

Introduction to Computer Programming

SCHOOL OF ADVANCED TECHNOLOGY

Course Number: CST8110	Contribution to Program: Vocational Core	Normative Hours: 75
Applicable Program(s):	AAL:	Approval Date: 25/08/2009
0006X01FWO Computer Eng. Technology - Comp. Science	1	
0006X03FWO Computer Eng. Technology - Comp. Science	1	
0336X01FWO Computer Programmer	1	
0336X03FWO Computer Programmer	1	
1502X03FWO Business Intelligence Systems Analyst	1	
Prepared by: Barbara McIntyre Professor		Approved by: Charlie Inglis Chair
Co-Requisites N/A		Approved for Academic Year: 2009-2010
Pre-Requisites N/A		

COURSE DESCRIPTION

This course introduces the fundamental problem-solving methodologies needed in computing, such as structured analysis, structured design, structured programming, and an introduction to object-oriented programming. Pseudocode, flowcharting, as well as, a programming language are used to develop solutions to real-world problems of increasing complexity. Students learn the basics of robust computer programming, with emphasis on correctness, structure, style, and documentation using Java. This course reinforces theory with application by means of practical laboratory assignments.

RELATIONSHIP TO VOCATIONAL LEARNING OUTCOMES

This course contributes to your program by helping you achieve the following Vocational Learning Outcomes:	
Computer Eng. Technology - Comp. Science 0006X01FWO	
1	Diagnose, solve, troubleshoot, and document technical problems involving computing devices using appropriate methodologies. (T,A)
6	Analyze, build, test, implement, and maintain applications. (T,A)
8	Articulate, defend, and conform to workplace expectations found in technology environments. (T,A)
Computer Eng. Technology - Comp. Science 0006X03FWO	
1	Diagnose, solve, troubleshoot, and document technical problems involving computing devices using appropriate methodologies. (T,A)
6	Analyze, build, test, implement, and maintain applications. (T,A)
8	Articulate, defend, and conform to workplace expectations found in technology environments. (T,A)
Computer Programmer 0336X01FWO	
2	Develop, test, document, deploy, and maintain secure program code based on specifications.(T,A)
6	Use relevant methodologies, policies, and standards to develop secure program code.(T,A)
8	Conform to workplace expectations found in information technology (IT) environments.(T,A)
Computer Programmer 0336X03FWO	
2	Develop, test, document, deploy, and maintain secure program code based on specifications. (T,A)
6	Use relevant methodologies, policies, and standards to develop secure program code. (T,A)
8	Conform to workplace expectations found in information technology (IT) environments. (T,A)
Business Intelligence Systems Analyst 1502X03FWO	

1	Analyze and resolve information technology problems through the application of systemic approaches and diagnostic tools.(T,A)
2	Install, configure, troubleshoot, maintain, and upgrade business intelligence platforms.(T,A)
7	Apply knowledge of networking concepts to develop and deploy secure integrated solutions.(T,A)
8	Contribute as an effective team member to the successful completion of a business intelligence project.(T,A)

ESSENTIAL EMPLOYABILITY SKILLS

The course contributes to your program by helping you achieve the following Essential Employability Skills:

1	Communicate clearly, concisely and correctly in the written, spoken and visual form that fulfills the purpose and meets the needs of the audience.(T,A)
3	Execute mathematical operations accurately.(T,A)
4	Apply a systematic approach to solve problems.(T,A)
5	Use a variety of thinking skills to anticipate and solve problems.(T)
9	Interact with others in groups or teams in ways that contribute to effective working relationships and the achievement of goals. (T)
10	Manage the use of time and other resources to complete projects.(T,A)

COURSE LEARNING REQUIREMENTS/EMBEDDED KNOWLEDGE AND SKILLS

COURSE LEARNING REQUIREMENTS When you have earned credit for this course, you will have demonstrated the ability to:	EMBEDDED KNOWLEDGE AND SKILLS
1. Describe in detail the representation of data in a computer and be able to manipulate data as represented in the basic number systems used within a computer.	<ul style="list-style-type: none"> review basic computer architecture, specifically memory. detail the characteristics of the typical data types used in a high level language such as Java. These characteristics include size, format, range and precision. demonstrate knowledge of the data types <i>int</i> (integer), <i>char</i> (character) and <i>float</i> with respect to characteristics and use in a high level language such as Java. explain in detail the concepts of integer <i>overflow</i> and <i>underflow</i> for various sizes of <i>signed</i> and <i>unsigned</i> integer variables.
2. Define, analyze and document the logic of the solution to a given problem.	<ul style="list-style-type: none"> demonstrate logical problem solving techniques by solving assigned logic puzzles. develop software solutions by thinking in a systematic step-by-step manner. define the problem by using a narrative or prose description of the problem. document the problem solution by using diagrams such as a UML static class diagram and memory maps of run-time objects. devise and document the solution to the given problem by using the stepwise refinement process. develop the plan for the solution by writing the descriptive algorithm for the solution. determine the viability of the solution by performing a desk-check of the solution to the given problem.
3. Design a comprehensive <i>user-oriented test plan</i> to evaluate the solution developed for a given problem.	<ul style="list-style-type: none"> develop a user-oriented test plan that focuses on the results expected by the user based on varying input values.
4. Implement the solution of a given problem by writing the appropriate code in a high level language (Java).	<ul style="list-style-type: none"> use the console input/output instructions in Java; specifically <i>scanner class</i>. define elementary data types and operators using them in Java. evaluate Java expressions including Boolean expressions. use the Java online reference material to research use of Java library classes and methods in writing code. use Java instructions to write proper code to create selection pathways (<i>if</i>, <i>if-else</i>, <i>if-else-if</i>, and <i>switch-case</i>).

	<ul style="list-style-type: none"> • use Java instructions to write proper code to create iterative repetition (<i>for</i>, <i>while</i>, and <i>do-while</i>). • create and use classes to cluster related variables. • implement program modularity by creating appropriate user methods in proper Java code with reasonable use of return values and arguments. • create and use one-dimensional arrays of type <i>int</i>, <i>float</i> and programmer-defined classes. • implement elementary visual/graphical input/output.
5. Design a comprehensive <i>programmer-oriented test plan</i> to use when evaluating the program coded solution developed for a given problem.	<ul style="list-style-type: none"> • develop a programmer-oriented test plan that focuses on the value of variables at different stages of program execution (to be used in conjunction with the integrated debugger). • identify boundary conditions. • develop test conditions for both sides of boundaries. • develop comprehensive test data, including valid and invalid data.
6. Use test plans in conjunction with an integrated debugger to evaluate solution developed for a given problem.	<ul style="list-style-type: none"> • describe the function and operation of the compiler and virtual machine. • describe the outputs of the compiler. • use the IDE debug tools: breakpoints, step-in, step-over, locals, watch, call stack. • use console output for debugging. • use the compiler to find syntax errors and correct those errors. • use the debugger to find run-time logic errors and correct those errors. • use programmer-oriented test plan to predict variable value(s) at any point during program execution and use debug to verify the correct logical operation and correct any logic errors detected. • define the expected outputs and desk-test to evaluate the validity of the expectations.
7. Create documentation that adheres to the Computer Department submission standard.	<ul style="list-style-type: none"> • document the problem definition, algorithm and comprehensive test plan. • use the Computer Studies Department standard template for submissions and programming assignments as directed in the assignment description.

LEARNING RESOURCES

Recommended Textbooks:

- 1) Java How to Program, 8th Edition, by Deitel and Deitel, Published by Pearson Education Inc. in 2007, ISBN: 0-13-605306-8
- 2) eclipse Distilled, by David Carlson, Published by Pearson Education Inc in 2005, ISBN: 0-32-128815-7

LEARNING ACTIVITIES

During this course, you are likely to experience the following learning activities:

- classroom lectures,
- software demonstrations,
- online lectures,
- online software demonstrations,
- individual in-lab exercises (smaller tasks) and assignments,
- paired lab assignments (larger tasks),
- reading assignments,
- research of course-related material

Hybrid delivery:

This course is consists of 2 hours of lecture in classroom and 2 hours of lab work per week, as well as one hour per week of online material and activities available through Blackboard software.

Classroom Lectures:

- Theoretical course material will be presented, aided by use of overhead projections, videos, demonstrations and brief lecture notes.
- This course is hybrid. Approximately one hour per week outside of regular classroom lecture time will be required to review online materials and complete activities.
- Online theoretical material will consist of multi-media lectures and examples. This material will be accessible with free downloadable-from-Internet viewers (links provided) for use on home computers or with Access Centre/Resource Centre computers. Note: Students must bring in their own headsets to hear the audio component of material.
- Students are expected to attend all lectures and review all materials posted on the professor's website.
- Students will be expected to find and read applicable material in the textbook, and to be prepared to answer oral or written questions in lectures.
- Students are encouraged to ask questions during lectures and to consult with the professors on topics that they do not clearly understand. Students are also encouraged to ask/answer questions or initiate discussions using the Discussion Board forums in Blackboard.
- Professors will inform students, at the beginning of the course, of suitable times for consultations.

Labs:

- Students will apply the lecture material to a series of assignments which are closely integrated with the current lecture materials.
- The students' ability to successfully complete the assigned exercises will directly correlate with their level of success on tests and the final exam.
- Students should seek advice and help from the professor in the laboratory.
- Larger lab assignments can be completed on college computers or at home if students have appropriate equipment and software. Smaller lab exercises should be completed during the scheduled period.

Working in Pairs :

- Larger lab assignments will usually be handled in pairs. (Smaller lab exercises will be completed individually.)
- The pairings may be defined by the course professor.
- Typically, pairs will be rearranged every two to three weeks.
- Both members of a pairing must be present during demos and other forms of evaluation. An absent member may not receive credit for that particular lab assignment.

EVALUATION/EARNING CREDIT

The following will provide evidence of your learning achievements:	This activity validates the following Course Learning Requirements and/or Essential Employability Skills:
Classroom tests and quizzes 30%	<ul style="list-style-type: none"> • Describe in detail the representation of data in a computer and be able to manipulate data as represented in the basic number systems used within a computer. - [CLR 1] • Define, analyze and document the logic of the solution to a given problem. - [CLR 2] • Design a comprehensive <i>user-oriented test plan</i> to evaluate the solution developed for a given problem. - [CLR 3] • Implement the solution of a given problem by writing the appropriate code in a high level language (Java). - [CLR 4] • Design a comprehensive <i>programmer-oriented test plan</i> to use when evaluating the program coded solution developed for a given problem. - [CLR 5] • Communicate clearly, concisely and correctly in the written, spoken and visual form that fulfills the purpose and meets the needs of the audience. - [EES 1] • Respond to written, spoken or visual messages in a manner that ensures effective communication. - [EES 2] • Execute mathematical operations accurately. - [EES 3] • Apply a systematic approach to solve problems. - [EES 4] • Use a variety of thinking skills to anticipate and solve problems. - [EES 5]
Final examination 30%	<ul style="list-style-type: none"> • Describe in detail the representation of data in a computer and be able to manipulate data as represented in the basic number systems used within a computer. - [CLR 1] • Define, analyze and document the logic of the solution to a given problem. - [CLR 2] • Design a comprehensive <i>user-oriented test plan</i> to evaluate the solution developed for a given problem. - [CLR 3]

	<ul style="list-style-type: none"> • Implement the solution of a given problem by writing the appropriate code in a high level language (Java). - [CLR 4] • Design a comprehensive <i>programmer-oriented test plan</i> to use when evaluating the program coded solution developed for a given problem. - [CLR 5] • Communicate clearly, concisely and correctly in the written, spoken and visual form that fulfills the purpose and meets the needs of the audience. - [EES 1] • Respond to written, spoken or visual messages in a manner that ensures effective communication. - [EES 2] • Execute mathematical operations accurately. - [EES 3] • Apply a systematic approach to solve problems. - [EES 4] • Use a variety of thinking skills to anticipate and solve problems. - [EES 5] • Manage the use of time and other resources to complete projects. - [EES 10]
In-lab Exercises 10%	<ul style="list-style-type: none"> • Define, analyze and document the logic of the solution to a given problem. - [CLR 2] • Design a comprehensive <i>user-oriented test plan</i> to evaluate the solution developed for a given problem. - [CLR 3] • Implement the solution of a given problem by writing the appropriate code in a high level language (Java). - [CLR 4] • Design a comprehensive <i>programmer-oriented test plan</i> to use when evaluating the program coded solution developed for a given problem. - [CLR 5] • Use test plans in conjunction with an integrated debugger to evaluate solution developed for a given problem. - [CLR 6] • Communicate clearly, concisely and correctly in the written, spoken and visual form that fulfills the purpose and meets the needs of the audience. - [EES 1] • Respond to written, spoken or visual messages in a manner that ensures effective communication. - [EES 2] • Execute mathematical operations accurately. - [EES 3] • Apply a systematic approach to solve problems. - [EES 4] • Use a variety of thinking skills to anticipate and solve problems. - [EES 5] • Locate, select, organize and document information using appropriate technology and information systems. - [EES 6] • Analyze, evaluate and apply relevant information from a variety of sources. - [EES 7] • Manage the use of time and other resources to complete projects. - [EES 10]
Assignments 25%	<ul style="list-style-type: none"> • Describe in detail the representation of data in a computer and be able to manipulate data as represented in the basic number systems used within a computer. - [CLR 1] • Define, analyze and document the logic of the solution to a given problem. - [CLR 2] • Design a comprehensive <i>user-oriented test plan</i> to evaluate the solution developed for a given problem. - [CLR 3] • Implement the solution of a given problem by writing the appropriate code in a high level language (Java). - [CLR 4] • Design a comprehensive <i>programmer-oriented test plan</i> to use when evaluating the program coded solution developed for a given problem. - [CLR 5] • Use test plans in conjunction with an integrated debugger to evaluate solution developed for a given problem. - [CLR 6]

		<ul style="list-style-type: none"> • Create documentation that adheres to the Computer Department submission standard. - [CLR 7] • Communicate clearly, concisely and correctly in the written, spoken and visual form that fulfills the purpose and meets the needs of the audience. - [EES 1] • Respond to written, spoken or visual messages in a manner that ensures effective communication. - [EES 2] • Execute mathematical operations accurately. - [EES 3] • Apply a systematic approach to solve problems. - [EES 4] • Use a variety of thinking skills to anticipate and solve problems. - [EES 5] • Locate, select, organize and document information using appropriate technology and information systems. - [EES 6] • Analyze, evaluate and apply relevant information from a variety of sources. - [EES 7] • Interact with others in groups or teams in ways that contribute to effective working relationships and the achievement of goals. - [EES 9] • Manage the use of time and other resources to complete projects. - [EES 10]
Hybrid Exercises	5%	<ul style="list-style-type: none"> • Describe in detail the representation of data in a computer and be able to manipulate data as represented in the basic number systems used within a computer. - [CLR 1] • Communicate clearly, concisely and correctly in the written, spoken and visual form that fulfills the purpose and meets the needs of the audience. - [EES 1] • Respond to written, spoken or visual messages in a manner that ensures effective communication. - [EES 2] • Execute mathematical operations accurately. - [EES 3] • Apply a systematic approach to solve problems. - [EES 4] • Use a variety of thinking skills to anticipate and solve problems. - [EES 5] • Locate, select, organize and document information using appropriate technology and information systems. - [EES 6] • Analyze, evaluate and apply relevant information from a variety of sources. - [EES 7] • Manage the use of time and other resources to complete projects. - [EES 10]

COLLEGE GRADING NUMERICAL EQUIVALENT TABLE

Final Grade	Mark Equivalent	Numeric Value	Final Grade	Mark Equivalent	Numeric Value
A+	90-100%	4.0	C+	67-69%	2.3
A	85-89%	3.8	C	63-66%	2.0
A-	80-84%	3.6	C-	60-62%	1.7
B+	77-79%	3.3	D+	57-59%	1.4
B	73-76%	3.0	D	53-56%	1.2
B-	70-72%	2.6	D-	50-52%	1.0
			F	0-49%	0
			FSP	0	0

PRIOR LEARNING ASSESSMENT AND RECOGNITION

See College Directive E35 for details on eligibility and process. For this course, evidence of learning achievement for PLA candidates will include the successful completion of: a challenge exam with a breadth of coverage and level of difficulty equivalent to the final exam in the course; a hands-on or practical component to ensure that the requisite skill level has been achieved; and a computer programming (where applicable) assignment comparable to a representative assignment in the course.

RELATED INFORMATION

The following information is course-specific:

In order to pass this course, at least **50% or 'D-' (i.e. 20/40)** must be achieved in the exercises and the assignments. Additionally, at least **50% or 'D-' (i.e. 30/60)** must be achieved in the combined grade for tests and final examination in order for the exercise and assignment marks to be added into the final grade. **(Students who have a failing grade on the combined tests and the exam will receive a grade of 'F').**

The Computer Studies Department requires that all course assignments (homework exercises, laboratory work, projects, etc) be submitted by students using a standard which could be specific to one or more courses. Professors will ensure, at the beginning of the term, that students are advised of the exact details of these course specific submission requirements. Professors will also post them online alongside the course outline. Student submissions that do not meet the course published submission standards may not be marked, and may incur a penalty of up to 100% of the submission mark.

All students are required to write the final exam. There are no provisions for 'making up' a missed final exam. If, as a result of being off-track in your program or some unforeseen circumstance, you note that there is a scheduling conflict in your final exam schedule, it is your responsibility to alert your course professor no later than one week before final exams start, to allow for any special arrangements.

IN ORDER TO CONTINUE IN THE FOLLOW-UP COURSE CST8132 IN SECOND SEMESTER OF COMPUTER ENGINEERING TECHNOLOGY-COMPUTING SCIENCE OR CST8284 IN SECOND SEMESTER OF COMPUTER PROGRAMMER, STUDENTS MUST ACHIEVE AT LEAST A GRADE OF C-.

The following information is program-specific:

The following information is school/department-specific:

Retention of course material . It is your responsibility to retain copies of all assignments, labs and mid-term tests (returned from the professor), and any other evaluations and pertinent records (except for final exams, which are not returned) in case you become involved in an appeal hearing at a later date.

It is also your responsibility to retain course outlines for possible future use to support applications for transfer of credit to other educational institutions.

See College Directives E15 or E24 for details in your Instaguide.

Harassment/Discrimination/Violence will not be tolerated. Any form of harassment (sexual, racial, gender or disability-related), discrimination (direct or indirect), or violence, whether involving a professor and a student or amongst students, will not be tolerated on the college premises. Action taken will start with a formal warning and proceed to the full disciplinary actions as outlined in Algonquin College Directive - A8.

Harassment means one or a series of vexatious comment(s) (whether done verbally or through electronic means), or conduct related to one or more of the prohibited grounds that is known or ought reasonably to be known to be unwelcome/unwanted, offensive, intimidating, derogatory or hostile.

This may include, but is not limited to: gestures, remarks, jokes, taunting, innuendo, display of offensive materials, offensive graffiti, threats, verbal or physical assault, stalking, slurs, shunning or exclusion related to the prohibited grounds.

For further information, a copy of the official policy statement can be obtained from the Student Association.

Violation of the Copyright Act

- **General** – The Copyright Act makes it an offence to reproduce or distribute, in whatever format, any part of a publication without the prior written permission of the publisher. For complete details, see the Government of Canada website at <http://www.cb-cda.gc.ca/info/act-e.html> . Make sure you give it due consideration, before deciding not to purchase a textbook or material required for your course.
- **Software Piracy** - The Copyright Act has been updated to include software products. Be sure to carefully read the licensing agreement of any product you purchase or download, and understand the term and conditions covering its use, installation and distribution (where applicable). Any infringement of licensing agreement makes you liable under the law.

Disruptive Behaviour is any conduct, or threatened conduct, that is disruptive to the learning process or that interferes with the well being of other members of the College community. It will not be tolerated.

Members of the College community, both students and staff, have the right to learn and work in a secure and productive environment. The College will make every effort to protect that right.

Incidents of disruptive behaviour must be reported in writing to the departmental Chair as quickly as possible. The Chair will hold a hearing to review available information and determine any sanctions that will be imposed. Disciplinary hearings can result in penalties ranging from a written warning to expulsion.

For further details, consult the Algonquin College Directive - E27 in your Instaguide.

The following information is College-wide:

Email

Algonquin College provides all full-time students with an e-mail account. This is the address that will be used when the College, your professors, or your fellow students communicate important information about your program or course events. It is your responsibility to ensure that you know how to send and receive e-mail using your Algonquin account and to check it regularly.

Centre for Students with Disabilities (CSD)

If you are a student with a disability, it is strongly recommended that you identify your needs to the professor and the Centre for Students with Disabilities (CSD) by the end of the first month of the semester in order that any necessary support services can be arranged for you.

Academic Integrity

Adherence to acceptable standards of academic honesty is an important aspect of the learning process at Algonquin College. Academic work submitted by a student is evaluated on the assumption that the work presented by the student is his or her own, unless designated otherwise. For further details consult Algonquin College Directives

E16 (<http://www.algonquincollege.com/directives/sectionE/E16.pdf>)

and E43 (<http://www.algonquincollege.com/directives/sectionE/E43.pdf>).

Course Assessments

It is Algonquin College's policy to give students the opportunity to complete a course assessment survey in each course that they take which solicits their views regarding the curriculum, the professor and the facilities. For further details consult Algonquin College Directive E38

(<http://www.algonquincollege.com/directives/sectionE/E38.pdf>).

Use of Electronic Devices

With the proliferation of small, personal electronic devices used for communications and data storage, Algonquin College believes there is a need to address their use during classes and examinations. During classes, the use of such devices is disruptive and disrespectful to others. During examinations, the use of such devices may facilitate cheating. For further details consult Algonquin College Directive E39

(<http://www.algonquincollege.com/directives/sectionE/E39.pdf>).

Transfer of Credit

Students, it is your responsibility to retain course outlines for possible future use to support applications for transfer of credit to other educational institutions.