0
Enter your total FAIL credits: 20
Progress (module trailer)

Would you like to enter another set of data?
Enter 'y' for yes or 'q' to quit and view results: y

Enter your total PASS credits: 80
Enter your total DEFER credits: 20
Enter your total FAIL credits: 20
Do not progress – module retriever

Would you like to enter another set of data?
Enter 'y' for yes or 'q' to quit and view results: y

Enter your total PASS credits: 60
Enter your total DEFER credits: 0
Enter your total FAIL credits: 60
Do not progress – module retriever

Would you like to enter another set of data?
Enter 'y' for yes or 'q' to quit and view results: y

Enter your total PASS credits: 40
Enter your total DEFER credits: 0
Enter your total FAIL credits: 80
Exclude

Would you like to enter another set of data?
Enter 'y' for yes or 'q' to quit and view results: q

------------------------------------------------------------
Horizontal Histogram
Progress 1  : *
Trailer 1   : *
Retriever 2 : **
Excluded 1  : *

5 outcomes in total.
------------------------------------------------------------
```

- The program will make use of loops and user-defined functions.
- Submit the completed part 3 test plan (in appendix) with your final part 3 solution.

Part 4 - Vertical Histogram (optional extension)

Extend your program to add a vertical histogram (stars in a category should go downwards), e.g.;

```
Progress  Trailing  Retriever  Excluded
   *        *         *          *
                      *
```

- Hint: as a line is printed decide if each category needs a star or space.
- If attempted, the code for **both** staff versions (Part 3 and Part 4) must be submitted for marking.
- Submit the completed part 4 test plan (in appendix) with your final part 4 solution.

Part 5 – Alternative Staff Version (optional extension)

- For this version the data will be obtained from a list, tuple or dictionary and NOT from user input. **Hint:** you could use a two-dimensional list (a list of lists) to hold the data.
- A histogram related to the data stored in the list, tuple or dictionary should be displayed.
- The data to use is shown in the table below.
- The solution should use user-defined functions.
- Submit the completed part 5 test plan (in appendix) with your final part 5 solution.

Data to use for Part 5
Pass = 120, Defer = 0, Fail = 0
Pass = 100, Defer = 20, Fail = 0
Pass = 100, Defer = 0, Fail = 20
Pass = 80, Defer = 20, Fail = 20
Pass = 60, Defer = 40, Fail = 20
Pass = 40, Defer = 40, Fail = 40
Pass = 20, Defer = 40, Fail = 60
Pass = 20, Defer = 20, Fail = 80
Pass = 20, Defer = 0, Fail = 100
Pass = 0, Defer = 0, Fail = 120

Part 5 - Example of a program run (no user input used).

```
Progress
Progress (module trailer)
Progress (module trailer)
Do not Progress - module retriever
Do not Progress - module retriever
Do not Progress - module retriever
Do not Progress - module retriever
Exclude
Exclude
Exclude

Progress  1: *
Trailing  2: **
Retriever 4: ****
Excluded  3: ***

10 outcomes in total.
```

References

- Reference any code taken from other sources in your program code.
- Include the following at the top of your program(s).

```
# I declare that my work contains no examples of misconduct, such as plagiarism, or collusion.
# Any code taken from other sources is referenced within my code solution.

# Student ID: ……

# Date:  ……
```

Coursework Demo

Demonstrate your working solution to your tutor during a scheduled online tutorial. **NOTE: If you do not attend your demo only your solutions for Part 1 and Part 2 will be marked.**

Marking scheme

The coursework will be marked based on the following marking criteria**:**

Criteria	**Max for Subcomponent**	**Max Demo ****	**Max Sub total**
Assignment - Progress Outcomes			
Part 1 - Student Version • Credits entered & progress outcome displayed (15) • Efficient use of conditional statements (5)	20		(20)
Part 1 Test Plan: 1-10 * **Demo ****	5	5	(30)
Part 2 - Student Version (Validation) • Catches input that is wrong data type • Credits outside range, 20, 40, 60, 80, 100, 120 • Credit total not 120	12		(42)
Part 2 Test Plan: 11-13 * **Demo ****	3	3	(48)
Part 3 - Staff Version with Histogram • Predict progression outcomes for multiple students. ○ User enters 'q' to quit ○ 'Histogram' correct ○ Category and overall totals correct	21		(69)
Part 3 Test Plan: 14-23 * **Demo ****	5	5	(79)
Part 4 - Vertical histogram • Implementation • Test plan: 24-25 * • Demo **	4	2	(85)
Part 5 - Alternative version • Implementation • Test Plan 26 * • Demo **	6	3	(94)
• User-defined functions	6		
Totals	(82)	(18)	(100)
Test plan**. PASS & matches submission * Demo**: ○ Marks allocated for your ability to answer questions and demonstrate understanding of your solutions. ○ If you are cannot explain you code and are unable to point to a reference within your code of where this code was found (i.e., in a textbook or on the internet) then no marks will be given for the demo of that component.			
NOTE: If you do not attend your online demo only Part 1 and Part 2 will be marked.			
Total:			

APPENDIX 1 – TEST PLAN for Part 1 & 2

Student Name: Gnei Zikra Amjah			**Student ID: 2019096**	
TEST PLAN for Part 1 Submit completed test plan with your code solution				
Test No.	Test Input	Expected Result	Actual Result (or state 'not attempted')	Pass / Fail ('Actual Result' matches 'Expected Result')
1	Pass = 120 Defer = 0 Fail = 0	'Progress' is displayed	Progress	Pass
2	Pass = 100 Defer = 20 Fail = 0	'Progress (module trailer)' is displayed	Progress (module trailer)	Pass
3	Pass = 100 Defer = 0 Fail = 20	'Progress (module trailer)' is displayed	Progress (module trailer)	Pass
4	Pass = 80 Defer = 20 Fail = 20	'Do not Progress – module retriever' is displayed	Do not progress – module retriever	Pass
5	Pass = 60 Defer = 40 Fail = 20	'Do not Progress – module retriever' is displayed	Do not progress – module retriever	Pass
6	Pass = 40 Defer = 40 Fail = 40	'Do not Progress – module retriever' is displayed	Do not progress – module retriever	Pass
7	Pass = 20 Defer = 40 Fail = 60	'Do not Progress – module retriever' is displayed	Do not progress – module retriever	Pass
8	Pass = 20 Defer = 20 Fail = 80	'Exclude' is displayed	Exclude	Pass
9	Pass = 20 Defer = 0 Fail = 100	'Exclude' is displayed	Exclude	Pass
10	Pass = 0 Defer = 0 Fail = 120	'Exclude' is displayed	Exclude	Pass
TEST PLAN for Part 2				
11	Pass = a	'Integer required' displayed	Integer required!	
12	Pass = 5	'Out of range' displayed	Out of Range	Pass
13	Pass = 100 Defer = 40 Fail = 0	'Total incorrect' displayed	Total incorrect	Pass

APPENDIX 2 – TEST PLAN for Part 3 & 4 - (Staff Version with Histogram)

TEST PLAN for Part 3 Submit this completed test plan with your solution code				
Test No.	Test Input	Expected Result	Actual Result (or state 'not attempted')	Pass / Fail
14	Pass = 120 Defer = 0 Fail = 0	'Progress' is displayed	Progress	Pass
15	Pass = 100 Defer = 20 Fail = 0	'Progress (module trailer)' is displayed	Progress (module trailer)	Pass
16	Pass = 100 Defer = 0 Fail = 20	'Progress (module trailer)' is displayed	Progress (module trailer)	Pass
17	Pass = 80 Defer = 20 Fail = 20	'Do not Progress – module retriever' is displayed	Do not progress – module retriever	Pass
18	Pass = 60 Defer = 20 Fail = 40	'Do not Progress – module retriever' is displayed	Do not progress – module retriever	Pass
19	Pass = 40 Defer = 80 Fail = 0	'Do not Progress – module retriever' is displayed	Do not progress – module retriever	Pass
20	Pass = 40 Defer = 40 Fail = 40	'Do not Progress – module retriever' is displayed	Do not progress – module retriever	Pass
21	Pass = 20 Defer = 20 Fail = 80	'Exclude' is displayed	Exclude	Pass
Displaying Histogram				
22	Enter 'q' to quit	Exits loop		
23	Exit loop	Progress 1 : * Trailer 2 : ** Retriever 4 : **** Excluded 1 : * 8 outcomes in total.	Same as the left. (Screenshot attached below.)	Pass
TEST PLAN for Part 4				
24	Use test data from Part 3 TEST plan 14-21			
25	Enter 'q' to quit / Exit loop	Progress 1 | Trailer 2 | Retriever 4 | Exclude 1 * * * * * * * *	Same as the left. (Screenshot attached below.)	Pass

		8 outcomes in total.		

```
Enter your credits at pass: a
Integer required!
Enter your credits at pass: 5
Out of Range
Enter your credits at pass: 100
Enter your credits at defer: 40
Enter your credits at fail: 0
Total incorrect

Horizontal Histogram
Progress  1  : *
Trailer   2  : **
Retriever 4  : ****
Exclude   1  : *
8  outcomes in total.

Vertical Histogram
Progress  1 | Trailer  2 | Retriever  4 | Exclude  1
        *       *              *              *
                *              *
                               *
                               *
8  outcomes in total.
```

APPENDIX 3 – TEST PLAN for Part 5

TEST PLAN for Part 5 Submit this completed test plan with your solution code				
Test No.	Test Input	Expected Result	Actual Result (or state 'not attempted')	Pass / Fail
26	Program is run	Progress Progress (module trailer) Progress (module trailer) Do not Progress - module retriever Do not Progress - module retriever Do not Progress - module retriever Do not Progress - module retriever Exclude Exclude Exclude Progress 1: * Trailing 2: ** Retriever 4: ** Excluded 3: *** 10 outcomes in total.	Progress Progress (module trailer) Progress (module trailer) Do not progress – module retriever Do not progress – module retriever Do not progress – module retriever Do not progress – module retriever Exclude Exclude Exclude Horizontal Histogram Progress 1 : * Trailer 2 : ** Retriever 4 : **** Exclude 3 : *** 10 outcomes in total.	Pass