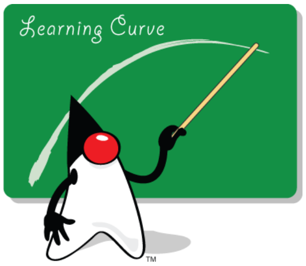
Computer Science, Grade 12, University Preparation (ICS4U)

TDSB – Forest Hill Collegiate Institute (416 393-1860)

COURSE DESCRIPTION



This course enables students to further develop knowledge and skills in computer science. Students will use modular design principles to create complex and fully documented programs, according to industry standards. Student teams will manage a large software development project, from planning through to project review. Students will also analyse algorithms for effectiveness. They will investigate ethical issues in computing and further explore environmental issues, emerging technologies, areas of research in computer science, and careers in the field.
(Computer Studies: The Ontario Curriculum Grades 10 to 12, Revised, 2008)

Prerequisite: Introduction to Computer Science, Grade 11, University Preparation (ICS3U).

CURRICULUM & OVERALL EXPECTATIONS

Course Strands	***Concepts***
A. Programming Concepts & Skills	A1. Data Types and Expressions A2. Modular Programming A3. Designing Algorithms A4. Code Maintenance
B. Software Development	B1. Project Management B2. Software Project Contribution
C. Designing Modular Programs	C1. Modular Design C2. Algorithm Analysis
D. Topics in Computer Science	D1. Environmental Stewardship and Sustainability D2. Ethical Practices D3. Emerging Technologies and Society D4. Exploring Computer Science

PROGRAMMING CONCEPTS & SKILLS

By the end of the 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 sub-programs to solve a variety of problems.
- Use proper code maintenance techniques when creating computer programs.

SOFTWARE DEVELOPMENT

By the end of the course, students will:

- Demonstrate the ability to manage the software development process effectively, through all of its stages -- planning, development, production, and closing (analysis, design, implementation, testing, maintenance).
- Apply standard project management techniques in the context of a student-managed team project.

DESIGNING MODULAR PROGRAMS

By the end of the 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 the 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.

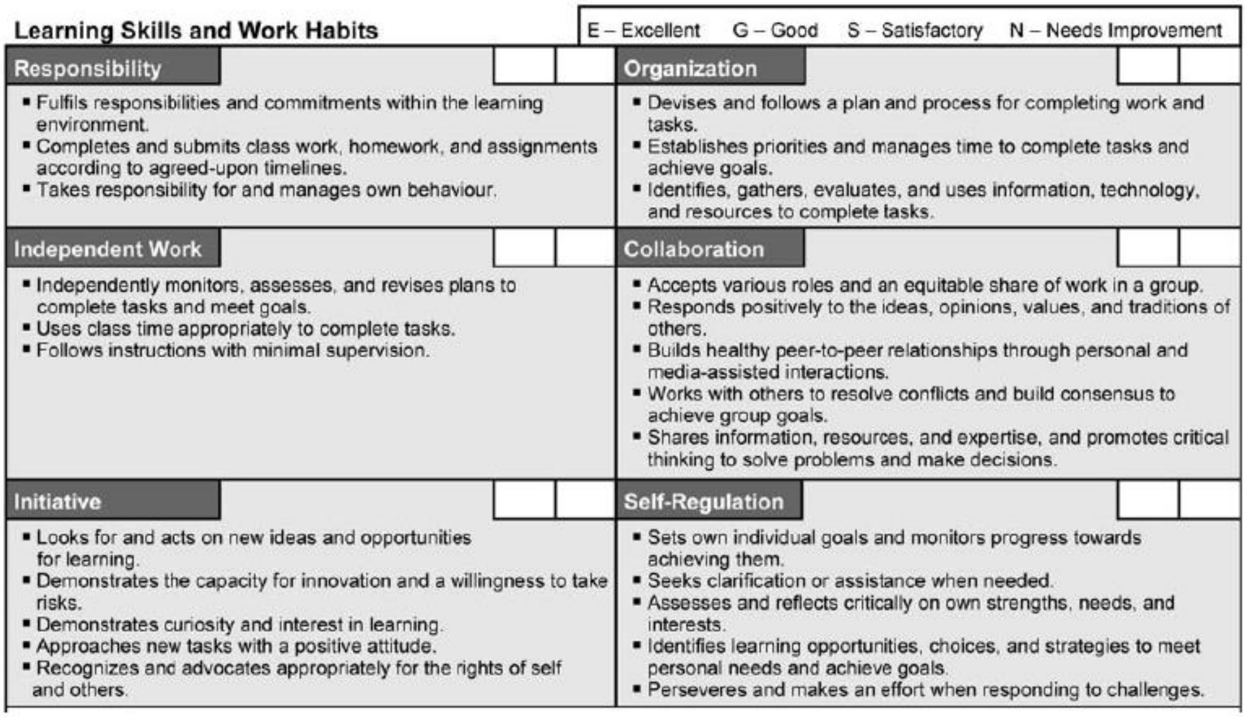
ASSESSMENT & EVALUATION

Evaluations will consist of traditional tests & quizzes, assignments, projects, group work, and presentations. Students can expect at least one written evaluation (quiz, test) and at least one major assignment or project per unit. To promote student success, ongoing formative assessment and feedback will be given to students.
The course expectations will be evaluated according to the four categories of the achievement chart.

Term Evaluations		
Knowledge & Understanding	20%	**70%**
Application	30%	
Thinking/Inquiry & Problem-solving (TIPS)	30%	
Communication	20%	
Course Culminating Evaluations		**30%**
Final Programming Project	15%	
Final Written Test	15%	
Final Course Mark		**100%**

LEARNING SKILLS ASSESSMENT

These skills will be assessed throughout the course and reflected on the report card.



Learning Skills and Work Habits

E – Excellent G – Good S – Satisfactory N – Needs Improvement

Responsibility
- Fulfils responsibilities and commitments within the learning environment.
- Completes and submits class work, homework, and assignments according to agreed-upon timelines.
- Takes responsibility for and manages own behaviour.

Organization
- Devises and follows a plan and process for completing work and tasks.
- Establishes priorities and manages time to complete tasks and achieve goals.
- Identifies, gathers, evaluates, and uses information, technology, and resources to complete tasks.

Independent Work
- Independently monitors, assesses, and revises plans to complete tasks and meet goals.
- Uses class time appropriately to complete tasks.
- Follows instructions with minimal supervision.

Collaboration
- Accepts various roles and an equitable share of work in a group.
- Responds positively to the ideas, opinions, values, and traditions of others.
- Builds healthy peer-to-peer relationships through personal and media-assisted interactions.
- Works with others to resolve conflicts and build consensus to achieve group goals.
- Shares information, resources, and expertise, and promotes critical thinking to solve problems and make decisions.

Initiative
- Looks for and acts on new ideas and opportunities for learning.
- Demonstrates the capacity for innovation and a willingness to take risks.
- Demonstrates curiosity and interest in learning.
- Approaches new tasks with a positive attitude.
- Recognizes and advocates appropriately for the rights of self and others.

Self-Regulation
- Sets own individual goals and monitors progress towards achieving them.
- Seeks clarification or assistance when needed.
- Assesses and reflects critically on own strengths, needs, and interests.
- Identifies learning opportunities, choices, and strategies to meet personal needs and achieve goals.
- Perseveres and makes an effort when responding to challenges.

For further details, see "*Growing Success*" (p. 9 – 14): http://www.edu.gov.on.ca/eng/policyfunding/growSuccess.pdf

ONTARIO SKILLS PASSPORT (OSP)

- Essential skills needed for work, learning, and life: Reading text, Writing, Computer Use, Measurement and Calculation, and Problem Solving (http://www.skills.edu.gov.on.ca/OSP2Web/EDU/Welcome.xhtml)

POLICIES & CLASSROOM EXPECTATIONS

- ***Academic Honesty*** – Students are expected to be academically honest. They are expected to submit their own work, so that the mark received reflects their own academic achievement.
- ***Lab Rules*** – computers and equipment are to be used for course-related work (see Acceptable Use Policy for more details)
- ***Care & Respect for Property*** – no food or drink in the lab, take care of equipment, report any issues/problems to teacher
- ***Online Code of Conduct*** as in the school agenda or http://www.tdsb.on.ca/communications/code_of_online_conduct/occ.html
- ***Assignments & Projects*** – assignments and projects are to be completed and submitted by the due date and are subject to late penalties (work submitted after the ultimate deadline will not be accepted and will receive a mark of zero)

RESOURCES, COURSE MATERIALS, USEFUL WEBSITES (PROGRAMMING & ONLINE TUTORIALS)

- USB flash drive or external HDD for storing and transporting digital material and work between school and home
- 3-ring binder, paper, writing utensils (pen, pencil, eraser)
- *Ready to Program Java IDE (http://compsci.ca/holtsoft/)*
- *Dr. Java IDE (http://drjava.org/)*
- *Eclipse IDE (https://www.eclipse.org/)*
- *Google classroom*
- *Textbooks: various (several texts are available as resource/reference in classroom and online)*
- *Chortle Online Java Tutorial (http://chortle.ccsu.edu/CS151/cs151java.html)*
- *Programming Using Java (an online text) (http://math.hws.edu/javanotes/)*
- *Oracle Java Tutorial (http://download.oracle.com/javase/tutorial/)*
- *The Java API (http://download.oracle.com/javase/1.4.2/docs/api/)*

PROVISIONS FOR STUDENT SUCCESS

Teacher Support:	**Student Responsibilities:**
• *Extra help* • *Computer Lab hours* *During lunch or after school (arrange time with teacher in advance)*	• Keep an organized binder to assist with your studies • Keep an organized "digital binder" of all digital materials provided • Set and focus on realistic goals for each class, each unit, the whole course • Record daily achievements to set and meet new challenges • Review learning at home daily • Home study in preparation for each class to enrich your learning • Take advantage of extra help and lab hours to assist in meeting goals • Provide help to peers to consolidate your learning and increase confidence • Ask and answer questions; look hard for answers; participate!

Teacher Support:	**Student Responsibilities:**
• *Extra help* • *Lab hours* • *Lunch, after school, or arranged time teacher* • *Web sites addresses given according to the curriculum.*	• Organize a digital binder to assist with your studies • Set and focus on realistic goals for each class • Record daily achievements to set and meet new challenges • Home study in preparation for each class to enrich your learning • Take advantage of extra help and school lab time to assist in meeting goals • Provide peer help to consolidate your learning and increase confidence • Ask questions and look for answers

Computer Science (ICS4U) – Course Content by Unit

Unit 1: Introduction & Review

Review concepts from grade 11 (ICS3U).

Concepts for review:

- variables & data types (including Strings)
- scope (local vs. global)
- arithmetic operators & expressions
- type conversions
- basic structures including sequence, selection, repetition, nesting
- boolean operators & expressions
- modular programming – sub-programs (methods)
- errors - syntax, logic, run-time vs. compile-time
- user-friendly design, formatting
- tracing/testing, debugging
- planning - flow charts, pseudo-code, software design cycle
- arrays (1D)
- error-checking and handling with user input

Unit 2: Object-Oriented Programming (OOP) & Modular Design

OOP concepts within Java and Java languages framework.

- user-defined multiple classes (super class / sub class hierarchies)
- class hierarchical diagrams
- modular programming (libraries, utility files, data files, etc.)
- encapsulation, inheritance, polymorphism
- overriding/overloading
- abstract and interface classes
- GUIs

Assignments:

1. modify existing code to
   - add more features
   - correct defects
   - critique/evaluate
   - add graphics
2. potential classes to build on through semester using Java

Unit 3: Designing & Analyzing Algorithms

Several concepts in units 1 and 2 are now to be applied as components within larger OO programs through multiple files and inheritance.

- searching & sorting algorithms (efficiency)
- File IO (text/binary)
- recursion (efficiency & analysis)
- 2D arrays
- data encapsulation

Real Life Project will reinforce the concepts listed above
Working in groups will develop the following:

- project management
- code maintenance
- code reusability
- external & internal documentation

Unit 4: Other Topics in Computer Science

These topics are through research assignments, group discussions through out the term.

- ethics
- environmental
- research in computer science & careers

Unit 5: Team Project

Final projects: Assigned team projects will tie everything together from previous units. Students should concentrate on the following throughout the project:

1. Software project planning
   - UMLs
   - algorithmic design
   - flow charting, pseudo-code, UML diagrams
   - modularized code
   - user-friendly design
2. Project management
   - client requirements
   - meeting project goals/deadlines
   - project progress reports
   - use of version control
   - self assessment
3. Software project contributions
   - teamwork
   - time management
   - individual contributions
4. Code Maintenance
   - code testing & debugging
   - internal documentation to industry standard
   - external documentation
5. Final presentation of project through emerging technologies

Since this is a university prep course, there would be a written exam which may include material from the student's term project.