Subject Outline

|  |  |
| --- | --- |
| **Subject Name** | Programming II |
| **Subject Code** | CP1404 |
| **Study Period** | SP53 |
| **Study Mode** | Internal |
| **Campus** | Singapore |
| **Subject Coordinator** | Lindsay Ward |
| **Subject Lecturers** | Dr. Shailey Chawla |

*We acknowledge the Traditional Owners of the lands and waters where our University is located and actively seek to contribute and support the JCU Reconciliation Statement, which exemplifies respect for Australian Aboriginal and Torres Strait cultures, heritage, knowledge and the valuing of justice and equity for all Australians.*

© Copyright 2018 – v1

This publication is copyright. Apart from any fair dealing for the purpose of private study, research, criticism, or review as permitted under the Copyright Act, no part may be reproduced by any process or placed in computer memory without written permission.

**Contents**

[1 Subject at a glance 3](#_Toc506879079)

[1.1 Subject Staff Contact Details 3](#_Toc506879080)

[1.2 Key dates 3](#_Toc506879081)

[2 Subject Calendar 4](#_Toc506879082)

[3 Subject Details 4](#_Toc506879083)

[3.1 Subject description 4](#_Toc506879084)

[3.2 Subject learning outcomes 4](#_Toc506879085)

[3.3 Learning and teaching in this subject 5](#_Toc506879086)

[3.4 Subject design - Progressive feedback on student learning 6](#_Toc506879087)

[3.5 Subject textbook and resources 6](#_Toc506879088)

[3.6 Prerequisite subjects 6](#_Toc506879089)

[4 Assessment Information 7](#_Toc506879090)

[4.1 Assessment Tasks 7](#_Toc506879091)

[4.2 Submission of assessment 9](#_Toc506879092)

[4.3 Return of assessment 10](#_Toc506879093)

[5 Important Information for Successful Completion of this Subject 10](#_Toc506879094)

[5.1 Requirements for successful completion of subject 10](#_Toc506879095)

[5.2 Student participation requirements 10](#_Toc506879096)

[5.3 Plagiarism 11](#_Toc506879097)

[5.4 Referencing 11](#_Toc506879098)

[5.5 Final Examination 11](#_Toc506879099)

[5.6 Student feedback on subject 11](#_Toc506879100)

[5.7 Inherent requirements 12](#_Toc506879101)

[5.8 Student Support 12](#_Toc506879102)

[6 Rubrics 14](#_Toc506879103)

# Subject at a glance

## Subject Staff Contact Details

| **Teaching Team** | **Staff Member** | **Room** | **Phone** | **Email** |
| --- | --- | --- | --- | --- |
| Subject Coordinator | Lindsay Ward | 17-048 Townsville | 4781 4619 | [lindsay.ward@jcu.edu.au](mailto:lindsay.ward@jcu.edu.au) |
| Lecturer | Shailey Chawla | C3-06 | 67093717 | [shailey.chawla@jcu.edu.au](mailto:shailey.chawla@jcu.edu.au) |

## 

## Key dates

| **Key dates** | **Date** |
| --- | --- |
| Census date | See [2018 Study Period and Census Dates](https://www.jcu.edu.au/students/important-dates) |
| Last date to withdraw without academic penalty | See [2018 Study Period and Census Dates](https://www.jcu.edu.au/students/important-dates) |
| Assessment Task 1: Assignment 1, 20% | **Due 09/12/2018** |
| Assessment Task 2: Assignment 2, 20% | **Due 06/01/2019** |
| Assessment Task 3: Practicals, 30% | Due each week during practical time |
| Assessment Task 4: Examination, 30% | During examination period |

# Subject Calendar

Please note, the sequence of some topics may change due to staff availability, resourcing, or due to unforeseen circumstances.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Week** | **Date** | **Lecture** | **Book** | **Practical** | **Related Assessment** |
| 1 |  | Intro, Python Basics & Control | 1, 2 | 1. PyCharm, basics, selection, loops | *(Note: all practicals are assessed)* |
| 2 |  | Working with Strings,  Files and Exceptions I | 4, 6 | 2. Files, string formatting | Assignments, Exam |
| 3 |  | Functions, Version Control  (Git & GitHub), Debugging | 5 | 3. Functions, Git & GitHub | Assignments, Exam |
| 4 |  | Lists and Tuples, More on Functions | 7, 8 | 4. Lists, tuples | Assignments, Exam |
| 5 |  | Dictionaries and Sets, Code Reviews Using GitHub Pull Requests | 9 | 5. Dictionaries, Code Reviews with PRs | Assignments, Exam |
| 6 |  | Intro to Classes | 11 | 6. Classes | Assignment 2, Exam |
| 7 |  | Kivy |  | 7. Kivy | Assignment 2 |
| 8 |  | More on Classes (Inheritance) | 12 | 8. Inheritance | Assignment 2, Exam |
| 9 |  | Files and Exceptions II | 14 | 9. OS - renaming/moving files | Exam |
| 10 |  | Testing\*, Recursion | 15 | 10. Testing, Recursion and Fun with Flask | Exam |

\*Testing is covered in chapter 15 of the 2nd edition of the textbook but is not in the 3rd edition. It is available as a PDF on LearnJCU.

# Subject Details

## Subject description

In this subject, students are introduced to fundamental principles and practices of computer programming. Students learn and use problem-solving and software development techniques including planning, prototyping, version control, testing and debugging. Students will develop procedural and object-oriented programs with console and graphical user interfaces.

## Subject learning outcomes

Students who successfully complete this subject will be able to:

* apply fundamental application development skills in a modern programming environment;
* develop and utilise best-practice coding techniques;
* adopt self-learning with modern support resources.

These outcomes will contribute to your overall achievement of **course learning outcomes.** Your course learning outcomes can be located in the entry for your course in the electronic JCU [Course Handbook 2018](https://www.jcu.edu.au/course-and-subject-handbook)(see*Academic Requirements for Course Completion*).

## Learning and teaching in this subject

This subject is designed to develop your programming skills step-by-step using a variety of teaching and learning approaches. You should understand that the best way to learn to program is to program! The more time you spend actually planning and writing programs, the better. The lectures teach specific programming skills and patterns (often with recommended ways to do common tasks) in an interactive manner. Here is one student's comment about the value of attending lectures:

* *"The in class examples were the best part of this class. It is one thing to read about it in the book, but working out the problems with <the lecturer> was very fun and helped me learn the various programming techniques much quicker."*

You will then use and develop the concepts and skills in the practical sessions. Practical sessions are scheduled for 2 hours, but if you find it takes you longer to complete them, that's OK - just start them before coming to the practical session, as the following student comment shows:

* *"The practicals were very long, it was hard to finish in the allocated time. Making it a necessesity to start the practical before entering. Although this isn't a huge problem, it means that it is hard to manage the time efficiently."*

Aim for a minimum of 2 hours per week practising your skills outside class time. It's like learning to play guitar or basketball - you can't expect to get really good if you just go to your lessons or training and do nothing in between - you also need to spend time practising on your own. Keep working at each skill until you really understand it and can use it comfortably. For example, when you learn about using lists to store sequential data, write as many different (but probably similar) programs as you can that use lists. We have a great collection of practice projects to help you with this at: <https://github.com/CP1404/Starter/wiki/Practice-Programming-Projects>

When you get stuck or need to do something you're unsure of, it is very tempting to just "Google it", but you will often find misleading and unhelpful material online and we suggest you start first with the material in the subject (including lecture notes, practicals, and the programming patterns summary at <https://github.com/CP1404/Starter/wiki/Programming-Patterns>). We are very confident that you will find exactly what you need to know in the subject resources because that's how we've designed the subject - it's all here. The assignments are built based on the concepts and skills taught in lectures and practicals, so anything you need to know for the assignment, you can find here. If you find yourself struggling with the assignment, go back to the lectures and practicals that relate to your area of concern and work on that area until you can comfortably apply it in your assignment. OK?

Here are some actual comments from CP1404 students, written as part of their reflection on the second assignment:

* *Instead searching the internet for solutions i found it a lot easier to run programs from the kivy demos to find examples of the answers i was looking for.*
* *I would dedicate more time to following lectures and practicals which I have missed a lot of this semester and I am feeling the effects now.*
* *there were times I had problems (which I wasn't sure how to solve). After analysing the problem, finding appropriate resources (e.g. lecture notes), and a bit of iterative development, I was able to overcome these problems*

You are expected to be an **active participant in the learning process** and are encouraged to participate in the lectures and practicals and undertake weekly chapter readings prior to the lectures.

|  |  |  |  |
| --- | --- | --- | --- |
| **Key subject activities** | **Time** | **Day & Date** | **Room/Location** |
| Lectures (2 hours per week) | Refer to [JCU Timetable 2018](https://timetable.jcu.edu.au/2018/Login.aspx) | | |
| Practicals (3 hours per week) | Refer to [JCU Timetable 2018](https://timetable.jcu.edu.au/2018/Login.aspx) | | |

For information regarding class registration, visit the [Class Registration Schedule](https://www.jcu.edu.au/students/enrolment/class-registration-schedule).

## Subject design - Progressive feedback on student learning

You will be given feedback as appropriate each week during practicals, and through the formal assessment results. You should also consider that your weekly preparation and practice will provide a good guide to how well you are progressing with the subject. You are welcome to ask teaching staff for further feedback as appropriate.

## Subject textbook and resources

Prescribed Textbook

* **The Practice of Computing Using Python** 3rd Edition by Punch & Enbody. Pearson, 2017.   
  ISBN-13: 978-0134379760 (or the 2nd edition is fine)

Software

* All the software used in this subject is free or freely available for JCU IT students. Instructions for getting setup on your own computer can be found at: <https://github.com/CP1404/Starter/wiki/Software-Setup>   
  Please complete this setup as soon as you can.

## Prerequisite subjects

CP1801 OR CP1401 OR CP1200 OR EG1002 OR CP2200

For information on prerequisites for this subject please enter the subject code “CP1404” into the JCU Studyfinder: <https://secure.jcu.edu.au/app/studyfinder/>

# Assessment Information

## Assessment Tasks

**ASSESSMENT TASK 1: Assignment 1**

|  |  |
| --- | --- |
| **Aligned subject learning outcomes** | * apply fundamental application development skills in a modern programming environment * develop and utilise best-practice coding techniques |
| **Group or individual** | Individual |
| **Weighting** | 20% |
| **Due date** | **09/12/2018, 11 pm** |

**ASSESSMENT TASK 1: DESCRIPTION**

You will be given two programming assignments in this subject. These will involve planning and implementing Python programs for a small-to-medium-sized application. Assessment will be based on the quality of your algorithms (pseudocode), documentation and solutions. It is important that you don’t just try and get a program to work, but that you develop a good program systematically, starting with the planning process and then writing good clean code.

**Assignment 1** will help you build skills using techniques like selection, repetition, file input/output, exceptions, lists, tuples, functions and string formatting.

**ASSESSMENT TASK 1: CRITERIA SHEET**

See Rubrics in Section 6.

**ASSESSMENT TASK 2: Assignment 2**

|  |  |
| --- | --- |
| **Aligned subject learning outcomes** | * apply fundamental application development skills in a modern programming environment * develop and utilise best-practice coding techniques |
| **Group or individual** | Individual |
| **Weighting** | 20% |
| **Due date** | **06/01/2018,11pm** |

**ASSESSMENT TASK 2: DESCRIPTION**

**Assignment 2** will build on the assignment 1 program with more advanced code constructs such as dictionaries, classes and a Graphical User Interface (GUI) using Kivy.

**ASSESSMENT TASK 2: CRITERIA SHEET**

See Rubrics in Section 6.

**ASSESSMENT TASK 3: Practicals**

|  |  |
| --- | --- |
| **Aligned subject learning outcomes** | * apply fundamental application development skills in a modern programming environment * develop and utilise best-practice coding techniques * adopt self-learning with modern support resources |
| **Group or individual** | Individual |
| **Weighting** | 30% |
| **Due date** | each week during practical time |

**ASSESSMENT TASK 3: DESCRIPTION**

There will be 10 assessed practicals, each split into multiple parts. Part 1 is usually a walkthrough or a warm up that should not take long but shows you working code to learn from and extend; Part 2 is usually intermediate exercises, “fill-in-the-blanks” or similar style where you do a small task, mostly modifying or extending existing code; and Part 3 is “do-from-scratch” exercises where you do the whole task with the benefit of having completed the earlier parts. In each practical, there is an additional “extension” section, which we strongly recommend you to do to practise and extend your programming skills. This is so important!

You should aim to complete this work in the practical session, but you have one additional week to complete each of these tasks and still earn your mark (except for the final week, which must be completed to a satisfactory level by the end of the session). If you do not finish a practical in the scheduled week, you must show it to your practical supervisor at the **start** of the next practical to be given more marks. If you find yourself not completing the work most weeks, then please start the work before your normal practical session and aim to finish on time.

You also have **10 online tests** via LearnJCU, which are marked as part of these practicals. They are specifically designed to help you assess your own learning and get feedback on how you are going with the subject. They must be completed by their due dates, usually the end of the week after the related content is introduced.

The total mark for practicals will be weighted as ¼ for tests, ¾ for practicals.

**ASSESSMENT TASK 3: CRITERIA SHEET**

The 10 numbered practicals in the subject calendar below are the ones that will be marked. The "Assignment Work" practicals are an excellent opportunity for you to get expert help with your assignment but are not marked. Practical assessment will be based on attempting the tasks up to but not including the extension section to a satisfactory standard (not necessarily getting everything correct). You will be marked as follows:

0 – not attempted at all  
 1 – some of the work attempted with minimal effort  
 2 – some of the work attempted with reasonable effort  
 3 – most or all of the work successfully completed

Online tests are an effective way for you to revise the material you have been learning, and to evaluate how well you are keeping up with things. Because they are mostly for keeping you on track, the specific test scores are not as important as whether you do them or not, so they are marked as follows:

0 – not attempted or a test result of zero  
 1 – test result of more than zero

**ASSESSMENT TASK 4: EXAMINATION**

|  |  |
| --- | --- |
| **Aligned subject learning outcomes** | * develop and utilise best-practice coding techniques |
| **Weighting** | 30% |
| **Date** | During examination period |
| **Duration** | 2.5 hours |
| See [Special Consideration, Supplementary, Deferred and Special Examinations Policy](https://www.jcu.edu.au/policy/student-services/special-consideration,-supplementary,-deferred-and-special-examinations-policy) | |

**ASSESSMENT TASK 4: EXAMINATION DESCRIPTION**

You will complete a final examination during the University’s exam period. The examination is the final opportunity to show your understanding of important concepts as well as demonstrate specific programming skills. It is important to note that while you may do most of your programming on a computer, with the help of a development environment, the exam will test your abilities on paper, without the computer’s help. You need to build your understanding and experience deeply to be able to read and write code on paper. This is a valuable skill and something often covered in IT job interviews.

The exam will use both multiple-choice questions and short-answer written questions to test your knowledge and your ability to apply that knowledge. Questions will include reading and writing code. The exam will cover material from the entire subject (unless otherwise stated in class).

**ASSESSMENT TASK 4: CRITERIA SHEET**

Marks are given for correct answers, and will not be deducted for incorrect answers (i.e. you will not be penalised for an incorrect answer, you will simply not receive the marks for that question). Answers to short-answer questions can earn part marks for partially correct answers.

## Submission of assessment

The ability to adhere to deadlines is a highly desirable attribute that employers seek in our graduates.   
Right from the beginning, new students should acquire the habit of meeting deadlines for their work, by organising their study time appropriately. The following points apply to the submission of assessment:

1. Assessment must be submitted to the Assignment Drop Box on the LearnJCU subject site (click on Assessment in the subject site menu).
2. You must contact your lecturer well before the due date if you are likely to require an extension.
3. Extensions will be granted in cases of illness or personal issues (supported by strong evidence – at least medical certificate/counsellor’s statement required). It is at the discretion of the subject coordinator/lecturer that extensions will be granted for inescapable, unexpected, documented work commitments (provide documentation).
4. Where no prior extension has been approved, late submissions will incur a penalty of 5% of the total mark available per day including part-days, weekends and public holidays. Assessment tasks will generally not be graded after 14 days past the due date.

Note that the [Learning, Teaching and Assessment Policy](https://www.jcu.edu.au/policy/learning-and-teaching/learning-teaching-and-assessment-policy) (5.22.3) outlines a uniform formula of penalties that will be imposed for submission of an assessment task after the due date. **This formula is 5% of the total possible marks for the assessment item per day including part-days, weekends and public holidays**. After 20 days, the assessment item thus would be awarded 0 marks (i.e. 5% x 20 = 100% of total possible marks in penalties).

## Return of assessment

The marked assessment and feedback will be available online through LearnJCU no later than 21 days after the due date (click on My Grades in the subject site menu).

# Important Information for Successful Completion of this Subject

## Requirements for successful completion of subject

In order to pass this subject, you must:

* Achieve an overall percentage of 50% or more;
* Submit a credible attempt at all assessment items within this subject. Students who have completed less than 100% of the assessment will be subject to review by the College Assessment Committee which could result in an ‘X’ grade (*Fail (did not sit for exam/s or did not complete at least 80% of assessment requirements or deferred exam not granted)).*
* Demonstrate regular attendance and engagement with the content of this subject in accordance with student participation requirements as outlined in 5.2, including but not limited to any mandatory face to face attendance or online session participatory attendance.

Assessment items and final grades will be reviewed through moderation processes ([Learning, Teaching and Assessment Policy](https://www.jcu.edu.au/policy/learning-and-teaching/learning-teaching-and-assessment-policy), 5.13-5.18). It is important to be aware that assessment “is always subject to final ratification following the examination period and that no single result represents a final grade in a subject” (Learning, Teaching and Assessment Policy, 5.22).

## Student participation requirements

The JCU [Learning, Teaching and Assessment Policy](https://www.jcu.edu.au/policy/learning-and-teaching/learning-teaching-and-assessment-policy) (4.3) indicates that, “a **3 credit point subject** will require a **130 hour work load** of study-related participation including class attendance over the duration of the study period, **irrespective of mode of delivery”**. This work load comprises **timetabled hours** and **other attendance requirements**, as well as **personal study hours,** including completion of online learning activities and assessment requirements.

Note that “**attendance** at specified classes will be a mandatory requirement for satisfactory completion of some subjects” (Learning, Teaching and Assessment Policy, 5.10); and that additional hours may be required per week for those students in need of **English language, numeracy** or **other learning support.**

**Mandatory Attendance Requirements**

Minimum attendance requirements at practicals are a mandatory component of this subject. To be eligible to sit the final exam or submit the final assessment task you must attend a minimum 70% of all practicals. Therefore, if your subject has 12 practicals you must attend a minimum 8 practicals. Attendance will be taken at all practicals. If your practicals attendance is deemed lacking you will be required to meet with the Associate Dean, Learning and Teaching to show cause.

If this is an external study subject (EXT) this refers to study that does not require on-campus attendance but does require online attendance and participation. All other rules, guidelines and expectations apply. Assessment due dates, learner responsibility in terms of participation and engagement and independent learning skills are necessary.

Find out more at the JCU Off-Campus Student Information website, Getting Started: <https://www.jcu.edu.au/off-campus-students>

## Plagiarism

Plagiarism occurs when writers claim ownership of written words or ideas that are not their own. Plagiarism is a **form of cheating** and any instances of plagiarism will be dealt with promptly according to University procedures.

Please see the **JCU Student Academic Misconduct Requirements Policy**

* [www.jcu.edu.au/policy/student-services/student-academic-misconduct-requirements-policy](http://www.jcu.edu.au/policy/student-services/student-academic-misconduct-requirements-policy)
* Also see the definition of self plagiarism in the **JCU Learning, Teaching and Assessment Policy** and note policy statement 5.9 in regard to submission of one’s own work that has been previously submitted for assessment and received a grade [www.jcu.edu.au/policy/learning-and-teaching/learning-teaching-and-assessment-policy](http://www.jcu.edu.au/policy/learning-and-teaching/learning-teaching-and-assessment-policy)
* What is Plagiarism? [www.jcu.edu.au/students/exams-and-results/what-is-plagiarism](http://www.jcu.edu.au/students/exams-and-results/what-is-plagiarism)
* Further information can be found via **Help and Support** in your LearnJCU subject site menu under **Plagiarism (Academic Misconduct)**.

## Referencing

Referencing is a systematic way of acknowledging the sources that you have used. Students should check out the very helpful online resources relating to academic writing and referencing at:

* Writing and Maths Skills Online [www.jcu.edu.au/students/learning-skills/learning-skills-online](http://www.jcu.edu.au/students/learning-skills/learning-skills-online) (In particular, the booklet [Summarising, Paraphrasing & Avoiding Plagiarism](http://www.jcu.edu.au/office/tld/writingskills/documents/SP&Pac030406.pdf) is a very useful guide).
* The Referencing Libguide <http://libguides.jcu.edu.au/referencing>
* Further information can be found via **Help and Support** in your LearnJCU subject site menu under **Academic Skills Support**.

## Final Examination

For face to face final exams the location will be published via the Students Online area of the JCU website. Visiting this area of the website will provide you with an access path to your personal examination timetable.

For external study students, JCU has arranged external exam centres around the world. Your exam centre will be assigned to you based on your Semester Residential Address in eStudent and you will be sent a letter advising the centre details. Therefore, it is vitally important that you have the correct address details lodged on eStudent.

If you are an external student living in proximity to Cairns or Townsville, your exam location will be published via the Students Online area of the JCU website. Visiting this area of the website will provide you with an access path to your personal examination timetable.

Please note that a draft exam timetable is published 7 weeks prior to the scheduled commencement date of the exam period. The final exam timetable is published 5 weeks before the commencement of the exam period.

The examination period for SP53 2018 has been scheduled in the Important Dates calendar: <https://www.jcu.edu.au/students/important-dates>

## Student feedback on subject

As part of our commitment at JCU to improving the quality of our courses and teaching, we regularly seek feedback on your learning experiences. Student feedback informs evaluation of subject and teaching strengths and areas that may need refinement or change. ***YourJCU Subject and Teaching Surveys*** provide a formal and confidential method for you to provide feedback about your subjects and the staff members teaching within them. These surveys are available to all students through [LearnJCU](https://learnjcu.jcu.edu.au/webapps/portal/frameset.jsp). You will receive an email invitation when the survey opens. We value your feedback and ask that you provide constructive feedback about your learning experiences for each of your subjects, in accordance with responsibilities outlined in the [Student Charter](https://www.jcu.edu.au/students/support/student-charter). Refrain from providing personal feedback on topics that do not affect your learning experiences. Malicious comments about staff are deemed unacceptable by the University.

## Inherent requirements

[Inherent requirements](https://www.jcu.edu.au/learning-and-teaching/resources/inherent-requirements) are the fundamental abilities, attributes, skills and behaviours needed to achieve the learning outcomes of a course while preserving the academic integrity of the university’s learning, assessment and accreditation processes. Students and prospective students must be able to demonstrate that they have acquired or have the ability to acquire the inherent requirements for their degree.

Reasonable adjustments may be made to assist students manage additional circumstances impacting on their studies provided these do not change the academic integrity of a degree. Reasonable adjustments do not alter the need to be able to demonstrate the inherent requirements of the course. Students who believe they will experience challenges completing their degree or course because of their disability, health condition or other reason should discuss their concerns with an AccessAbility Services team member or a member of College staff, such as the Course Coordinator. In the case where it is determined that inherent requirements cannot be met with reasonable adjustments, the University staff can provide guidance regarding other study options.

## Student Support

**Business Online** responds to requests for assistance with all learning technology and associated issues. Support is provided for LearnJCU, Pebble Pad, Blackboard Collaborate, recording presentations, Respondus LockDown Browser and online exams, and other learning technologies. Help and Support is available 7 days a week, 8.30am - 8.00pm weekdays, emails checked by 3.00pm weekends. Contact: [businessonline@jcu.edu.au](mailto:businessonline@jcu.edu.au)

For more student support services and related information look for [Student Support Services](https://learnjcu.jcu.edu.au/webapps/blackboard/content/listContent.jsp?course_id=_72928_1&content_id=_2183819_1) in the **Help and Support** menu item on all CBLG LearnJCU subject sites.

Please see the Student Resources tab in the top menu of the LearnJCU for other important student information pertaining to plagiarism and referencing, examinations advice and student support services.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **FOR OFFICE USE**  This subject outline has been prepared by Lindsay Ward for the College of Business, Law and Governance, Division of Tropical Environments & Societies, James Cook University. Updated 16/02/2018.   |  |  |  | | --- | --- | --- | | Q1. This subject is offered across more than one campus and/or mode and/or teaching period within the one calendar year. | Yes | No | | Q2. If Yes (Q1), the design of all offerings of this subject ensure the same learning outcomes and assessment types and weightings. | Yes | No | |  | | |   Subject Outline Peer Reviewer   |  |  | | --- | --- | | **Name** | **Dr Jason Holdsworth** | | **Position** | **IT Lecturer** | | **Date Reviewed** | **13/02/2018** | |

# Rubrics

Sample assessment criteria sheets (rubrics) for assignments are provided here.

**Assignment 1**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Criteria** | **Exemplary (9, 10)** | **Good (7, 8)** | **Satisfactory (5, 6)** | **Limited (2, 3, 4)** | **Very Limited (0, 1)** |
| **Planning  Pseudocode for algorithms** | Clear, well-formatted, consistent and accurate pseudocode that completely and correctly solves the problem, for two functions. | Exhibits aspects of exemplary (left) and satisfactory (right) | Some but not many problems (e.g. an incomplete solution, inconsistent use of terms, inaccurate formatting, not for two functions). | Exhibits aspects of satisfactory (left) and very limited (right) | Very many problems or pseudocode not done. |
| **Program Execution**  **Correctness** *Worth double* | Program works correctly for all functionality required. | Program mostly works correctly for most functionality, but there is/are some required aspects missing or that have problems. | Program works incorrectly for all functionality required. |
| **Error checking** | Invalid inputs are handled well using exceptions and control logic as instructed, for all user inputs. | Invalid inputs are mostly handled correctly as instructed, but there is/are some problem(s), e.g. exceptions not well used. | Error checking is not done or is very poorly attempted. |
| **Similarity to sample output (including all formatting)** | All outputs match sample output perfectly, or only one minor difference, e.g. wording, spacing. | Multiple differences (e.g. typos, spacing, formatting) in program output compared to sample output. | No reasonable attempt made to match sample output. Very many differences. |
| **Quality of Code Identifier naming** | All function, variable and constant names are appropriate, meaningful and consistent. | Multiple function, variable or constant names are not appropriate, meaningful or consistent. | Many function, variable or constant names are not appropriate, meaningful or consistent. |
| **Use of code constructs** | Appropriate and efficient code use, including good logical choices for data structures and loops, good use of constants, etc. | Mostly appropriate code use but with definite problems, e.g. unnecessary code, poor choice of data structures or loops, no use of constants. | Many significant problems with code use. |
| **Use of functions** | Functions and parameters are appropriately used, functions reused to avoid code duplication. | Functions used but not well, e.g. incorrect/missing parameters or calls, unnecessary duplication or main code outside main function. | No functions used or functions used very poorly. |
| **Formatting** | All formatting is appropriate, including correct indentation, horizontal spacing and consistent vertical line spacing. PyCharm shows no formatting warnings. | Multiple problems with formatting reduces readability of code. PyCharm shows formatting warnings. | Readability is poor due to formatting problems. PyCharm shows many formatting warnings. |
| **Commenting** | Helpful block/inline comments and meaningful docstrings for all functions, top docstring contains all program details (name, date, basic description, GitHub URL). | Comments contain some noise (too many/unhelpful comments) or some missing program details in top docstring or some inappropriate or missing block/inline comments. | Commenting is very poor either through having too many comments (noise) or too few comments. |
| **Use of version control** | Git/GitHub used effectively and the repository contains a good number of commits with good messages that demonstrate incremental code development. | Aspects of the use of version control are poor, e.g. not many commits, or meaningless messages that don’t represent valuable incremental development. | Git/GitHub not used at all. |

**Assignment 2**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Criteria** | **Exemplary (9, 10)** | **Good (7, 8)** | **Satisfactory (5, 6)** | **Limited (2, 3, 4)** | **Very Limited (0, 1)** |
| **Thoughtful and useful project reflection**  ***Worth double*** | The project reflection is complete and describes development and learning well, shows careful thought, highlights insights made during code development. | Exhibits aspects of exemplary (left) and satisfactory (right) | Project reflection contains some good content but is insufficient in coverage, depth or insight. | Exhibits aspects of satisfactory (left) and very limited (right) | Many aspects of the project reflection are missing or could be improved. |
| **Use of version control** | Git/GitHub has been used effectively and the repository contains a good number of commits with good messages that demonstrate incremental code development **starting with classes and testing then console before GUI**. | Git/GitHub used but several aspects of the use of version control are poor, e.g. not enough commits, or meaningless messages that don’t represent valuable incremental development in an appropriate order. | Git/GitHub not used. |
| **Console program** | Book class is used correctly in console program. | Book class is used in console program but not correctly. | Book class is not used in console program. |
| **GUI layout** | GUI layout is well constructed and contains all the required widgets. | Multiple aspects of the GUI layout are incomplete or poorly done. | GUI layout is very poor or not done. |
| **Error handling** | Errors are handled correctly and robustly as required. | Some errors are handled but not all, or errors are not handled properly. | No reasonable error handling. |
| **Correctness**  ***Worth double*** | GUI program works correctly for all other functionality required. | There are some significant problems with other functionality required. | Program works incorrectly for all other functionality required. |
| **Identifier naming** | All function, variable and constant names are appropriate, meaningful and consistent. | Several function, variable or constant names are not appropriate, meaningful or consistent. | Many function, variable or constant names are not appropriate, meaningful or consistent. |
| **Use of code constructs** | Appropriate and efficient code use, including no unnecessary duplication, good logical choices for control and storage, good use of constants, no global variables etc. | Several problems, e.g. unnecessary duplication, poor control, no use of constants, improper use of global variables. | Many problems with code use. |
| **Use of methods in main Kivy app** | Methods and parameters in main app are appropriately used with good design including good reuse. | Some problems with methods, e.g. unsuitable parameters or calls, global code, or not designing methods for reuse. | No methods used or methods used very poorly. |
| **Use of classes and methods in Book and BookCollection** | Classes and methods are used correctly as required. Method inputs and outputs are well designed. | Some aspects of classes and methods are not well used, e.g. methods not used where they should be, problems with method/parameter design, incorrect use of objects. | Classes and methods used very poorly or not used at all. |
| **Commenting** | Code contains helpful # block comments, all classes and methods have meaningful docstrings and main module docstring contains all details (name, date, basic description, GitHub URL). | Comments are reasonable, but some classes and methods have no docstrings, and/or there is some noise (too many comments), and/or missing details in main module docstrings. | Commenting is very poor or not done. |
| **Formatting** | All formatting is appropriate, including indentation, horizontal spacing and vertical line spacing. PyCharm shows no formatting warnings. | Problems with formatting reduces readability of code. PyCharm shows multiple formatting warnings. | Readability is poor due to formatting problems. PyCharm shows many formatting warnings. |