COMPUTER SCIENCE I (Spring 2017)

Course Information

Section:	CSCI 1370.01
Lectures:	9:25am-10:40am MW (in ENGR 1.290)
Laboratory:	10:40am-1:10pm F (in ASB 2.110)
Course Webpage:	http://meganstrait.com/teaching/csci1370/
Programming Environment:	https://repl.it/
C++ Reference Guide:	http://www.cplusplus.com
Textbook (optional):	https://zybooks.zyante.com/

Teaching Staff

Instructor:	Dr. Megan Strait
Email:	megan.strait@utrgv.edu
Office Hours:	3pm-5pm MW, 9:25am-10:40am R (in ENGR 3.275)
Anonymous Feedback:	http://sayat.me/meganstrait
Laboratory Instructor:	Dr. Emmett Tomai
Email:	emmett.tomai@utrgv.edu
Office Hours:	10:45am-12:00pm TR, 1pm-2pm W (in ENGR 3.210)
Teaching Assistant:	Pengfei Gu
Email:	pengfei.gu01@utrgv.edu
Office Hours:	4:20pm-6pm TR (in ENGR 3.273A)

Important Dates

February 1:	Census Date (last day to drop without record)
February 8:	Quarter Exam 1
March 8:	Quarter Exam 2
March 13-17:	Spring Recess (no class/lab)
April 12:	Quarter Exam 3
April 13:	Drop Deadline (last day to drop)
April 14:	Easter Holiday (no lab)
May 10:	Final Exam (8:00am-9:45am)

Objectives

This course presents an introduction to computer science. The fundamentals of a high-level programming language (C++) will be introduced. Methods of problem solving, techniques of algorithmic development, and concepts of object-oriented programming will be emphasized. In particular, students will learn:

1. How to design and write basic computer programs, including:
   a. How to analyze a problem and develop an appropriate solution.
   b. How to implement solutions in C++.
   c. How to organize and compile code into a working program.
   d. How to use testing and debugging strategies to identify and fix program faults.

2. How programming language, libraries and development environment each impact the way programs are written.

3. How a program can be written different ways. In particular:
   a. How different algorithms meet different requirements.
   b. How to modularize code for clarity, testing and reuse.
   c. How to evaluate, use and modify existing algorithms.

4. Learning how to write and document your code so that it is useful to other programmers.

Schedule

Below is a rough outline of the programming topics covered. The exact schedule is subject to change, but will be maintained and updated on the course webpage.

Weeks 1-2:	introduction to C++
Weeks 2-4:	variables and functions
Weeks 5-8:	control structures
Weeks 10-13:	arrays and pointers
Weeks 13-16:	classes

Materials and Grading

The course grade is computed as the sum (out of 100 points) of one's performance on three types of cumulative materials (see following) and converted based on UTRGV's point-to-letter mapping: A (90-100 points), B (80-89), C (70-79), D (60-69), F (0-59).

Materials and Grading (cont.)

Participation (25 points): Students are expected to complete regular assignments (7 points) and weekly labs (6.5). Students will also be assigned a programming group and graded based on attendance of weekly group meetings (6.5 points) and peer-based evaluations (5 points) of one's contribution to their group.

Quizzes (25 points): Each lecture period (with the exception of the first class meeting and exam days) will start with an individual, 10-minute, pen-&-paper quiz (1 point each).

Exams (50 points): There will be three 60-minute quarter exams (10 points each), assessed at the end of Weeks 4, 8, and 13, as well as a 90-minute final (20 points), assessed at the end of the semester. All exams are pen-&-paper and to be completed in-class, individually.

Course Policies

Attendance: Students are expected to attend all scheduled classes, be punctual, remain on task, and stay through the entire meeting period. The UTRGV attendance policy excuses students from attending class if they are participating in university-sanctioned activities (e.g., athletics, observance of religious holy days, or military service). In the case of *planned*, *excusable* absences, students must arrange with the instructor *in advance* of their absence to schedule make-up work. For medical emergencies and other unplanned but excusable absences, students should connect with the instructor as soon as possible to receive accommodations as warranted by the situation. No make-up options will be provided for unexcused absences during scheduled course assessments (quizzes, exams).

Collaboration: Collaboration amongst students enrolled in CSCI 1370 or 1380 is both welcome and strongly encouraged. At a minimum, students are expected to meet with their group members at least one hour per week towards working collaboratively on the participation activities. However, students are individually responsible for their respective submissions. That is, students are expected to produce their own submission and must be able to fully explain/reproduce the work that they submit – inability to do so if requested will result in a zero on the submission in question. Any assistance received from sources external to one's group must be acknowledged and appropriately credited. However, note that copy-pasting from any source (including from within one's group) is strictly prohibited.

Communication: The UTRGV email system is the primary medium for communication (Blackboard messages will *not* be checked). It is thus expected that students and staff check their UTRGV emails regularly. However, towards respecting external committments and balancing one's course involvement, all email communications will assume a window of up to 24 (M-F) or 48 (weekends/holidays) hours for the receiver to respond.

Course Policies (cont.)

Enrollment: Students must have *each* of the following prerequisites: computer literacy, credit for or concurrent enrollment in CSCI 1101, *and* a grade of "C" or better in MATH 1314 or MATH 1342 to 4399. In addition, students must be concurrently enrolled in CSCI 1170.01. This supplemental laboratory course provides hands-on instruction and practice with programming in C++and applying the principles taught in the CSCI 1370 lecture.

Extra Credit & Grade Adjustments: Extra credit in the amount of 1 point will be offered on each exam (excluding make-up exams) via "challenge questions" to encourage advanced critical thinking and problem solving. In addition, a bonus of up to 1.5 points will be added to students' exam scores (excluding make-up exams) based on the average of their group's performance (wherein a mean group score, M_g, of $\geq 90\%$ achieves a +1.5 point bonus; $M_g \geq 80\%$ a +1.0 point bonus; and $M_g \geq 70\%$ a +0.5 bonus). Finally, as the course material is cumulative, better performance on latter exams will replace student grades on all exams with lower scores taken prior to the exam in question (e.g., better performance on the second quarter exam replaces the grade of the first quarter exam).

Late Work: To provide some flexibility in the 60% completion rule (see below), late submissions of participation activities will be *accepted* up to three days past the first exam to follow the respective item. However, late work will *not* be graded. Specifically, late submissions will count towards one's completion rate but they will not receive grade credit.

60% Rule: Participation activities (assignments, labs, and group collaborations) are specifically intended to facilitate learning and quizzes are intended to provide regular feedback to students on their understanding. Exams, on the other hand, are exclusively for *demonstrating one's mastery* of the course material. Hence, it is assumed that *only* completion rates of $\geq 60\%$ of these preparatory materials (participation activities and quizzes) will lead to passing scores on exams. Thus, in order to take an exam, students must have completed *at least* 60% of *all* preparatory materials leading up to it. Students who do not meet this completion rate (by the last lecture preceding an exam) will be allowed three days from the original exam date to satisfy the requirement. Note: quizzes cannot be made up without a university-sanctioned excuse for missing the assessment. Excluding the final exam, students who did not initially meet the 60% requirement – but who subsequently satisfy it in time – will be offered a make-up exam seven days from the original exam date.

Resources: Students must sign up for an account with repl.it, the programming environment of this course. No textbook is required for this course, but students may optionally sign up for a zyBook account (course code: "UTRGVCSCI1370StraitSpring2017"). The zyBook is an online, interactive textbook and is available to students for $48. Alternatively or in addition, this reference guide is a freely available resource on programming in C++.

University Policies

Course Evaluation: Students are requested to complete an online evaluation of this course, accessed through your UTRGV account. You will be contacted through email with further instructions. Students who complete their evaluations will have priority access to their grades. Online evaluations will be available April 12 - May 3. Note: in addition to the university-requested evaluations, students may provide anonymous feedback at any time via the instructor's sayat.me page.

Course Withdrawal: According to UTRGV policy, students may drop any class without penalty earning a grade of DR until the official drop date. Following that date, students must be assigned a letter grade and can no longer drop the class. Students considering dropping the class should be aware of the *3-peat rule* and the *6-drop rule* so they can recognize how dropped classes may affect their academic success. The *6-drop rule* refers to Texas law that dictates that undergraduate students may not drop more than six courses during their undergraduate career. Courses dropped at other Texas public higher education institutions will count toward the six-course drop limit. The *3-peat rule* refers to additional fees charged to students who take the same class for the third time.

Scholastic Integrity: As members of a community dedicated to honesty, integrity, and mutual respect in all interactions and relationships, students, faculty, and administration of our university pledge to abide by the principles in the Vaquero Honor Code. Students should be reminded that those who engage in scholastic dishonesty are subject to disciplinary penalties, including the possibility of failure in the course and expulsion from the University. Scholastic dishonesty includes but is not limited to: cheating, plagiarism, and collusion; submission for credit of any work or materials that are attributable in whole or in part to another person; taking an examination for another person; any act designed to give unfair advantage to a student; or the attempt to commit such acts. Since scholastic dishonesty harms the individual, all students, and the integrity of the University, policies on scholastic dishonesty will be strictly enforced and all incidents of scholastic dishonesty will be reported to the Dean of Students. For more information, please see the Student Conduct and Discipline Code.

Sexual Harassment, Discrimination, and Violence: In accordance with UT System regulations, your instructor is a "responsible employee" for reporting purposes under Title IX regulations and so must report any instance, occurring during a student's time in college, of sexual assault, stalking, dating violence, domestic violence, or sexual harassment about which she/he becomes aware during this course through writing, discussion, or personal disclosure. More information can be found at `www.utrgv.edu/equity`, including confidential resources available on campus. The faculty and staff of UTRGV actively strive to provide a learning, working, and living environment that promotes personal integrity, civility, and mutual respect in an environment free from sexual misconduct and discrimination.

Accessibility

The instructor of this course is committed to providing a welcoming and comfortable environment to all students. Toward promoting student engagement and providing an environment free of socio-emotional, physical, and economic barriers to the successful study of Computer Science, the instructor is committed to accommodating – to the best of her ability – the interests and needs of students, proactively and as they arise.

Bilingual, Bicultural, and Biliterate Initiative Toward creating an accessible curriculum and reducing barriers to the study of computer science, the instructor is committed to promoting the core educational missions of UTRGV. In particular, the instructor aims to promote a learning environment that accommodates the linguistic needs of local communities and international students. In addition, the instructor will work to incorporate the bilingual, bicultural, and biliteracy mission into the course curriculum. Students with suggestions, questions, and/or concerns regarding this initiative and the current curriculum are encouraged to connect further with the instructor.

Students with Disabilities: If you have a documented disability (physical, psychological, learning, or other disability which affects your academic performance) and would like to receive academic accommodations, please inform your instructor and contact Student Accessibility Services to schedule an appointment to initiate services. It is recommended that you schedule an appointment with Student Accessibility Services before classes start. However, accommodations can be provided at any time. *Brownsville Campus*: Student Accessibility Services is located in Cortez Hall Room 129 and can be contacted by phone at (956) 882-7374 (Voice) or via email at `ability@utrgv.edu`. *Edinburg Campus*: Student Accessibility Services is located in 108 University Center and can be contacted by phone at (956) 665-7005 (Voice), (956) 665-3840 (Fax), or via email at `ability@utrgv.edu`.

ABET Learning Outcomes

Throughout this course, students will begin to develop the ability to: (A) apply knowledge of computing and mathematics appropriate to the discipline; (B) analyze a problem, and identify and define the computing requirements appropriate to its solution; (C) design, implement, and evaluate a computer-based system, process, component, or program to meet desired needs; (D) function effectively on teams to accomplish a common goal; (E) understand professional, ethical, legal, security and social issues and responsibilities; (I) use current techniques, skills, and tools necessary for computing practice; (J) apply mathematical foundations, algorithmic principles, and computer science theory in the modeling and design of computer-based systems in a way that demonstrates comprehension of the tradeoffs involved in design choices; (K) apply design and development principles in the construction of software systems of varying complexity.

Agreement

By signing below, the student acknowledges that:

1. They are responsible for all information contained in the syllabus and as communicated in-class and via email, the course website, and the course Blackboard section.

2. The syllabus in its current form is subject to change (students remain responsible for its content).

3. They pledge to abide by the principles in the Vaquero Honor Code.

Signature: _____

Date: _____

This page must be signed and returned to the instructor by January 25, 2017.