



Ancaster High School
Course Outline 2013/2014
ICS 4U – Computer Science
Grade 12 University Preparation
Department: Business



TEACHER: Mrs. H. Dziewa

PREREQUISITE: ICS 3U

HOURS: 110

CREDIT VALUE: 1

DEPARTMENT HEAD: Mr. J. Russell

TEXTBOOK: None

GUIDELINE: The Ontario Curriculum, Grade 12, 2008

OVERALL EXPECTATIONS:

Programming Concepts and Skills

By the end of this course, students will:

- Demonstrate the ability to use different data types and expressions when creating computer programs;
- Describe and use modular programming concepts and principles in the creation of computer programs;
- Design and write algorithms and subprograms to solve a variety of problems;
- Use proper code maintenance techniques when creating computer programs.

Software Development

By the end of this course, students will:

- Demonstrate the ability to manage the software development process effectively, through all of its stages – planning, development, production, and closing;
- Apply standard project management techniques in the context of a student-managed team project.

Designing Modular Programs

By the end of this course, students will:

- Demonstrate the ability to apply modular design concepts in computer programs;
- Analyse algorithms for their effectiveness in solving a problem.

Topics in Computer Science

By the end of this course, students will:

- Assess strategies and initiatives that promote environmental stewardship with respect to the use of computers and related technologies;
- Analyse ethical issues and propose strategies to encourage ethical practices related to the use of computers;
- Analyse the impact of emerging computer technologies on society and the economy;
- Research and report on different areas of research in computer science, and careers related to computer science.

TEACHING STRATEGIES (include, but not limited to):

- Providing appropriate accommodation for students on IEP's and for English Language Learners and for those who are First Nations, Metis or Inui;
- Utilizing Student Support and Student Alternative Support Programs;
- Contacting parents for support and assistance;
- Using diagnostic assessment and check-in points to monitor student progress;
- Providing differentiation of instruction and assessment to meet the needs of diverse learners;
- Providing ongoing descriptive feedback that is clear, specific, meaningful, and timely to support improved student learning;
- Creating lessons, and assessment and evaluations, that are carefully planned to relate to the curriculum expectations and learning goals, and as much as possible to the interests, learning styles and preferences of all students;
- Developing students' self-assessment skills to enable them to assess their own learning, set specific goals, and plan next steps for their learning.

ASSESSMENT AND EVALUATION OF WORK:

Assessment and evaluation will be based on the provincial curriculum expectations and the achievement levels outlined in the curriculum policy document. Students will be given numerous and varied opportunities to demonstrate their achievement of the expectations across the four categories of knowledge and skills.

Midterm and final marks will be calculated using the prescribed learning strands with the following weighting:

Strand	Weighting
Programming Concepts and Skills	50%
Software Development	20%
Designing Modular Programs	20%
Topics in Computer Science	10%

Evidence of achievement can be determined from a variety of sources, including but not limited to: in-class assignments, class presentation, open-ended questions, observations, quizzes, unit tests, investigations, projects, conversations, portfolios, anecdotal records, self assessments, etc.

CULMINATING ACTIVITY

Culminating activities occur at or near the end of a course. They form part of the final 30% of a student's mark. If a student is absent from a culminating activity, they must provide a doctor's note. The culminating activity will not normally be re-scheduled.

For this course, the culminating activity will occur during the **last 6 weeks of the semester**.

And will consist of the following:

Students will use the C++ programming language to design and develop application software of their choice while following the stages of the software development process: planning, analysis, design, implementation, coding, debugging, testing and documentation (user manual).

MARK CALCULATION:

Interim: A report will be given to reflect how well the student is progressing with suggestions for improvement.

Term Work: 70% of the overall grade (from all term evaluations)

Final Evaluation(s): 30% of the overall grade (may include culminating activity, final exam or a combination of the two)

Teachers will take various considerations into account before making a decision about the grade to enter on the report card. Determining a report card grade will involve teacher's professional judgement and interpretation of the evidence and should reflect the student's most consistent level of achievement with special considerations given to the more recent evidence. Marks are not merely a calculation of averages, but an evaluation of the consistent achievement of the student.