Mr. Kim's Course Yearly Plan Email – kkim@wvschools.ca Website – https://mrkim.2myclass.com Classroom - https://classroom.google.com Linear – 43 weeks 301 days – 193 days or 37 weeks with 186 instructional days or 952 hours

Course	Week	Week	Week	Week	Week
IT 8	-Classroom Expectations & Procedures -File Management -SMART Goals -set up Google Classroom and other LMS -introduce algorithmic constructs (shapes and arguments, functions, variables, if conditional statements, for loops, mouse, key and text input, arrays)	**-MYP Unit-** **-Making Algorithmic Designs (MYP Criteria A, B, C, D)** **Computational Thinking and Design Process** – summative task – p5.js algorithmic constructs / HTML (shapes, shape parameters/arguments, variables, libraries, functions, conditionals)	**Data Analysis** – summative task – WPM spreadsheet using data from typing quizzes and telling a story using the bar chart as the visualization	**Keyboarding Technique** – summative task – SMART goal setting – using proper posture and finger technique on a QWERTY keyboard to complete 10 typing tests	**Digital and Financial Literacy** – summative task – Everfi Ignition Digital Wellness and Safety 1. Connections and Community 2. Safety and Privacy 3. Screen Time vs Online Time 4. Technology and Privacy 5. Rights and Literacy 6. Evaluating Content

Course	September	October	November	December	January	February	March	April	May	June
IT 9	-Classroom Expectations & Procedures -File Management -SMART Goals -set up Google Classroom and other LMS	-MYP Unit – Problem Solving Unit (explore problem solving process and the different ways humans and computers solve problems) MYP Criteria C 2 weeks -introduction to computational thinking -algorithmic framework – the building blocks of programming -summative task – recognize the different types of programming constructs with pseudocode and flowcharts	- MYP Unit- Programming Unit (use p5.js to create programs that work on algorithmic constructs) MYP Criteria A, B, C, D 8 weeks Key Concepts: U1 Draw and Draw with Functions U2 Respond and Draw with Canva U3 Arrays, Loops, Media U4 Motion, Objects, Transformation U5 Final Project -summative task – create programs using shape functions, function libraries, conditionals, event actions, arrays, loops, media, motion, objects			-MYP Unit – Data and Society Unit (learn how information is represented, collected, analyzed, and visualized by computers) (investigate how data is collected online and weigh the potential benefits and harms to individuals and society at large) MYP Criteria C, D 2 weeks -summative task - manipulate data using an infographic to persuade an audience on topic of choice (use Python modules – Matplot lib – four subplots – scatter, histogram, bar chart, pie chart with a story			-MYP Unit – Process Design Unit (discover the programming constructs that allows us to design apps) MYP Criteria A, B, C, D 7 weeks CS4All – App Inventor Unit 0 - Apps Unit 1 – I have a Dream App Unit 2 – Paint Pot App Unit 3 – Game Apps Unit 4 – Magic 8 Ball App Unit 5 – Design Project -summative task - app design project	

Course	Week	Week	Week	Week	Week	Week	Week	Week	Week	Week	Week	Week	Week
ICT 11 Programming **ICTP 11 (4 credits)** -Classroom Expectations & Procedures -File Management -SMART Goals -set up Google Classroom and other LMS	**Programming Constructs** -Bad Choices: An Illustrated Introduction to Computational Thinking -Algorithm Framework - Programming Constructs (flowcharts and pseudocode) (sequencing, selection, iteration, combination)	**-Python Programming CMU CS1** **Unit 1 – Creating Drawings** -basic -colours -shapes	**-Python Programming CMU CS1** **Unit 2 – Functions, Mouse Events, Properties** -functions -mouse events -properties	**-Python Programming CMU CS1** **Unit 3 – Mouse Motion Events, Conditionals, Helper Functions** -mouse motions events -conditionals -helper functions	**-Python Programming CMU CS1** **Unit 4 – More Conditionals. Key Events, Methods** -more conditionals (if-elif-else) -key events -methods	**-Python Programming CMU CS1** **Unit 5 – Complex Conditionals and More Key Events** -complex conditionals -more key events	**-Python Programming CMU CS1** **Unit 6 – Groups, Step Events, Motion** -groups -group methods -step events and motion -special types of motion	**-Python Programming CMU CS1** **Unit 7 – New Shapes, Local Variables, For Loops, Looping Through Groups** -basic -colours -shapes	**-Python Programming CMU CS1** **Unit 8 – Math Functions, Random Values, and Nested Loops** -math functions -random values -nested For Loops	**-Python Programming CMU CS1** **Unit 9 – Types, Strings, and While Loops (Optional)** -types and inputs -strings -string methods -While Loops	**-Python Programming CMU CS1** **Unit 10 – Lists and Return Values** -types and inputs -strings -string methods -While Loops	**-Python Programming CMU CS1** **Unit 11 – 2D Lists and Board Games** -2D lists -board games	**-Python Programming CMU CS1** **Unit 12 – Final Project** -final create project

Course	Week	Week	Week	Week	Week	Week	Week	Week	Week
ICT 11 Programming Pre-AP **ICTP 11 (4 credits)**	-Classroom Expectations & Procedures -File Management -SMART Goals -set up Google Classroom and other LMS	**- Python Programming Unit** -Bad Choices: An Illustrated Introduction to Computational Thinking -Algorithm Framework - Programming Constructs (flowcharts and pseudocode) (sequencing, selection, iteration, combination)	**-Python Chatbots Programming** **Unit 0 – What is Computer Science?** This unit introduces Computing Science as the design of algorithms and their implementation using code. It also covers the history of Computing Science, including the first programmer. ● problem solving, algorithms, comments and output	**Unit 1 - Chatbots with Personality** The theme of this unit is "Chatbots", and students are introduced to the idea of the Turing Test and the field of natural language processing. ● variables, naming, concatenation, input (first example only), lists, modules ● conditionals (if/else, if/elif, if only), Boolean statements, logical operators (or/and) ● attributes of good software and clean code, code reviews, software engineering ● syntax and semantic error types, string methods (except 9.5.1), in keyword for lists and strings ● function chaining, for loops, break keyword, creating lists from input, integer datatype, addition	**Unit 2 – Recommendation Systems** The theme of this unit is "You might also like" and introduces students to recommendation systems such as those used in Netflix and Amazon. ● arithmetic operators, loops with range, string/int/float data types, accumulators ● type conversion, list length ● working with data files, reading files, splitting strings into lists, indexing/accessing elements in lists, comparison operators, operator precedence ● comparing two lists, measuring similarity, nested loops (files and lists)	**Unit 3 – Under the Hood** This unit explains to students how their code is eventually translated into binary representations. ● binary, ASCII, bytes, Unicode, string indexing, loops over strings, exponents	**Unit 4 – Graphics and Animation** The theme of this unit is "Graphics and Animation" and reminds students that many 3D games and movies are built with code. Recursion is taught here using a visual recursive tree example, and functions are taught in order to draw pictures with various parameters. ● Turtle module, functions and parameters, functions with Turtle, local variables, more Turtle methods, Turtle tutorials (all pages), while loop, while for input ● Recursion, three laws of recursion, dictionaries ● Fruitful functions, sentinel values and validating input, logical operator (not)	**Unit 5 – Computer Vision** This unit introduces the field of Computer Vision. It reveals the RGB representations of pixels and provides a first experience with image processing. As image processing can be slow, it also motivates the need for writing efficient code. ● modules, from... import, 2D arrays, nested loops (images), RGB pixel representations, image processing, tuples, tuples as return values, PIL image module ● making your own module, understanding Python documentation ● appending/concatenating to lists, string formatting for output ● problem solving strategies: break down large problems into smaller ones, test small pieces of code first ● efficiency, using the time module to profile your code	**Unit 6 – Internet and Big Data** This unit emphasizes the recent need to process large amounts of data, thanks to the Internet. Students use large datasets pulled from the Internet to learn searching and sorting algorithms, as well as their complexities. ● list/string indexing from the end, linear search, sorting, selection sort, introduction to complexity, ● swapping elements, range sub listing (begin/end), list slicing, Binary Search, Algorithm analysis, Big-O Notation (n, n2, logn), Analysis of: Linear Search, Selection Sort, Binary Search ● recursive summation, recursive Fibonacci, binary search analysis proof, remainder (modulo), converting base using fruitful recursion, Python beyond the browser

Course	Week	Week	Week	Week	Week	Week	Week	Week	Week	Week
ICT 12 Programming **ICTP 12 (4 credits)**	-Classroom Expectations & Procedures -File Management -SMART Goals -set up Google Classroom and other LMS	**- Python Programming Unit APCSP CMU Academy** -Bad Choices: An Illustrated Introduction to Computational Thinking -Algorithm Framework - Programming constructs (flowcharts and pseudocode) (sequencing, selection, iteration, combination)	**CMU Academy Unit 1 – Intro to CMU Graphics** Learn fundamental coding techniques by drawing images in Python.	**CMU Academy Unit 2 – Functions, Mouse Events, Conditionals** Expand your programming abilities by introducing events and coding techniques to behave differently based on input.	**CMU Academy Unit 3 – Groups, Lists, and Loops** Build programs that implement iterating over loops and group objects and data.	**CMU Academy Unit 4 – Complex Conditionals, More Events, and Libraries** Learn how to manage complex inputs and events, as well as simplify your code by using pre-written code from libraries.	**CMU Academy Unit 5 – Create Performance Task** Practice and complete the Create Performance Task (PT).			

Course	Week	Week	Week	Week	Week	Week	Week	Week	Week	Week
AP Computer Science Principles 12 (Blended) **APCSP 12 (4 credits)**	**Unit 1 - Digital Information (3 weeks):** Students explore the way computers store and represent complex information like numbers, text, images, and sound. **Major Assessment Project:** The unit project asks students to consider and debate issues that arise in modern society due to the digitizing of information. **Unit 2 – The Internet (3 weeks)** Students learn how the Internet works and discuss its impacts on politics, culture, and the economy. **Major**	**Unit 3 – Intro to App Design (3 weeks)** Students design their first app while learning both fundamental programming concepts and collaborative software development processes. . **Major Assessment Project:** The unit project asks students to collaborate with a classmate to design an app that can teach others about a topic of shared interest. Students practice interviewing classmates to identify the goals of the project, mockup designs, collaboratively program the app, and run simple user tests. The app itself must include at least three	**Unit 4 – Variables, Conditionals, and Functions (3 weeks)** Students expand the types of apps they can create as they learn how to store information (variables), make decisions (conditionals), and better organize code (functions). **Major Assessment Project:** The unit project asks students to design an app that makes a recommendation based on input information from the user. Students are given a great deal of freedom to choose their topic, design their user interface, and decide how to actually program their app's behavior. **Unit 5 – Data (2**	**Unit 6 - Lists, Loops, and Traversals (4 weeks)** Students learn to build apps that use, and process lists of information. Like the previous unit, students learn the core concepts of lists, loops, and traversals through a series of EIPM lesson sequences. **Major Assessment Project:** The unit project asks students to spend five days as part of a "Hackathon" project that they have nearly complete independence to scope and design. Students must choose one dataset from the Data Library in AppLab to be a component of their project to demonstrate what	**Unit 7 – Parameters, Return, and Libraries (3 weeks)** Students learn how to design clean and reusable code that can be shared with a single classmate or the entire world. **Major Assessment Project:** The unit project asks students to design a library of functions that they can share with classmates. Their library must contain at least two functions and at least one of those functions must include a parameter, return, a loop, and an if-statement.	**Unit 8 – Cybersecurity and Global Impact (2 weeks)** Students research and debate current events at the intersection of data, public policy, law, ethics, and societal impact. **Major Assessment Project:** Students complete the "future school" simulation throughout this unit. Working in teams of roughly five people, students are assigned a role and a set of interests that they'll need to investigate. They research real-world innovations that could improve schools and align	**Unit 9 – Create PT (4 weeks)** This short unit prepares students to complete the AP® Create Performance Task (PT). Students will have learned the skills and concepts necessary to complete the task in previous units and will even have seen components of the task itself. **Major Assessment Project:** The unit project asks students to complete the Create PT in 12 hours. This is worth 30% of the total score.	**Unit 10 – Algorithms (3 weeks)** Students learn to design and analyze algorithms to understand how they work and why some algorithms are considered more efficient than others. This short unit is entirely unplugged and features hands-on activities that help students get an intuitive sense of how quickly different algorithms run and the pros and cons of different algorithms. **Major Assessment Project:** Students will complete an end-of-unit assessment that is aligned with CS Principles	**Exam Prep – (2 weeks)** The exam has a time limit of 120 minutes. There are 70 multiple choice questions. 57 of these questions are single select, 5 are single select with a reading about a computing innovation, and 8 are multiple select. This is worth 70% of the total score. Students will receive a final exam score of 1–5, derived from their performance on both the through-course assessment and the end-of-course exam. AP Score 5 – Extremely Well Qualified 4 – Well Qualified 3 – Qualified 2 – Possibly Qualified	**CSP Post-AP – Databases and Using Data in your Apps (20 hours – 17 classes)** 5% to final term mark – complete / incomplete

	Assessment Project: The unit project asks students to design a policy position for an imaginary political candidate related to an "Internet Dilemma." Students must analyze news stories about their topic to identify impacted groups, explain those groups' interests, explain technical background about the dilemma, and then recommend a policy solution that the candidate should advocate for.	screens and demonstrate what students have learned about user interface design and event-driven programming.	**weeks)** Students explore and visualize datasets from a wide variety of topics as they hunt for patterns and try to learn more about the world around them from the data. **Major Assessment Project:** Students use the data visualizer to find and present a data story. Using what they've learned about the data analysis process, students either choose a dataset inside the data library, or upload one, of their own and create visualizations that find interesting patterns that possibly reveal new insights and knowledge.	they have learned about lists and list processing; otherwise, scoping the project is completely up to them.		with the interests of their character. Throughout the unit, they are given opportunities to refine their proposals as a team and debate the benefits and risks of different computing innovations.		framework objectives covered in this unit.	1 – No Recommendation	
the topics and assignments in the above courses are subject to change without notice at the discretion of the instructor depending on the availability of time, resources, and adequate facility										
school year 10 months - 43 weeks - 301 days **School Act - school year approximately 193 days with 186 days in session or 37 weeks or 952 hours of instruction for students in grades 8 to 12**										
semester system 19 weeks - 88 instructional days – 117 hours										

Course	Week	Week	Week	Week	Week	Week	Week	Week	Week	Week
Physical Education 8 Coed	-Expectations and Procedures -MYP Unit- -Team Games- End Zone Games- Rugby Football Ultimate	-MYP Unit- -Team Games- Goalie Games- Soccer Lacrosse Interdisciplinary Unit (ID) Baseline Testing - Fitness, Strength, and Conditioning	-MYP Unit- -Team Games- Goalie Games-continued Floor Hockey Floorball Ringette Sexual Health Education -The Circuit Sexual Health Stations -cybersafe -sexual exploitation and healthy relationships	-MYP Unit- -Dance- Solo and Partner -Dance Forms- Polka Dancing Square Dancing Jive Cha Cha Mexican Hat Dance Electric Slide Macarena Bird Dance Limbo Line Dancing	-MYP Unit- -Team Games- Basketball Kabbadi Tchoukball Quidditch ACT High School CPR and AED and Opioid Overdose Response Training Program	-MYP Unit- -Movement- Wrestling Open Parachute Mental Health- -New Hurdles - Starting High School -Peer Inclusion and Exclusion -Understanding Mental Health -Independent Thinking and Healthy Decision Making	-MYP Unit- -Weight Room / Active Living- -Weight room Introduction -Weight room Safety -Health Screening -Goal Setting -Strength and Conditioning Program Development	-MYP Unit- -Racket Games- Badminton Pickleball Tennis	-MYP Unit- -Non-Invasive Games- Volleyball Softball Stickball Cricket Disc Golf Post-Testing - Fitness, Strength, and Conditioning	-MYP Unit- -Minor Games- Capture the Flag Man Tracker Man Hunter Pirate Ship Hide and Seek Red Light Green Light Red Rover Duck Duck Goose British Bulldogs Star Wars GaGa Ball Juggling Kubb Game Speed Cup Stacking First People’s Games
	Timed Runs / Core Training Movement, Bones, Muscles, Flow Mobility Sequence, Mental Health, Healthy Sleeping and Eating, Gratitude PE Leadership Training What is a warmup? How do I learn a new skill? How do I improve my locomotor movement?									
Physical Education 9 Coed	-Expectations and Procedures -MYP Unit- -Team Games- End Zone Games- Rugby Football Ultimate	-MYP Unit- -Team Games- Goalie Games- Soccer Lacrosse Baseline Testing - Fitness, Strength, and	-MYP Unit- -Team Games- Goalie Games-continued Floor Hockey Floorball Ringette	-MYP Unit- -Dance- Solo and Partner -Dance Forms- Polka Dancing Square Dancing Jive Cha Cha Mexican Hat	-MYP Unit- -Team Games- Basketball Kabbadi Tchoukball Quidditch ACT High School CPR and AED	-MYP Unit- -Movement- Wrestling Open Parachute Mental Health- -Boundaries and Consent	-MYP Unit- -Weight Room / Active Living- -Weight room Introduction -Weight room Safety -Health Screening	-MYP Unit- -Racket Sports- Badminton Pickleball Tennis	-MYP Unit- -Non-Invasive Sports- Volleyball Softball Stickball Cricket Disc Golf	-MYP Unit- -Minor Games- Capture the Flag Man Tracker Man Hunter Pirate Ship Hide and Seek Red Light Green Light

		Conditioning		Dance Electric Slide Macarena Bird Dance Limbo Line Dancing	and Opioid Overdose Response Training Program	-Self Image and Stereotypes -Eating and Body Image Challenges -Objectification and Respect	-Goal Setting -Strength and Conditioning Program Development		Interdisciplinary Unit (ID) Post-Testing - Fitness, Strength, and Conditioning	Red Rover Duck Duck Goose British Bulldogs Star Wars GaGa Ball Juggling Kubb Game Speed Cup Stacking First People's Games
	Timed Runs / Core Training Movement, Bones, Muscles, Flow Mobility Sequence, Mental Health, Healthy Sleeping and Eating, Gratitude PE Leadership Training What is a warmup? How do I learn a new skill? How do I improve my locomotor movement?									