Computer Programming 11

The ability to design and make, acquire skills as needed, and apply technologies is important in the world today and a key aspect of educating citizens for the future. This curriculum is an experiential, hands-on program of learning through the design and creation of computer programs using the Python programming language.

Course Content

- design opportunities
- design cycle
- problem decomposition
- structures within existing code
- ways to modify existing code to meet a particular purpose
- strategies to predict effects of code modification
- pair programming
- programming language constructs to support input/output, logic, decision structure, and loops
- requirements of a problem statement
- ways to transform requirements into algorithms
- translation of design specifications into source code
- tools to aid in the development process
- pre-built libraries and their documentation
- inline commenting to document source code
- use of test cases to detect logical or semantic errors
- computational thinking processes
- appropriate use of technology, including digital citizenship, etiquette, and literacy

Curricular Competencies

Applied Design

Understanding context

- Conduct user-centred research to understand design opportunities and barriers

Defining

- Establish a point of view for a chosen design opportunity
- Identify potential users, intended impact, and possible unintended negative consequences

- Make inferences about premises and constraints that define the design space

Ideating

- Identify gaps to explore a design space
- Generate ideas and add to others' ideas to create possibilities, and prioritize them for prototyping
- Critically analyze how competing social, ethical, and sustainability considerations impact designed solutions to meet global needs for preferred futures
- Work with users throughout the design process

Prototyping

- Identify and apply sources of inspiration and information
- Choose an appropriate form, scale, and level of detail for prototyping, and plan procedures for prototyping multiple ideas
- Analyze the design for the life cycle and evaluate its impacts
- Construct prototypes, making changes to tools, materials, and procedures as needed
- Record iterations of prototyping

Testing

- Identify feedback most needed and possible sources of feedback
- Develop an appropriate test of the prototype
- Collect feedback to critically evaluate design and make changes to product design or processes
- Iterate the prototype or abandon the design idea

Making

- Identify appropriate tools, technologies, materials, processes, and time needed for production
- Use project management processes when working individually or collaboratively to coordinate production

Sharing

- Share progress while creating to increase opportunities for feedback
- Decide on how and with whom to share or promote their product, creativity, and, if applicable, intellectual property
- Consider how others might build upon the design concept
- Critically reflect on their design thinking and processes, and identify new design goals
- Assess ability to work effectively both as individuals and collaboratively while implementing project management processes

Applied Skills

- Apply safety procedures for themselves, co-workers, and users in both physical and digital environments

- Identify and assess skills needed for design interests, and develop specific plans to learn or refine them over time

Applied Technologies

- Explore existing, new, and emerging tools, technologies, and systems to evaluate their suitability for their design interests
- Evaluate impacts, including unintended negative consequences, of choices made about technology use
- Analyze the role technologies play in societal change
- Examine how cultural beliefs, values, and ethical positions affect the development and use of technologies

Learning Materials and Assessment

All assignments, resources and assessment rubrics will available at:

```
transformativesuccess.ca
```

Content:
```
01  Python – Variables (data types and structures)
02  Python – Lists (data types and structures)
03  Python – Basic Operators (output and syntax)
04  Python – Conditions (flow control and logic)
05  Python – Placeholders (output and syntax)
06  Python – Loops (flow control and logic)
07  Python – Boolean Logic and User Input (flow control and logic, input)
07b Python – Password Challenge (flow control and logic, input)
08  Python – Error Handling
09  Python – Classes and Objects (data types and structures)
09b Python – MesoProject in Applied Design (application of concepts)
10  Python – Functions
11  Python – Dictionaries (data types and structures)
12  Python – Roll the Dice (application of concepts)
13  Python – Guess the Number Game (application of concepts)
14  Python – Mad Lib (application of concepts)
15  Python – Adventure Game (application of concepts)
16  Python – Custom Adventure Game (application of concepts)
17  Python – Hangman Game (application of concepts)
18  Python – Validating String Input (application of concepts)
```

```
19  Python – Applied Design Collaborative Project (application of
concepts)
```

Important Assessment Notes:

- Students must do unit tests in order to have the activity marks counted for that unit. IE. If you do four activities then skip the test, none of the four activity marks will be counted.
- There will be two mandatory, self-directed "Applied Design" projects. One will take place approximately mid-way through the course, and the second will be near the end of the course.
- Tests 40%, Activities and Projects 60%

Expectations

- Respect Yourself and Others
- Always demonstrate respectful behavior
- Respect Effort and Learning
- Respect your Community and the Environment
- Attend class on time
- Attend tutorials to complete work
- Always put forth your best effort
- Listen to the teacher and to your work-partners
- Absolutely no food or drink in the computer labs
- Be careful not to damage school equipment. Don't unplug any of the computer cabling without the direction of the teacher