



# Course Outline

## Faculty of Science and Technology

<b>COURSE CODE (ID):</b>	CMPS1131
<b>COURSE TITLE:</b>	Principles of Programming 1
<b>CREDITS:</b>	3
<b>SECTION:</b>	O1
<b>COURSE MODALITY:</b>	Online
<b>COURSE MODALITY TYPE:</b>	<input checked="" type="checkbox"/> Synchronous <input type="checkbox"/> Asynchronous <input type="checkbox"/> N/A
<b>COURSE INSTRUCTOR:</b>	Mr. Farshad Rabbani, MSc.
<b>PROGRAM:</b>	AINT
<b>SEMESTER:</b>	2020-1
<b>CLASS DAYS:</b>	MW
<b>CLASS TIME:</b>	11:00 am – 12:15 pm
<b>CLASS VENUE:</b>	Online
<b>COMMUNICATION TOOL:</b>	<input checked="" type="checkbox"/> Zoom <input type="checkbox"/> Google Meet <input type="checkbox"/> N/A
<b>COURSE PLATFORM:</b>	<input checked="" type="checkbox"/> MOODLE <input type="checkbox"/> Google Classroom <input type="checkbox"/> N/A
<b>PRE-REQUISITES:</b>	None
<b>CO-REQUISITES:</b>	None
<b>FIELD TRIP(S):</b>	<input type="checkbox"/> Yes <input checked="" type="checkbox"/> No
<b>OFFICE LOCATION:</b>	Jaguar-U2 Room #D5, Belmopan Campus
<b>OFFICE/VIRTUAL HOURS:</b>	Tuesday, Thursday, Friday – 9:00 am - 11:00 or by appointment

**TELEPHONE:** Office: 822-1000 Ext. 305

**E-MAIL ADDRESS:** [frabbani@ub.edu.bz](mailto:frabbani@ub.edu.bz)

**RESOURCES:**

**REQUIRED:** Deitel, P.J. and Deitel H.M. (2005). *C++ How To Program* (5th ed.). Boston, Massachusetts: Prentice Hall.

**RECOMMENDED:**

**OTHER:** <http://doit.ub.edu.bz>

**COURSE DESCRIPTION:**

This course is intended primarily to teach students to analyze and solve problems effectively in computer programming. Topics include: Problem Solving – Analysis, Solution, Computer Algorithm. Programming Language Syntax and Semantics. Algorithm Design and Implementation using Conventional Programming Techniques. Data Types. Elementary Data Structures. Extensive Programming Activities Using a Modern Programming Language, such as C or C++.

**COURSE INTENDED LEARNING OUTCOMES (CILOs)**

1. Use the grammar, punctuation, and vocabulary of a third generation programming language
2. Explain and demonstrate the rudiments of the environment and the resources available to help them develop programs, including but not limited to editors, basic computer architecture, compilers, libraries, reference manuals, and on line help.
3. Use basic programming constructs: types (built in types, basic provided types including strings), values (constants, variables, assignment statements, and expressions), control structures (sequential flow, conditional statements, and loops) and functions.
4. Examine the value of applying software engineering techniques
5. Analyze a problem from an English description and design a program to solve it, so that the program is correct, easy to write, read, modify, and repair.
6. Use critical thinking skills to evaluate what a program is supposed to accomplish and devise a means to achieve it.
7. Demonstrate the ability to be creative and innovative in identifying solutions used to solve programming problems.
8. Demonstrate proper time management skills, and what sequence of steps to go through, in designing, writing, testing and debugging a program.
9. Use good documentation, formatting and naming conventions to ensure program readability.
10. Edit, compile, debug, execute and produce both a hard and soft copy of a simple program.
11. Communicate ideas clearly and efficiently in a variety of mediums (written, oral, visually and electronically).

12. Locate, gather and organize information using appropriate technology and information systems.
13. Work effectively within a group.
14. Provide support to a software development team while simultaneously monitoring the success of a project and identifying ways to improve it.
15. Work effectively and independently within prescribed timeframes.
16. Plan, design and carry out projects from start to finish, with well-defined objectives and outcomes.
17. Use research skills to access, analyze and apply knowledge to software development.
18. Respect creativity ownership as evidenced in innovations attributable to other individuals.
19. Adhere to the requirements and implementation practices involving software development to ensure compliance with local and industry laws, rules, and regulations.
20. Practice the code of ethics that governs the profession of Computer Science.

### COURSE SCHEDULE:

Date	Topics/Sub-topics	Teaching Strategies and Activities	Readings and Assignments	Assessment and Due Dates	Modality
	<b>Course review</b> - Course outline - Course requirements - Grading - Department Dishonesty Policies ----- 1. Basic hardware and software concepts 2. Basic object-technology concepts, such as classes, objects, attributes, behaviors, encapsulation and inheritance. 3. The different types of programming languages. Review a typical C++ program development environment.	a. Lectures – aid of presentation software  Discussions	<b>Readings:</b> - Course Outline  - IT department Cheating Policy document  <b>Chapter 1:</b> Sections 1.8, 1.9, 1.14	Quiz	Online
	Memory Organization, Number Systems and Data Storage	a. Lectures – aid of presentation software	<b>Readings:</b> - Lecture Notes 1	Quiz	Online

		Discussions			
	<ol style="list-style-type: none"> <li>1. Write simple computer programs in C++.</li> <li>2. Write simple input and output statements.</li> <li>3. How to use fundamental types.</li> <li>4. Basic computer memory concepts.</li> <li>5. How to use arithmetic operators.</li> <li>6. The precedence of arithmetic operators.</li> </ol> <p>How to write simple decision-making statements.</p>	<ol style="list-style-type: none"> <li>a. Lectures – aid of presentation software</li> <li>b. Discussions</li> <li>c. Code analysis</li> </ol> <p>Writing code blocks</p>	<b>Readings:</b> <b>- Chapter 2</b>	Quiz	Online
	<ol style="list-style-type: none"> <li>1. What classes, objects, member functions and data members are.</li> <li>2. How to define a class and use it to create an object.</li> <li>3. How to define member functions in a class to implement the class's behaviors.</li> </ol> <p>How to declare data members in a class to implement the class's attributes.</p>	<ol style="list-style-type: none"> <li>a. Lectures – aid of presentation software</li> <li>b. Discussions</li> <li>c. Code analysis</li> </ol> <p>Writing code blocks</p>	<b>Readings:</b> <b>Chapter 3</b> Sections: 3.1, 3.2, 3.3, 3.4, 3.5	<b>Assignments:</b> Program Set 1 (Due 2 weeks from assigned date)  Quiz	Online
	<ol style="list-style-type: none"> <li>1. How to call a member function of an object to make that member</li> </ol>	<ol style="list-style-type: none"> <li>a. Lectures – aid of presentation software</li> </ol>	<b>Readings:</b> <b>Chapter 3</b> Sections: 3.6, 3.7, 3.8	Quiz	Online

	<p>function perform its task.</p> <p>2. The differences between data members of a class and local variables of a function.</p> <p>3. How to use a constructor to ensure that an object's data is initialized when the object is created.</p> <p>How to engineer a class to separate its interface from its implementation and encourage reuse.</p>	<p>b. Discussions</p> <p>c. Code analysis</p> <p>Writing code blocks</p>			
	<p>1. Basic problem-solving techniques.</p> <p>2. To develop algorithms through the process of top-down, stepwise refinement.</p> <p>3. To use the <b>if</b> and <b>if...else</b> selection statements to choose among alternative actions.</p> <p>To use the <b>while</b> repetition statement to execute statements in a program repeatedly.</p>	<p>a. Lectures – aid of presentation software</p> <p>b. Discussions</p> <p>(Programming tasks)</p>	<p><b>Readings:</b> <b>Chapter 4</b> Sections: 4.1, 4.2, 4.3, 4.4, 4.5, 4.6, 4.7</p>	Quiz	Online
	<p>1. Counter-controlled repetition and sentinel-controlled repetition.</p> <p>To use the <b>increment, decrement and assignment operators.</b></p>	<p>a. Lectures – aid of presentation software</p> <p>b. Discussions</p> <p>Test1</p>	<p><b>Readings:</b> <b>Chapter 4</b> Sections: 4.8, 4.9, 4.10, 4.11, 4.12</p>	<p><b>Assignments:</b> Program Set 2 (Due 2 weeks from assigned date)</p> <p>Quiz</p>	Online
	<p>1. The essentials of counter-controlled repetition.</p> <p>2. To use the <b>for</b> and <b>do...while</b> repetition statements to execute statements</p>	<p>a. Lectures – aid of presentation software</p> <p>b. Discussions</p> <p>c. Code analysis</p>	<p><b>Readings:</b> <b>Chapter 5</b> Sections: 5.1, 5.2, 5.3, 5.4, 5.5, 5.6</p>	Quiz	Online

	in a program repeatedly. To understand multiple selection using the <b>switch</b> selection statement.	Writing code blocks			
	To use the <b>break</b> and <b>continue</b> program control statements to alter the flow of control.	a. Lectures – aid of presentation software b. Discussions  (Programming tasks)	<b>Readings:</b> <b>Chapter 5</b> Sections: 5.7	Quiz	Online
	1. To use the logical operators to form complex conditional expressions in control statements.  To avoid the consequences of confusing the equality and assignment operators.	a. Lectures – aid of presentation software b. Discussions c. Code analysis  Writing code blocks	<b>Readings:</b> <b>Chapter 5</b> Sections: 5.8, 5.9, 5.10	Quiz	Online
	1. To construct programs modularly from functions. 2. To use common math functions available in the C++ Standard Library.  To create functions with multiple parameters.	a. Lectures – aid of presentation software b. Discussions c. Code analysis  Writing code blocks	<b>Readings:</b> <b>Chapter 6</b> Sections: 6.1, 6.2, 6.3, 6.4, 6.12, 6.13	<b>Assignments:</b> Program Set 3 (Due 2 weeks from assigned date)  Quiz	Online
	The mechanisms for passing information between functions and returning results.	a. Lectures – aid of presentation software b. Discussions c. Code analysis  Writing code blocks	<b>Readings:</b> <b>Chapter 6</b> Sections: 6.14, 6.15	Quiz	Online
	1. How the function call/return mechanism is supported by the function call stack	a. Lectures – aid of presentation software b. Discussions	<b>Readings:</b> <b>Chapter 6</b> Sections: 6.7, 6.11	Quiz	Online

	and activation records. To use random number generation to implement game-playing applications.	c. Code analysis  Writing code blocks			
	1. How the visibility of identifiers is limited to specific regions of programs. To write and use recursive functions, i.e., functions that call themselves.	a. Lectures – aid of presentation software  b. Discussions Test2	<b>Readings:</b> <b>Chapter 6</b> Sections: 6.9, 6.10, 6.16, 6.17, 6.18, 6.19	Quiz	Online
	Review and Recap of material covered during the semester	a. Lectures – aid of presentation software  b. Discussions Laboratory work (Programming tasks)	<b>Readings:</b> Review	Possible Quiz	Online
		EXAM WEEK			

## **COURSE POLICIES AND REGULATIONS:**

### IT department's Policy on Academic dishonesty and cheating

#### **Academic Dishonesty**

Students attending UB are expected to earn degrees in their respective field of study on the basis of individual effort and determination. Therefore, any form of cheating or plagiarism on assigned coursework is deemed to be unacceptable deceit and dishonesty. Within the University community, such behavior is insupportable and will be punishable, according to the seriousness of the offense, as outlined in the student handbook of this institution.

#### **Cheating**

<sup>1</sup>Cheating is defined as follows: (a) the unauthorized granting or receiving of aid during the prescribed period of a course-graded exercise: students may not consult written materials such as notes or books, may not look at the paper of another student, nor consult orally with any other student taking the same test; (b) asking another person to take an examination in his/her place; (c) taking an examination for or in place of another student; (d) stealing visual concepts, such as drawings, sketches, diagrams, musical programs and scores, graphs, maps, etc., and presenting them as one's own; (e) stealing, borrowing, buying, or disseminating tests, answer keys or other examination material except as officially authorized, research papers, creative papers, speeches, etc. (f) Stealing or copying of computer programs and presenting them as one's own. Such stealing includes the use of another student's program, as obtained from the magnetic media or interactive terminals or from cards, print-out paper, etc.

#### **Punishment Guidelines for Academic Dishonesty:**

Punishments for academic dishonesty will depend on the seriousness of the offense and may include receipt of an "F" or "Zero" on the subject paper, lab report, etc., an "F" in the course, suspension or expulsion from the University.

- a. For observation of or exchanging test information with other students during the course of a classroom test, the students who receive or give such information may receive an "F" with a numerical value of zero on the test, and the "F" shall be used to determine the final course grade.
- b. For the use of any prohibited device, such as a cheat sheet, recording, calculator if forbidden on exam, etc., during the course of a classroom test to assist the student or other students, the student using such prohibited device may receive an "F" in the course.
- c. For the use of another student, a stand-in, to take an examination for the enrolled student, the matter will be referred to the Chair and Dean of the faculty for recommendation that the matter be sent to the Division of Student Affairs for disciplinary hearing with the suggestion that the enrolled student receive an "F" in the course and be suspended from school for one year and that the stand-in, if a University student, be suspended from school for one year.
- d. For stealing, borrowing, or buying of research papers, creative works, speeches or tests and other exam materials, or the dissemination of such materials, or the manipulation of recorded grades in a grade book or other class records, the matter will be referred to the Chair and Dean of the faculty for recommendation that the matter be sent to the Division of Student Affairs for disciplinary hearing with the suggestion that the student, if enrolled in the course, receive an "F" in the course and be expelled from the University.

---

<sup>1</sup> Policy taken from the University of South Florida, but modified for use at UB's department of Information Technology



- e. It is suggested that students who receive or give stolen computer programs receive an "F" with a numerical value of zero on the program or programs, and the "F" be used to determine the final course grade. It is the option of the instructor to fail the student in the course.

### **Policy Compliance Form**

#### Academic Dishonesty Policy

The policy on Academic Dishonesty at the University of Belize in the department of Information Technology states that:

- " a) The first occurrence will be punished by a zero, resulting in a reduction of the final grade.
- b) It is the option of the instructor to fail the student in the course if a second incident occurs.
- c) These penalties are to be applied to all students involved in the incident."

I, \_\_\_\_\_ have read and understand this policy.  
(print name)

\_\_\_\_\_  
(signature)

\_\_\_\_\_  
Date

1. **ACADEMIC HONESTY:** The administration of student discipline in the university community is a responsibility shared by students, faculty, and administrative staff. The University of Belize Academic Honesty Policy outlines the University's expectations for the integrity of student's academic work, the procedures for resolving alleged violations of those expectations, and the rights and responsibilities of students and faculty throughout the process. All students are expected to conform to the Academic Honesty Policy. Lecturers are expected to consult with academic department chairpersons to prevent and respond to violations of the Academic Honesty Policy. Students wishing to dispute a charge of academic dishonesty or a sanction made upon them because of such allegations can do so by appealing to the Dean of Student Affairs to invoke the Discipline Appeals Process as detailed in the Student Handbook. (Please visit [www.ub.edu.bz](http://www.ub.edu.bz) for a full description of violations to the Academic Honesty Policy and sanctions.)
2. **CLASSROOM POLICIES:** Once inside the computer lab, neither eating nor drinking are allowed under any circumstances. If, however, you must eat or drink, quietly excuse yourself from class and return when you have finished. As soon as class begins, no student is permitted to use the computer unless otherwise instructed by the lecturer.
3. **ASSESSMENT POLICIES:** Once a test has been returned to you, you have exactly one week from the date received to make queries or raise any concerns regarding your grade etc. Any concerns voiced after the specified period will not be valid or taken into consideration. Each student must take all tests. A makeup for a missed test will only be given when the student has made arrangements for such prior to the time the test is given to the class. I reserve the right to determine when an absence is to be excused or

unexcused. All programming and homework assignments **MUST** be turned in at the beginning of class on the date due. If, for any reason, you cannot attend class on a day an assignment is due, please notify me in advance or make necessary arrangements for it to get to me. Late homework assignments will **NOT** be accepted, and a grade of **ZERO** will be assigned. Barring tragedy, late programming assignments will not be accepted. There will be **NO** make-up quizzes given. I will **NOT** debug programs. Debugging is part of the assignment. If you are very confused about some compiler error messages, you can ask, but read the error message carefully and ask as a last resort, not a first. I am very unsympathetic toward questions that are asked on the program due date. You should be done by then.

4. **CLASS ATTENDANCE:** Students are expected to attend all sessions, and to be on time. Students must attend at least 80% of class sessions. Unexcused absences beyond this point may result in a reduction in your grade. You must provide the necessary documentation for any other absences e.g. doctors certificate if you are ill. You are expected to participate fully in class discussions and come to class prepared to contribute to class discussions and group work. Because participation in class is imperative for success, each student's final grade will be positively or negatively affected based upon the number of class sessions they attend. Absent students are responsible for obtaining class notes, handouts, and activities, as well as any other pertinent information. Points lost due to absences can be made up only through extra credit opportunities provided by the instructor.

If an instructor is late to a face to face class session or virtual classroom, the students are required to wait for the instructor at least **fifteen minutes** after the scheduled start of the class period. If the instructor has not arrived or logged in by that time, the students are free to leave unless specifically notified to await the instructor's arrival.

5. **PUNCTUALITY:** Kindly make every effort to arrive on time to all sessions. There is no penalty for coming to class late. However, walking into a session late is unprofessional and distracting to the rest your colleagues. As such, if you arrive late you must **NOT** disturb or distract the class while entering. If I see that you are inconsiderate and creating a lot of noise, I will ask you to leave. If you are late or absent, it is your responsibility to find out what information you have missed and what assignments are due. This does **NOT** mean coming to me to find out what you have missed or asking for a recap on what was presented earlier. If possible, it is recommended that you inform the instructor if you will be late for any session.
6. **DISABILITY CONSIDERATION:** Where necessary, and with adequate notice, instructors are required to make special arrangements, to allow students with special needs to participate in assessment without being placed at a disadvantage, for example use of enlarged fonts or Braille. Students requiring such considerations must be medically certified as having a specified special need.

- 7. PROFESSIONALISM AND ETHICS:** In your fieldwork and/or class activities, you are expected to conduct yourself in an ethical, legal, and professional manner. Most homework assignments in this course involve writing, testing, and debugging one or more programs. Unless otherwise specified by the instructor, the default policy in courses in the Department of Information Technology is that students are allowed and encouraged to discuss programming assignments with each other. However, the submitted programs must be their own work--each student must write the program independently. Sharing of code is not allowed.

In addition to the actual sharing of code, if students develop an outline together of an approach to a problem that yields code that is substantially identical between the students, the students will be considered to have improperly collaborated, even if no actual transfer of code took place.

It is, of course, understood that large parts of different student programs may be similar because each student used code samples from the course textbook or code pieces supplied by the instructor. Similarity due to such reasons is clearly not inappropriate collaboration if attributed appropriately. More precisely, whenever programs include code that the student did not write himself/herself, a comment that indicates the origin of the code should be included.

- 8. MAKING CONTACT:** If you need to see me outside of class, my hours of availability are posted outside my door. In addition, ALWAYS knock and ONLY when acknowledged do you enter the office. Please do not contact me at my home for any reason. Confine all correspondence to email and school time. If you are bothered by anything or would like to discuss any unfavorable situation that is affecting your learning, please feel free to see me. I will be more than happy to help. If I cannot assist you, then I will ask the chairman to look into the matter.
- 9. READINGS:** Readings will be assigned. You will be expected to attend the class session prepared to discuss the readings from the required text and/or any additional readings assigned. The aim is to enhance your understanding and skills related to these materials and to enable you to share your understanding with class members.
- 10. WRITINGS:** The ability to write clearly and effectively is essential to a profession. Written assignments represent your best professional abilities and excellence. Assignments must be written in Standard English. Written assignments should be typed and carefully proofread. Pages that are disorganized and contain errors in grammar, spelling, syntax, or typing will receive reduced grades. All written work should adhere to the APA style as directed by the instructor.

**11. CELL PHONES:** Cell phones can ONLY be set to vibrate mode or to ring ONCE. Once a call comes in, you have the option of answering it, but must do so OUTSIDE of the classroom and far enough to prevent distraction of other students.

**12. TECHNOLOGY:** Students are required to utilize word processing programs to complete all assignments. Internet and electronic mail systems will be used as needed to develop strategies, facilitate class discussion, and enhance communication between professor and students.

**13. PROGRAMMING STANDARDS:** Every program must contain, in the first few lines, a comment indicating the name(s) of the student(s) working on it and which assignment it is. Programs not containing this information, clearly visible, will get a zero.

- i. Every program must be accompanied by test cases, so I can see how it actually works. Programs with inadequate or poorly-chosen test cases will lose points. Many points will be lost if a program is turned in with no test runs.
- ii. I realize from past experience that sometimes you get stuck and are unable to get the program to work if your life depended on it. If this happens, turn in the program together with a detailed description of how the program fails, what you've tried in your attempts to fix it, and how those attempts didn't succeed. You won't get full credit, but if I'm convinced that you're working on it diligently, you'll get partial credit. Note that "how the program fails" does not mean saying "I got an error message": you need to tell me which error message you got, when you saw it, and what you think the error message means. Similarly, if the program fails by producing wrong answers, you need to tell me when it produces wrong answers (are they all wrong, or just in a few cases?), how they are wrong (e.g. are all the numbers consistently higher than you expected, are they the negatives of the correct answers, or are they all over the place with no apparent pattern?), and your speculations on how such an error might have arisen. I'm requiring all this not because I'm mean and horrible, but because by the time you've written all this down, you may have enough information to actually fix the problem, which is much better than turning it in incomplete.
- iii. I also expect you to maintain a log of what kinds of errors you encountered, how you discovered them, how long it took you to fix them, and what the actual problem was. This log must be turned in with each homework assignment.

**14. Microphone:** Upon entering a virtual classroom, kindly ensure that all microphones are on mute. To speak or participate in a discussion online, kindly unmute microphone. Once you have finish speaking kindly mute microphone in the classroom.

**15. Computer and Software Requirement:** Students enrolling in a web-enhanced course are required to complete Moodle for Students available on their course dashboard. Students enrolled in Blended and Online courses are required to complete Moodle for Students and E-learning for Students. In addition, students must have the basic computer, Internet

and Keyboarding skills. They must have access to a computer with Internet Service, a web browser, and a UB email account. Microsoft Office 2016 are standard at the UB Campus computer labs. Students may use any desktop productivity software of their choice as long as their instructor can access their work. When compatibility issues arise, it is the student's responsibility to resolve the problem with their instructor. Technical support for course navigation is available through the Office of Open and Distance Learning at [odl@ub.edu.bz](mailto:odl@ub.edu.bz).

- 16. E-mail Accounts:** Students taking courses will be assigned a UB e-mail account. University and course-related emails will be sent to this account. The emails inside of MOODLE will be used as the main communication tool for the University of Belize. All online students are expected to check their e-mail accounts on a regular basis to avoid missing important information sent out from the instructor or from the University.
- 17. Virtual Classroom:** Online students are encouraged to log into the virtual classroom as stipulated by the course instructor. This will allow students access to the instructor to seek clarity regarding an assessment, a course activity or to view the course weekly lecture. Virtual Classroom sessions are announced via the course page. This notification should include the day(s), time(s) and communication tool for the sessions.
- 18. Denying Course Access:** Approximately ten (10) workdays after the beginning of each semester, the office of Finance may purge students who have not paid their fees for the semester. Before students can be reinstated, they must pay their fees through the Office of Finance. Students may have their access reinstated completing the reinstatement form and getting the proper signatures for reinstatement. For Financial concerns kindly email [accountsreceivables@ub.edu.bz](mailto:accountsreceivables@ub.edu.bz)

#### **GRADING SCALE:**

The final grade will be assigned in adherence with the University of Belize's Grade Policy. Thus, final grades will be computed using a combination of semester grades (quiz, essays, tests, presentations, blog, wiki, quizzes, e-portfolio/ portfolios) and a final examination. The final letter grade will be assigned using the following University Grading Scale:

<b>95-100</b>	<b>A</b>	<b>80-84</b>	<b>B</b>	<b>65-69</b>	<b>D+</b>
<b>90-94</b>	<b>A-</b>	<b>75-79</b>	<b>C+</b>	<b>60-64</b>	<b>D</b>
<b>85-89</b>	<b>B+</b>	<b>70-74</b>	<b>C</b>	<b>0-59</b>	<b>F</b>

#### **CLASS ACTIVITIES AND ASSIGNMENTS:**

#### **ASSESSMENT:**

**(Note: This course outline is subject to change. Changes, if any, will be announced in advance, in class. Students will be held responsible for all changes.)**

Assignments – Program Sets (3)	15%
Quizzes	10%
Tests	35%
Final Exam	40%
<hr/>	
<b><u>TOTAL POINTS</u></b>	<b><u>100%</u></b>

#### **RUBRIC:**

The programming rubric below is used to evaluate most assignments. Each category can score a maximum of 20%.

	<b>Unsatisfactory 0-5%</b>	<b>Satisfactory 6% -10%</b>	<b>Good 11% - 15%</b>	<b>Excellent 16% - 20%</b>
<b>Delivery</b>	<ul style="list-style-type: none"> <li>Completed less than 70% of the requirements.</li> </ul>	<ul style="list-style-type: none"> <li>Completed between 70-80% of the requirements.</li> </ul>	<ul style="list-style-type: none"> <li>Completed between 80-90% of the requirements.</li> </ul>	<ul style="list-style-type: none"> <li>Completed between 90-100% of the requirements.</li> <li>Delivered on time, and in correct format (disk, email, etc.)</li> </ul>

	<ul style="list-style-type: none"> <li>Not delivered on time or not in correct format (disk, email, etc.)</li> </ul>	<ul style="list-style-type: none"> <li>Delivered on time, and in correct format (disk, email, etc.)</li> </ul>	<ul style="list-style-type: none"> <li>Delivered on time, and in correct format (disk, email, etc.)</li> </ul>	
<b>Coding Standards</b>	<ul style="list-style-type: none"> <li>No name, date, or assignment title included</li> <li>Poor use of white space (indentation, blank lines).</li> <li>Disorganized and messy</li> <li>Poor use of variables (many global variables, ambiguous naming).</li> </ul>	<ul style="list-style-type: none"> <li>Includes name, date, and assignment title.</li> <li>White space makes program fairly easy to read.</li> <li>Organized work.</li> <li>Good use of variables (few global variables, unambiguous naming).</li> </ul>	<ul style="list-style-type: none"> <li>Includes name, date, and assignment title.</li> <li>Good use of white space.</li> <li>Organized work.</li> <li>Good use of variables (no global variables, unambiguous naming)</li> </ul>	<ul style="list-style-type: none"> <li>Includes name, date, and assignment title.</li> <li>Excellent use of white space.</li> <li>Creatively organized work.</li> <li>Excellent use of variables (no global variables, unambiguous naming).</li> </ul>
<b>Documentation</b>	<ul style="list-style-type: none"> <li>No documentation included.</li> </ul>	<ul style="list-style-type: none"> <li>Basic documentation has been completed including descriptions of all variables.</li> <li>Purpose is noted for each function.</li> </ul>	<ul style="list-style-type: none"> <li>Clearly documented including descriptions of all variables.</li> <li>Specific purpose is noted for each function and control structure.</li> </ul>	<ul style="list-style-type: none"> <li>Clearly and effectively documented including descriptions of all variables.</li> <li>Specific purpose is noted for each function, control structure, input requirements, and output results.</li> </ul>
<b>Runtime</b>	<ul style="list-style-type: none"> <li>Does not execute due to errors.</li> <li>User prompts are misleading or non-existent.</li> <li>No testing has been completed.</li> </ul>	<ul style="list-style-type: none"> <li>Executes without errors.</li> <li>User prompts contain little information, poor design.</li> <li>Some testing has been completed.</li> </ul>	<ul style="list-style-type: none"> <li>Executes without errors.</li> <li>User prompts are understandable, minimum use of symbols or spacing in output.</li> <li>Thorough testing has been completed</li> </ul>	<ul style="list-style-type: none"> <li>Executes without errors excellent user prompts, good use of symbols, spacing in output.</li> <li>Thorough and organized testing has been completed and output from test cases is included.</li> </ul>
<b>Efficiency</b>	<ul style="list-style-type: none"> <li>A difficult and inefficient solution.</li> </ul>	<ul style="list-style-type: none"> <li>A logical solution that is easy to follow but it is not the most efficient.</li> </ul>	<ul style="list-style-type: none"> <li>Solution is efficient and easy to follow (i.e. no confusing tricks).</li> </ul>	<ul style="list-style-type: none"> <li>Solution is efficient, easy to understand, and maintain.</li> </ul>

## ASSIGNMENT DETAILS:

### 1. Assignment 1: Program Set 1 – Due Date: Week 6, (date subject to change with prior notice)

**Difficulty level: Low.** Program set consists of between 4-6 problems involving the fundamentals of programming. In particular displaying information and obtaining input from the user.

## GRADING CRITERIA

**Refer to above rubric**  
Program Set 1  
Worth 5%

**2. Assignment 2: Program Set 2 – Due Date: Week 9, (date subject to change with prior notice)**

**Difficulty level: Average.** Program set consists of between 4-6 problems involving the use of control structures such as repetition, selection and sequential.

**GRADING CRITERIA:**  
**Refer to above rubric**  
Program Set 2  
Worth 5%

**3. Assignment 3: Program Set 3 – Due Date week 14, (date subject to change with prior notice)**

**Difficulty level: High.** Program set consists of between 4-6 problems involving a more in-depth usage of control structures such as repetition, selection and sequential. It also includes the use of user-defined and library functions; parameter passing; classes and random number generation.

**GRADING CRITERIA:**  
**Refer to above rubric**  
Program Set 3  
Worth 5%

## **References**

### **On-Line Sources:**

Anon (2005) CIT Computer Programming Rubric 2005. Online. Bellevue University. Available from: <http://academic2.bellevue.edu/~partz/aac/rubricProgramming2005.doc> Accessed: 9 June 2010.