Object-Oriented Programming

Information and Communications Technology

<table>
<thead>
<tr>
<th>Course Number:</th>
<th>Contribution to Program:</th>
<th>Normative Hours:</th>
</tr>
</thead>
<tbody>
<tr>
<td>CST8132</td>
<td>Vocational</td>
<td>75</td>
</tr>
</tbody>
</table>

Applicable Program(s):

<table>
<thead>
<tr>
<th>AAL:</th>
<th>Core/Elective:</th>
</tr>
</thead>
<tbody>
<tr>
<td>0006X01FWO Computer Eng. Technology - Comp. Science</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>Core</td>
</tr>
<tr>
<td>0006X03FWO Computer Eng. Technology - Comp. Science</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>Core</td>
</tr>
</tbody>
</table>

Prepared by:
Rex Woollard
Course Professor

Co-Requisites
N/A

Pre-Requisites
CST8110

Course Professor

Approved by:
Andrew Pridham
Academic Chair, ICT

Approved for Academic Year: 2013-2014

COURSE DESCRIPTION

Students learn extended Object-Oriented Programming concepts and develop the solid programming skills that are prerequisites for higher-level courses. Java is used to illustrate the theoretical concepts and develop solutions to real-world problems of increasing complexity.

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

4. Plan, install, configure, modify, test, and maintain a variety of computer systems to meet functional requirements. (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)
9. Contribute to the successful completion of the project applying the project management principles in use. (T,A)

Computer Eng. Technology - Comp. Science 0006X03FWO

4. Plan, install, configure, modify, test, and maintain a variety of computer systems to meet functional requirements. (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)
9. Contribute to the successful completion of the project applying the project management principles in use. (T,A)

T: Teach A: Assess CP: Culminating Performance

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. (A)
3. Execute mathematical operations accurately. (A)
4. Apply a systematic approach to solve problems. (T,A)
5. Use a variety of thinking skills to anticipate and solve problems. (T,A)
6. Locate, select and document information using appropriate technology and information systems. (T,A)
7. Analyze, evaluate and apply relevant information from a variety of sources. (T,A)
10. Manage the use of time and other resources to complete projects. (T)
11. Take responsibility for one's own actions, decisions and consequences. (T)

T: Teach A: Assess CP: Culminating Performance
# 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:

<table>
<thead>
<tr>
<th>EMBEDDED KNOWLEDGE AND SKILLS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Explain the nature of and relationship between the following OO terms:</td>
</tr>
<tr>
<td>- class</td>
</tr>
<tr>
<td>- object</td>
</tr>
<tr>
<td>- encapsulation and data hiding</td>
</tr>
<tr>
<td>Draw diagrams that show the organization of objects and their fields in memory at run-time.</td>
</tr>
<tr>
<td>Explain the role of the garbage collector and its relationship to objects and references.</td>
</tr>
<tr>
<td>Control access and mutability using modifiers such as:</td>
</tr>
<tr>
<td>- private, public, protected and package-access</td>
</tr>
<tr>
<td>Write a test plan and Java code to implement the design using industry standard tools such as Eclipse or NetBeans.</td>
</tr>
<tr>
<td>Explain the role of new and reference variables in managing aggregation relationships.</td>
</tr>
<tr>
<td>Research potential paths to a solution using Java API documentation and other public resources.</td>
</tr>
<tr>
<td>Document program design using UML static class diagrams.</td>
</tr>
<tr>
<td>Explain the nature and relationship of the following Java-specific terms:</td>
</tr>
<tr>
<td>- reference variables</td>
</tr>
<tr>
<td>- primitive variables</td>
</tr>
<tr>
<td>- overloading based on method signature (a form of polymorphism)</td>
</tr>
<tr>
<td>Control mutability:</td>
</tr>
<tr>
<td>- using the final modifier.</td>
</tr>
<tr>
<td>- describe the immutable nature of the wrapper classes like Integer, Character and String and the implication of their immutability</td>
</tr>
<tr>
<td>Explain the difference between static fields and methods and instance fields and methods.</td>
</tr>
<tr>
<td>Explain the nature of and relationship between the following OO terms:</td>
</tr>
<tr>
<td>- inheritance</td>
</tr>
<tr>
<td>- polymorphism and overriding</td>
</tr>
<tr>
<td>Trace the execution sequence when subclass objects are instantiated, with particular attention to the chain of superclass constructor calls.</td>
</tr>
<tr>
<td>Draw diagrams that show object construction using the inheritance relationship of subclass objects in memory at run-time.</td>
</tr>
<tr>
<td>Implement abstract components to force subclass implementation.</td>
</tr>
<tr>
<td>Write a test plan and Java code to implement the design.</td>
</tr>
<tr>
<td>Explain the role of inheritance for code re-use.</td>
</tr>
<tr>
<td>Explain the difference between concrete classes, abstract classes and interfaces.</td>
</tr>
<tr>
<td>Write a test plan and Java code that captures subclass objects using super class references, and implement polymorphic behaviours.</td>
</tr>
<tr>
<td>Document program design using UML static class diagrams and sequence diagrams.</td>
</tr>
<tr>
<td>Use the assert statement to prove the correctness of program assertions.</td>
</tr>
</tbody>
</table>

## EMBEDDED KNOWLEDGE AND SKILLS

2. Analyze a problem description to identify inheritance relationships and polymorphic behaviours. Create corresponding OO class specifications.
operation. Co-relate uses of assert with test plan pre-conditions / post-conditions.

3. Explore the difference between arrays of primitives and arrays of references.
   - Draw diagrams that show the run-time organization of objects in memory for arrays of primitives and arrays of references.

4. Implement classes using selected design patterns.
   - Describe the philosophy of design patterns.
   - Identify existing classes that already implement design patterns, such as: Mutator/Assessor (Set/Get), Decorator (Wrapper), Factory Method, Iterator, Builder.
   - Implement design patterns such as Set/Get.

5. Implement program Input/Output operations.
   - Implement Serialization interface to support file I/O.
   - Explain the difference between Binary, Character, and Object streams.

6. Use existing exception handling mechanism to manage program errors at run-time.
   - Explain the exception handling mechanism, and the use of throws, throw, try, catch, finally. Explain the layers of use:
     - instantiate and throw an Exception object (using existing Exception classes).
     - handle an exception using:
       - catch to deal with an Exception object.
       - throw to transfer responsibility elsewhere.
   - Explain the difference between checked and unchecked exceptions.

7. Use existing simple Generic classes to manage objects.
   - Explain the underlying nature of object management and the role of the compiler in handling generic syntax.
   - Use selected Collection classes and describe the nature of memory allocation.
   - Use generic Interface reference-to variables to manage concrete generic Collection objects.

8. Generate program documentation and deploy Java application.
   - Generate class documentation using documentation generation tools.
   - Use existing tools to generate executable jar file.

9. Use the debugger to explore program structure and program execution.
   - Describe the difference between step-over, step-into, set-return.
   - Identify the call stack window and navigate to different views of program execution.
   - Document the relationship of reference-to variables and related objects.
   - Identify when the class loader is called and describe the work that it does.
   - Describe the differences between stack-oriented and heap-oriented variable allocation.

10. Implement simple JavaFX (optional and only if we have sufficient time).
    - Create a scenegraph, add displayable Node objects to the scenegraph.
    - Create a few selected GUI controls, and handle simple GUI Callbacks for those controls.

**LEARNING RESOURCES**

**Required Textbook:** (used in pre-requisite course CST8110)


This course is part of the mobile (laptop) program initiative at Algonquin College. Students are required to have a functioning laptop at all lecture and lab classes. The specifications for the required laptop and additional information about the mobile program initiative can be found at [http://mlearning.algonquincollege.com](http://mlearning.algonquincollege.com).

**Other Resources:**
Class Notes, Web-based resources

**Required Software:**
- Java Platform (JDK) 7u21 (or later update to Version 7)
- JavaFX Demos and Samples (2.2.21 or later update)
- Eclipse Juno 4.2.2 (or later update)
- NetBeans 7.3 (or later update)
  - [https://netbeans.org/downloads/](https://netbeans.org/downloads/)

**LEARNING ACTIVITIES**

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

- classroom lectures and program demonstrations,
- online lectures and program demonstrations,
- individual in-lab exercises and assignments,
- paired practical assignments,
- research of course-related material.

This course is delivered as a hybrid course. The course consists of 2 hours of classroom lecture and 2 hours of lab per week, as well as one hour per week online delivery of course material and associated activities. The online component of the course will be available through college web-based systems. Additionally, you will likely need to spend an average of 5 hours per week, of your own time working on assignments and study.

**Lectures:**

- In-class theoretical and practical course material will be presented using different visual aids.
- Approximately one hour per week outside your classroom lecture time will be required to review online materials and complete the related lecture activities.
- Students are expected to attend all of the lectures and review all of the materials posted in web-based repositories.
- Students will be expected to find and read applicable material in the textbook, and to be prepared to answer oral or written questions during class 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 Web based Discussion Board forums.
- Professors will inform students, at the beginning of the course, of suitable times for consultations.

**Labs:**

- The lab professor will commonly begin each period with an explanation of the current tasks. Students are expected to arrive at the start of lab sessions to receive that explanation.
- Students will apply the lecture material to a series of assignments which are closely integrated with the lecture material.
- The students' success in completing the assigned exercises independently will directly correlate with their level of success on tests and the final exam.
- Labs and assignments build on earlier work, and are increasingly complex as the course progresses.
- Students should seek advice and help from the professors in the laboratory. Students who have been attending labs may still need extra help from the professor outside of the scheduled lab period. In such cases, students should use the Blackboard discussion groups and seek the professor through e-mail and during his/her office hours.
- Larger assignments, and hybrid exercises can be completed at the college or at home. Smaller lab exercises should be completed during the scheduled period, especially, if they are graded and the grades recorded as in-lab activities.
- The students may be allowed to work in pairs on large assignments. The pairs may be defined by the lab professor. The pairs may be rearranged for every assignment. Both members of a paired team must be present during demonstrations and other forms of contact type evaluation. An absent member may not receive full credit (or credit at all) for a particular assignment. The team members may receive different credit depending on their level of contribution for a particular assignment.

**EVALUATION/EARNING CREDIT**

<table>
<thead>
<tr>
<th>The following will provide evidence of your learning achievements:</th>
<th>This activity validates the following Course Learning Requirements and/or Essential Employability Skills:</th>
</tr>
</thead>
<tbody>
<tr>
<td>Classroom quizzes 5%</td>
<td>- Analyze a problem description to identify inheritance relationships and polymorphic behaviours. Create corresponding OO class specifications.</td>
</tr>
<tr>
<td>Classroom mid-term test 20%</td>
<td>- [CLR 2]</td>
</tr>
<tr>
<td>Final examination 25%</td>
<td>- Use existing simple Generic classes to manage objects.</td>
</tr>
<tr>
<td></td>
<td>- [CLR 7]</td>
</tr>
<tr>
<td></td>
<td>- Implement classes using selected design patterns.</td>
</tr>
<tr>
<td></td>
<td>- [CLR 4]</td>
</tr>
<tr>
<td></td>
<td>- Use existing exception handling mechanism to manage program errors at run-time.</td>
</tr>
<tr>
<td></td>
<td>- [CLR 6]</td>
</tr>
</tbody>
</table>

© 2013 Algonquin College of Applied Arts and Technology
Implement program Input/Output operations. - [CLR 5]

Explore the difference between arrays of primitives and arrays of references. - [CLR 3]

Analyze a problem description and create Object-Oriented (OO) class specifications that correctly cluster related data and associated behaviours (in fields and methods); and that capture association relationships. - [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]

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]

Analyze, evaluate and apply relevant information from a variety of sources. - [EES 7]

Take responsibility for one’s own actions, decisions and consequences. - [EES 11]

Assignments 40%

Lab Exercises / Hybrid Activities 10%

Analyze a problem description to identify inheritance relationships and polymorphic behaviours. Create corresponding OO class specifications. - [CLR 2]

Use existing simple Generic classes to manage objects. - [CLR 7]

Generate program documentation and deploy Java application. - [CLR 6]

Analyze a problem description and create Object-Oriented (OO) class specifications that correctly cluster related data and associated behaviours (in fields and methods); and that capture association relationships. - [CLR 1]

Explore the difference between arrays of primitives and arrays of references. - [CLR 3]

Implement program Input/Output operations. - [CLR 5]

Implement classes using selected design patterns. - [CLR 4]

Use existing exception handling mechanism to manage program errors at run-time. - [CLR 6]

Use the debugger to explore program structure and program execution. - [CLR 9]

Implement simple JavaFX (optional and only if we have sufficient time). - [CLR 10]

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]

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.
Take responsibility for one’s own actions, decisions and consequences. - [EES 11]

**COLLEGE GRADING NUMERICAL EQUIVALENT TABLE**

<table>
<thead>
<tr>
<th>Final Grade</th>
<th>Mark Equivalent</th>
<th>Numeric Value</th>
<th>Final Grade</th>
<th>Mark Equivalent</th>
<th>Numeric Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>A+</td>
<td>90-100%</td>
<td>4.0</td>
<td>C+</td>
<td>67-69%</td>
<td>2.3</td>
</tr>
<tr>
<td>A</td>
<td>85-89%</td>
<td>3.8</td>
<td>C</td>
<td>63-66%</td>
<td>2.0</td>
</tr>
<tr>
<td>A-</td>
<td>80-84%</td>
<td>3.6</td>
<td>C-</td>
<td>60-62%</td>
<td>1.7</td>
</tr>
<tr>
<td>B+</td>
<td>77-79%</td>
<td>3.3</td>
<td>D+</td>
<td>57-59%</td>
<td>1.4</td>
</tr>
<tr>
<td>B</td>
<td>73-76%</td>
<td>3.0</td>
<td>D</td>
<td>53-56%</td>
<td>1.2</td>
</tr>
<tr>
<td>B-</td>
<td>70-72%</td>
<td>2.7</td>
<td>D-</td>
<td>50-52%</td>
<td>1.0</td>
</tr>
<tr>
<td>F</td>
<td>0-49%</td>
<td>0</td>
<td>FSP</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

**PRIOR LEARNING ASSESSMENT AND RECOGNITION**

Students who wish to apply for prior learning assessment and recognition (PLAR) need to demonstrate competency at a post-secondary level in all of the course learning requirements outlined above. Evidence of learning achievement for PLAR candidates includes:

- Portfolio
- Challenge Exam
- Other

The nature of a PLAR assessment will vary depending on the documented background of the student. The Challenge Exam may be oral or written, depending on the characteristics of the Portfolio presented by the student.

**RELATED INFORMATION**

The following information is course-specific:

In order to pass this course, at least **50% (i.e. 25/50)** must be achieved in the exercises and the assignments. Additionally, at least **50% (i.e. 25/50)** must be achieved in the combined grade for quizzes, test 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 test and the exam will receive a grade of "F").

Assignments must be submitted by the due date. Incomplete assignments will be accepted when they are due and will be eligible for appropriate partial marks. Late assignments will receive a grade of zero.

Some of the learning in this course can occur in partnered activities. To be a participating member of a partnership, you need to be present at all classes. A participation bonus of up to 5% may be added to a student’s grade if they have achieved a high threshold of attendance, have participated in bonus activities, and have been credited by their partner with contributing to partnered lab work. The details of this bonus will be defined by the course professor in class.

The final exam is designed to be a comprehensive, summative evaluation instrument. The final exam mark may replace the earlier test and quiz marks when it is to the advantage of the student. If a student’s mark on the final exam (as a percentage) is better than the test/quiz summary mark (as a percentage), then the test/quiz mark summary mark will be eliminated, and be substituted with a prorated exam mark. This mark adjustment process applies only when a student’s attendance has reached the participation bonus threshold.

The Computer Studies Department requires that all course assignments follow 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. 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.

The following information is school/department-specific:

**STUDENT ACADEMIC RESPONSIBILITIES**

Each student is responsible for:
- Knowing the due dates for marked out-of-class assignments.
- Attending all classes and knowing the dates of in-class marked assignments and exercises.
- Maintaining a folder of all work done in the course during the semester for validation claims in cases of disagreement with faculty.
- Keeping both paper and electronic copies of all assignments, marked and unmarked, in case papers are lost or go missing.
- Regularly checking both Blackboard announcements as well as one's Algonquin e-mail account for important messages from both professors.
and college administration.

- Participating in on-line and classroom exercises and activities as required.
- Retaining course outlines for possible future use to support applications for transfer of credit to other educational institutions.

**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 Policies - HR22 and SA07. 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://laws.justice.gc.ca/en/C-42. 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 terms 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 Policies AA32, SA07 and IT01 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** & **Plagiarism**

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 Policies AA18, http://www3.algonquincollege.com/directives/policy/academic-discipline/ and AA20 http://www3.algonquincollege.com/directives/policy/plagiarism/

**Student Course Feedback**

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 Policy AA25 http://www3.algonquincollege.com/directives/policy/course-assessment/  

**Use of Electronic Devices in Class**

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 Policy AA32 http://www3.algonquincollege.com/directives/policy/use-of-electronic-devices-in-the-academic-environment/ 

**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.

**Note:** College policies (previously called directives) are under review and redesign. The term directives is being retired. Students, it is your responsibility to refer to the Algonquin College Directives/Policies website for the most current information available at http://www3.algonquincollege.com/directives/.