THOMPSON RIVERS UNIVERSITY

Course Outline

Computing Science Department
Faculty of Science

COMP 2210
Programming Methods (3, 1, 0)

Instructor: Mohd Abdullah	**Phone/Voice Mail:** (250) 371-5961
Office: S140	**E-Mail:** Blackboard

Calendar Description

Students are introduced to the programming environments of visual and scripting languages along with tools and techniques of software development process. Students learn a combination of visual programing using C# and scripting language using Python in this course. Students learn the techniques of event driven visual application development, database and web connectivity, scripts, functions, strings, tuples and text file handling.

Learning Outcomes

Upon successful completion of the course, the student will demonstrate the ability to:

1. Provide a clear description of the software development process
2. Develop event-driven GUI applications
3. Construct solutions for small to medium sized problems using Python.
4. Describe the concept of designing, writing, debugging and testing software applications.

Prerequisites

C or better in COMP 1230 or equivalent

Required Texts/Materials

1. Starting out with Python, 4th Edition, by Tony Gaddis, Print ISBN: 9780134444321, 0134444329

2. Starting out with Visual C# (2015), 5th edition, by Tony Gaddis, Print ISBN: 9780135183519

2. TRU Lab/Network Computer Account.

Course Schedule:

- 3 Lectures per week (50 Minutes each)
- 1 Seminar per week (50 minutes each)

Student Evaluation

- **Python: 45%**
  - Assignments: 10%
  - Quizzes: 10%
  - Test: 25% (An oral test component could be included as part of a test)

- **C#: 55%**
  - Assignments: 15%
  - Quizzes: 10%
  - Tests: 30% (An oral test component could be included as part of a test)

In addition, to pass this course you must achieve and fulfilled the following requirements:

- Minimum 50% on the combined tests (i.e. 27.5 out of 55)
- Minimum 60% on the combined quizzes (i.e. 12 out of 20)
- Submitted at least 3 of the deliverable Python lab assignments and at least 4 of the deliverable C# lab assignments on time.
- Minimum 60% on the combined deliverable lab assignments (i.e. 15 out of 25)

Quizzes and Tests

- It is the department's and university's policy NOT to allow students to write quizzes and tests outside of the posted schedule time, except for bona fide compassionate or medical (A doctor's note is required) reasons.
- Each missed test or quiz receives a mark of zero (0) unless a note from a doctor is provided.
- It is your responsibility to attend all tests and quizzes as scheduled. Plan holidays and travel after you've seen the quiz and test schedule. Unavoidable absence should be reported to the Registrar as soon as possible. In order to write any missed tests or quizzes, a valid doctor's note is required before the quiz or test is due. Informing the instructor after the quiz or test is written will automatically disqualify you to write any quiz or test regardless of reason(s).
- After taking a test or quiz, generally there will be no chance to re-take the test or quiz.
- For final tests in each module, ONLY the marks are released, not the questions.

Grades

- Grade allocation is based on the standard for Academic/Career/Developmental Programs; refer to the current TRU Calendar. https://www.tru.ca/__shared/assets/ED_03-5_Grading_Systems35364.pdf

Attendance

- Where possible students are expected to attend EVERY lecture and lab. If a lecture or lab is missed, it is the student's responsibility to obtain the missed material from the LMS.

Assignments

- Any work submitted for marking must clearly identify the assignment, student name, student number, and anything else considered appropriated. All assignments will be prepared and submitted to a professional level of quality.
- A late assignment counts as a missed assignment and is generally not accepted, resulting in a zero mark, unless a medical note from a doctor is provided. If a due date needs to be extended it will be done for the whole class and not for individuals.

Seminars/Labs

- Seminars and labs will provide students with hand-on practices for the material learned in the lectures. Students are expected to complete each of them on their own time before the next seminar/lab session.

Academic Integrity

All assessments given for the computing science courses are governed by the academic honesty policy of both TRU and the Computing Science Department. Academic dishonesty is a serious offense. Academic and professional integrity policies are followed in every course. You should assume that any labs, exercises, or projects are to be done alone unless it is explicitly a team assignment. Any work you submit must be your own or that of your assigned team. Unless explicitly stated otherwise, all assignments and assessed activities must be your own original work.

In most cases, it is acceptable to share ideas with other students but not program source code or other written work. Jointly produced work, with the exception of designated teamwork, will result in a zero for that assignment for all students involved in the joint work and may result in loss of a letter grade for the course. The student(s) may also be expelled from the University. The TRU policy can be found at: https://www.tru.ca/_shared/assets/ed05-05657.pdf.

Plagiarism or cheating will not be tolerated and will be reported as per the policy. The Computing Science department is committed to reporting all transgressions of the Academic Integrity Policy. Note that this may result in zero grade for assignment, course failure, or expulsion from TRU.

Material obtained from the internet IS NOT YOUR OWN WORK, and thus its inclusion as part of your assignment, etc. would contravene this policy, unless you have your instructor's permission to include it. If your instructor does allow you to include material obtained from outside sources, any information, source code, or other asset that was obtained from someone other than yourself should be properly marked, such as with quotation marks and footnotes, and the author's name included where available.

Summary of Cheating: Description

What is Cheating?

- Sharing code: By copying, retyping, **LOOKING AT** distributing/supplying/providing/transmitting a file in any format
- Describing: Verbal description of code from one person to another person or group of persons
- Coaching: Helping someone to write an assignment, test, quiz, or exam whether wholly or partially
- Searching: Going online for Web solutions. NOTE: Material obtained from the internet **IS NOT YOUR OWN WORK**
- Copying code from a previous run of the course or online solution, except when the code snippet is supplied as part of an assignment, test, quiz, or exam

What is NOT cheating?

- Explaining how to use systems or tools
- Helping others with high-level design issues

Finally, ignorance is NOT an excuse.

Cheating Consequences:

- **Penalty for cheating:** Refer to: https://www.tru.ca/__shared/assets/ed05-05657.pdf

Detection of cheating:

- Various high-level sophisticated tools are now available for detecting code plagiarism
  DON'T DO IT!

Classroom Code of Conduct

All Students are expected to show respect for the rights of other students, the right to study and learn. Any behavior in a classroom that interferes with the instructor's ability to conduct the class will be treated as disruptive. In general students are expected to be attentive and courteous during class and lab time, to complete assigned work and to accept responsibility for their own achievement. In particular:

1. Students will not talk while the instructor is talking. Students will speak respectfully when asking a question or answering a question posed by the instructor.
2. Harassment is a serious offense and any violation under the B.C. Human Rights Act will be dealt with severely. The TRU policy can be found at https://www.tru.ca/careers/hr/respectful-workplace.html. When discussing assignments, marks, etc. with your instructor, nagging, badgering, etc. are not acceptable behaviors.

Ethical Standards

The following behaviors are considered unethical:

1. Telling the instructor that you "need" a certain grade.

2. Asking for extra assignments for the purpose of raising a grade.

3. Asking that the grade be raised because it is very close to the next higher grade.

4. Asking that the grade be raised because you did very well on one part of the course or grading scheme.

5. Asking for a higher grade because you don't like the grading scheme.

6. Asking to be allowed to turn in an assignment late – even a few minutes late – because of computer or printer

   problems or any other reasons.

7. Asking to be treated better than other students by making an exception to the rules.

8. Asking for any other unfair advantage in grading.

Lab Conduct/Behavior

1. During labs the student is expected to act in a professional manner towards other fellow students and instructors. Students not conducting themselves in such a manner may be asked to leave.

2. Students are to conduct themselves in a professional manner. Courtesy towards classmates and instructors is required at all times. Rude or disruptive behavior will **not** be tolerated.

3. Harassment is a serious offense and any violation under the B.C. Human Rights Act will be dealt with severely. The TRU policy can be found at http://www.tru.ca/__shared/assets/respectful_workplace_harrasment_prevention_poilcy28967.pdf.

4. When discussing assignments, marks, etc. with your instructor, nagging, badgering, etc. are not acceptable behaviors.

5. Internet browsing is strictly NOT allowed during the lecture time or lab time unless it is a part of the lecture and the instructor has told you to do so.

6. Use of e-mail program is strictly NOT allowed during the class and lab times.

7. There should be strictly no typing during the presentation of the lesson unless you have been asked to do so by the instructor.

8. If you are found not complying with any of the above stated the instructor has complete right to switch off your computer and/or to leave the room.

9. Lab assignments will **ONLY** be marked and given credit if the student had been present and participated in the seminar/lab activities and is in attendance for the entire lab class.

10. During lab/seminar times, the student is expected to work on course work only.

Tentative Course Topics (Python)   **Approx. Duration**

Chap. 1, Chap. 2. IPO, Chap. 3. Decision Structures and Logic	Week 1
Chap. 4. Repetition Structures, Chap. 5. Functions	Week 2
Chap. 6. Files & Exceptions	Week 3
Chap. 7. Lists & Tuples	Week 4
Chap 8. More about Strings, Chap. 9. Dictionaries	Week 5

Tentative Course Topics (C#)   **Approx. Duration**

Chap. 1 – 11, GUI Application development with C#	Week 6 - 10
Chap. 12, Databases and Chap. 14, LINQ (Time Permitting)	Week 11 - 12

Syllabus - Lab Topics:

Lab #	Topics	Tool	Week
1	Working with Python environment	Python	1
2	Making python scripts and functions	Python	2
3	Files and Exceptions	Python	3
4	List & Tuples	Python	4
5	Strings and Dictionaries	Python	5
6	Working with Visual Studio to make event driven applications	C#/Visual Studio	6
7	Working with Visual Studio to make event driven applications	C#/Visual Studio	7
8	Working with Visual Studio to make event driven applications	C#/Visual Studio	8

9	Working with Visual Studio to make event driven applications	C#/Visual Studio	9
10	Working with Visual Studio to make event driven applications	C#/Visual Studio	10
11	Working with Visual Studio to make event driven applications	C#/Visual Studio	11
12	Database connectivity	C#/Visual Studio	12
13	Database connectivity	C#/Visual Studio	12

Use of Technology

Python (Version 3.xx), Visual studio (2015/2017)

Lab Environment: Students will have all the software they need in the lab. The use of the lab computers is limited to COMP 2210 course activities: note taking, accessing course materials, and working on exercises/assignments.

Work from home: To work from home, students would need to have their own copy of the same software and similar environment. Please consult IT Services Web site and help desk for installation on your home computer.

E-mail system: *The students should use their TRU accounts for the e-mails to the instructor. The subject line should have the course number (COMP 2210), student name, and topic. E-mails sent from other accounts and/or without proper subject line will be NOT read by the instructor.*

To make the virtual classes interactive students are required to have the following:

• Webcam
• Microphone
• Headphone or speaker(s)
• The above tools might be needed for quizzes and tests as well.

ACM / IEEE Knowledge Area Coverage

Knowledge Areas that contain topics and learning outcomes covered in the course

Knowledge Area	Total Hours of Coverage
Algorithm and design	4
Fundamental programming constructs	4
Fundamental programming concepts	4
Visual Software Development	6
Project Development Concepts	6
Database Connectivity	6
Testing and documentation	6

Where does the course fit in your curriculum?
This course is designed as a second year course to introduce the student to the concepts of Software development in Command based and GUI environments.

What is covered in the course?
In this class two different programming languages are used to explain the concepts of Software Development. Python is used for explaining Console based applications and data structures and C# is used for Visual Software Design concepts.

What is the format of the course?

This is a one semester (Approx. 12 weeks) of class with 3 lecture hours and one seminar/lab hour per week.

Body of Knowledge coverage

KA	Knowledge Unit	Topics Covered	T1 hours	T2 hours	Elective hours
	Algorithm and Design	[Core-Tier1] • Role and purpose of algorithms • Functionality of a typical OS • Mechanism to support client/server model, hand-held devices • Design Issues • Influence of security, networking, multimedia, windowing systems	3	0	0
	Developing Console Applications	[Core-Tier1] • Lists • Arrays • Python Implementation	3	0	0
	Fundamental Programming Concepts	*Topics:* • Python shell files • Tuples and Dictionaries	6	0	0
	Visual Software Development	[Core-Tier2] • Basics of Event Driven Programming • Web Controls implementation • Visual software development using Visual Studio • Deadlines and real-time issues	0	6	0
	Project Development Concepts	[Core-Tier2] • Review of project development life cycle • Development of the required documentation	2	4	0
	Database Connectivity	• Need of databases • Connecting with database • Transferring data amongst databases		6	0
	Testing and documentation	• Test cases • Making documentations – project proposal & Technical document • Testing/Test cases • Software maintenance		6	

Changes to this Outline

Although changes are not expected as of now, the terms of this course outline could be subject to changes during the instruction period, in which case, students will be notified via an announcement through the LMS platform.