

PYTHON EOM PROJECT CHALLENGE RUBRIC - 2020

					Possible Mark	Mark Obtained
	Understanding the problem as reflected by the design structure(s).					
Interface Design	8	6	4	2	8	
	User Interface using tkinter implemented correctly with all widgets aligned correctly.	One or two widgets are misaligned.	Half of the widgets are not correctly aligned.	Only a ¼ of the widgets used and aligned appropriately.		
	Appropriate variable/class/function naming conventions.					
	4	3	2	1	4	
	Variable naming/class/function follow specific convention and its uniform.	Variable/class/function naming is consistent but with a few not meaningful.	Half of the variables/functions/classes are not meaningful.	No conventions used. Variable names not meaningful.		
	Use of Selection Structures (IF, ELIF, Else)					
	8	6	4	2	8	
	Used appropriate and most effective selection	Used appropriate and most effective selection	Appropriate and most effective use of selection	Inappropriate or ineffective use of selection structures.		

	structures to solve the problem in all instances.	structures in most instances.	structures in less than 50% of the instances.			
	Use of iteration structures. (For and While loops)					
	8	6	4	2	8	
	Used appropriate and most effective repetition structures to solve the problem in all instances.	Used appropriate and most effective repetition structures in most instances.	Appropriate and most effective use of repetition structures in less than 50% of the instances.	Inappropriate or ineffective use of repetition structures.		
Program Solution	Use of list or stacks or arrays or dictionaries or queues.					
	8	6	4	2	8	
	Used more than one structure correctly.	Implemented only one structure correctly.	Attempt to use structure done but with some errors.	Limited to no attempt of using any of the above collections.		
	Use of Functions (parameter passing) including in-built functions					
	8	6	4	2	8	
	Excellent interaction/communication between modules/classes. Includes parameter passing.	Proficient/appropriate use of modules/functions or other sub-procedures with small flaws. Includes parameter passing between	Limited use of modules/functions or other sub-procedures. Limited parameter passing.	No use of modules/functions or other sub-procedures. No parameters passed.		

		modules/ functions or other sub-procedures but not always appropriate or correct.				
	File handling					
	4	3	2	1	4	
	Excellent use of text files and works perfectly. All required data get's printed to the output file.	Text files implemented but with few errors.	Attempted to use text files but with errors(Nothing gets written to the file.)	text file defined but not used.(Code has no write function(s)).		
	Software testing - doctest and or unittest					
	Any of docstring/unit-test implemented.(Tests should be passing)		Attempt to test the code but with errors. (Some/all tests are failing).	Only imported unittest/ doctest, but not test done.		
	4	2		1	4	
	Complexity of Code					
	8	6	4	2	8	
	Advanced. Other techniques such as advanced UI GUI implemented.	Skilled	Intermediate.	Basic.		
	Error/Exception Handling					
	8	6	4	2	8	

	Excellent exception handling by the use of selection structures and try, catch statements. Proper Feedback given to users.	Exceptions minimized by the use of selection structures and try, catch statements. No Feedback given to user.	Exception minimized by the use of selection structures only. No Feedback given to the user.	Attempt to implement exception resulted in logic or syntax errors.		
	Program Compiles Successfully					
	8	6	4	2	8	
	No run time or syntax errors. All the options are executed successfully	Program produce runtime errors only	Some of the options produce errors when executed	Majority of the code produces errors. No evidence of testing the application.		
	Output					
	8	6	4	2	8	
	Output correct Matches the user requirements.	Output is correct but not meeting user requirements in all paths.	Some sections produce incorrect output.	Incorrect logic. Logic not meeting user requirements.		
	Program Documentation					
	8	6	4	2	8	
	Code well documented in all paths	75% of code documented	Only 50%of code is documented	Just 20% of the code is documented		
General Evaluation	Time Management: (0 = always late, work was never done)					
	8	6	4	8		

	All deadlines met	Project submitted few hours after due date	Project submitted after interventions		
				100	

FINAL COMPETENCE LEVEL INDICATORS				
Outstanding	Highly competent	Competent	Not yet competent	Not achieved
5	4	3	2	1
Student's Competence Level:		PASS <input type="checkbox"/> FAIL <input type="checkbox"/>		
Student's Name and Signature:				
Lecturer's Signature:				
Date:				
Feedback to student-(Facilitators are expected to give students constructive feedback on positive areas and also areas which require more attention)				