

Advanced Placement Computer Science A



Dominican International School

Course Syllabus

School Year 2026–27



Mr. Tim Sampson
Teacher

Email: tsampson@dishs.tp.edu.tw

Course Description

AP Computer Science A is an introductory college-level computer science course equivalent to a one-semester college course. Students learn to design, implement, analyze, test, and document programs using the Java programming language. The course develops computational thinking through object-oriented programming, algorithm development, class design, and the use of data collections.

The current College Board framework is organized around four areas: **Using Objects and Methods, Selection and Iteration, Class Creation, and Data Collections**. This course follows the CSAwesome2 curriculum aligned with the current AP Computer Science A framework and incorporates hands-on programming labs, AP Classroom progress checks, free-response practice, and unit assessments. For additional information, see the [College Board AP Computer Science A Course and Exam Description](#). ([College Board course overview](#))

Requirements

Students should have successfully completed a high school algebra course and be comfortable with mathematical reasoning, problem-solving, functions, and function notation. Students should also possess the reading and communication skills necessary to interpret technical problems, explain algorithms, and document their solutions.

Previous programming experience is helpful but not required. Students should be prepared to approach challenging problems systematically, learn from errors, revise their work, and devote regular time to programming practice. ([College Board prerequisites](#))

Course Objectives

The course emphasizes object-oriented problem-solving and program design using Java. Students develop solutions by breaking complex problems into manageable parts, selecting appropriate algorithms and data structures, and testing and refining their code.

By the end of the course, students will be able to:

- Design, implement, test, and document Java programs that solve problems.
- Use primitive data types, variables, expressions, assignment statements, and type conversion.
- Create and use objects, methods, parameters, return values, strings, APIs, and libraries.
- Develop algorithms using Boolean expressions, conditional statements, loops, and nested control structures.
- Design classes using instance variables, constructors, methods, parameters, scope, and encapsulation.
- Apply abstraction and modularity to organize programs and manage complexity.
- Store, traverse, modify, and analyze data using arrays, ArrayList objects, and two-dimensional arrays.
- Implement and evaluate common searching, sorting, and data-processing algorithms.
- Trace program execution, identify errors, compare possible solutions, and informally evaluate efficiency.
- Explain program behavior clearly through comments, documentation, written responses, and discussion.
- Consider privacy, data collection, ethical design, and the social implications of computing.
- Demonstrate the programming and analytical skills required for the AP Computer Science A Exam.

Primary Textbook and Other Resources

- Online access to programming labs, homework, e-text and supplemental videos:
 - <https://runestone.academy/ns/books/published/csawesome2/index.html>
- Our class site for discussion forums, collaboration, notes and assignments.
<https://classroom.google.com>
- Supplementary documents, activities, labs and examples:
 - [The Java™ Tutorials](#)
 - [AP CSA Labs](#) We will cover the classic labs, Magpi, PixLab and Elevens. As time allows, the new labs Steganography, Celebrity and Data and other labs that might be available in the upcoming semester.
 - Physical Copy ~ 5 Steps to a 5: AP Computer Science A 2018 2nd Edition by Dean R. Johnson , Carol A. Paymer , Aaron P. Chamberlain [5 Steps to a 5](#)
- AP Classroom AP Checks, Unit Reviews and Videos. <https://apclassroom.collegeboard.org/>

Lab Component

The course includes a structured-lab component in which students will complete a minimum of 20 hours of hands-on lab experiences. The curriculum has small coding assignments called Programming Challenges in each lesson, which they are encouraged to do using pair programming. In addition, students will complete at least three of the following College Board AP Computer Science A labs, as chosen by their teachers to complete a minimum of 20 hours:

- ❑ [MagPie Chatbot Lab](#) (built into the CSA curriculum)
- ❑ [Picture Lab](#) (built into the CSA curriculum)

- ❑ [Consumer Review Lab](#) (built into the CSA curriculum)
- ❑ [Celebrity Lab](#)
- ❑ [Steganography Lab](#)
- ❑ [Data Lab](#)

AP CS A Exam

Students who complete this course will be prepared to take the AP CS A Exam in May.

Assessment

- Assignments 30%
- Quizzes 30%
- Quarter Exam 30%
- Department 10 %

Additional Information

Submission Alignment with the TORCH Values

In this course, earning a top score, such as 9 or 10 out of 10, depends on the quality of your learning process as well as your final product.

Outstanding performance does not mean being perfect every day. It means consistently showing strong effort, responsibility, growth, and the values represented by our school's D'TORCH framework.

- **Truthful:** Your work is authentically yours. You provide evidence of the steps you followed, accurately represent your progress, and are honest about where you need help.
- **Organized:** You manage your time effectively, bring the required materials, follow instructions, and meet deadlines. When additional time is genuinely needed, you communicate proactively and request an extension before the due date.
- **Reflective:** You explain your thinking clearly using TEEL or another suitable structure. When asked about your work, you can articulate how you reached an answer, not simply state the answer itself, and use feedback to improve.
- **Courageous:** You take intellectual risks, contribute to discussions, attempt challenging tasks, and accept mistakes as a necessary part of learning.
- **Helpful:** You support your classmates appropriately and contribute to a respectful, positive, and collaborative learning environment.

A first-rate submission shows not only what you produced, but also how responsibly, thoughtfully, and purposefully you approached the learning process

Class Expectations

- Come to class on time and be prepared
- Have a positive attitude and be willing to learn.
- Respect yourself, others, and our school.

- Always complete your work and try your best.
- Actively participate, listen carefully, but don't speak out of turn.
- All assignments must be completed.

Homework and Quiz Rules

- All assignments must be turned in on the day they are due unless there is a valid reason.
- 1 day late = Maximum of only 80%
- 2+ days late = Project-I & Maximum of only 60%
- If a student has been absent, it is his/her duty to find out what work is due, and hand it in a day later.
- All assignments must satisfactorily be completed or Project I will be assigned.
- If you are absent on the day of the quiz, you will only be able to get a maximum of 60%.

Classroom Rules

- All students are expected to follow the rules. Consequences will follow if rules are broken.
- Read and follow the standard school rules.
- Be on time and neatly dressed, in full school uniform.
- Speak in ENGLISH ONLY.
- Respect your teachers, fellow students and their property.
- Keep your seating space and classroom clean and neat.
- No eating or drinking in the ICT Labs.
- Ask permission to leave the class.

Assessment

- Assignments 30%
- Quizzes 30%
- Quarter Exam 30%
- Deportment 10 %

Academic Honesty (Plagiarism)

Academic Dishonesty means employing a method or technique or engaging in conduct in an academic endeavor that contravenes the standards of ethical integrity expected at DIS. Academic dishonesty includes but is not limited to, the following:

1. Purposely incorporating the ideas, words of sentences, paragraphs, or parts thereof without appropriate acknowledgment and representing the product as one's own work; and
1. Representing another's intellectual work such as photographs, paintings, drawings, sculpture, or research or the like as one's own, including failure to attribute content to an AI.
2. Employing a tutor, making use of Artificial Intelligence without acknowledgement, getting a parent to write a paper or do an assignment, paying for an essay to be written by someone else and presented as the student's own work.
3. Committing any act that a reasonable person would conclude, when informed of the evidence, to be a dishonest means of obtaining or attempting to obtain credit for academic work.

Any act of academic dishonesty will result in an automatic zero on the entire assignment and trigger disciplinary actions.

Discipline

- Verbal warning
- Write-Up, entered into the discipline system and then referral to the Discipline Office.
- Parent-Teacher conference as required.

WEEKLY SCHEDULE

First Quarter Tentative Course Content

Week & Date	Topics / Projects / Assessments
Week 1 Aug 10 - 14	Monday Class Preparation ~ Student Holiday Tuesday Morning Assembly Friday Opening Mass & Assumption of Our Lady 8:00 Welcome to APCSA 1.1 Introduction to Algorithms, Programming, and Compilers 1.2 Variables & Data Types 1.3 Expressions and Output 1.4 Assignment Statements and Input Weekly CB Lab: Unit 1 Receipt: Activity 1 — Basic Receipt
Week 2 Aug 17 - 21	1.5 Casting and Range of Variables 1.6 Compound Assignment Operators 1.7 API and Libraries 1.8 Documentation with Comments 1.9 Method Signatures Weekly CB Lab: Unit 1 Receipt: Activity 2 — Enhanced Receipt
Week 3 Aug 24 - 28	CB Quiz Unit 1 MCQ Part A Q1 Quiz 1 Unit 1.1 to 1.9 1.10 Calling Class Methods 1.11 Using the Math Class 1.12 Objects Weekly CB Lab: Unit 1 Receipt: Activity 3 — Including Random
Week 4 Aug 31 - Sep 4	1.13 Object Creation and Storage (Instantiation) 1.14 Calling Instance Methods 1.15 Strings CB Quiz Unit 1 MCQ Part B Practice Test on Objects Weekly CB Lab: Unit 1 Receipt: Activity 4 — Interactive Receipt

Week 5 Sep 7 - 11	CB Quiz Unit 1 FRQ Part C Q1 Quiz 2 Unit 1.1 to 1.15 2.1 Algorithms with Selection and Repetition 2.2 Boolean Expressions Weekly CB Lab: Unit 1 Receipt: Activity 5 — Design a Receipt
Week 6 Sep 14 - 18	2.3 If Statements 2.4 Nested If Statements 2.5 Compound Bool Expressions Q1 Quiz 3 Unit 2.1 to 2.8 CB Quiz Unit 2 MCQ Part A Weekly CB Lab: Magpie Chatbot Lab Activity 1 and 2
Week 7 Sep 21 - 25	CB Quiz Unit 2 FRQ Part A 2.6 Comparing Bool Expressions 2.7 While Loops Q1 Final Exam Units 1 to 2. Weekly CB Lab: Magpie Chatbot Lab Activity 3 T-W-Th Pre-Exam Days Friday Moon Festival ~ Public Holiday
Week 8 Sep 28 - Oct 2	Monday Confucius' Birthday ~ Public Holiday 2.8 For Loops 2.9 Implementing Selection and Iteration Algorithms Thursday Major Exams Day 1 Friday Major Exams Day 2
Week 9 Oct 5 - 9	Monday Record Day ~ Student Holiday T-W-Th Teachers Conference ~ Student Holiday Friday Double 10 Day ~ Public Holiday

Second Quarter Tentative Course Content

Week & Date	Topics / Projects / Assessments
Week 1 (10) Oct 12 - 16	2.10 Implementing String Algorithms 2.11 Nested Iteration 2.12 Informal Runtime Analysis of Loops 2.1-2.6 Review (2 periods) → 2.13 Unit 2 Part 1 Summary on Selection → 2.14 Selection Toggle Code Practice → 2.15 Selection Coding Practice → 2.16 Selection Multiple Choice Exercises

	→ 2.17 Selection Free Response Question (FRQ) Game Practice → 2.18 FRQ Style Selection Coding Practice Weekly CB Lab: Magpie Chatbot Lab Activity 3
Week 2 (11) Oct 19 - 23	2.19 Group Work - Remainder Operator (%) 2.7-2.12 Review (2 periods) → 2.20 Unit 2 Part 2 Summary on Loops (2.7-2.12) → 2.21 Loops Toggle Code Practice (2.7-2.12) → 2.22 Loops Coding Practice (2.7-2.12) → 2.23 Loops Multiple Choice Exercises (2.7-2.12) Q2 Quiz 1 Unit 2.1 to 2.6 Weekly CB Lab: Magpie Chatbot Lab Activity 4
Week 3 (12) Oct 26 - 30	Monday Retrocession Day ~ Public Holiday CB Quiz Unit 2 MCQ Part B CB Quiz Unit 2 FRQ Part B 2.24 Unit 2 Test FRQ Practice Strings → 2.25 FRQ 1 MessageBuilder with Loops and Strings → 2.26 FRQ APCalendar for Loops and Control Structures → 2.27 FRQ Self Divisor A → 2.28 FRQ String Scramble A
Week 4 (13) Nov 2 - 6	Q2 Quiz 2 Unit 2.1 to 2.12 3.1 Abstraction and Program Design 3.2 Impact of Program Design 3.3 Anatomy of a Java Class (2P) Weekly CB Lab: Virtual Pet: Activity 1 and 2
Week 5 (14) Nov 9 - 13	3.4 Writing Constructors 3.5 Methods: How to Write Them (2P) 3.6 Methods: Passing and Returning References of an Object (2P) Weekly CB Lab: Virtual Pet: Activity 3 A
Week 6 (15) Nov 16 - 20	Q2 Quiz 3 Unit 3.1 to 3.6 3.7 Class (static) Variables and Methods 3.8 Scope and Access 3.9 this Keyword Weekly CB Lab: Virtual Pet: Activity 3 B, & C
Week 7 (16) Nov 23 - 27	3.10 Unit 3 Summary 3.11 Unit 3 Mixed Up Code Practice 3.12 Unit 3 Write Code Practice for Classes Q2 Final Exam Units 1 to 3 T-W-Th Pre-Exam Days

	Friday Grade 12 Exams
Week 8 (17) Nov 30 - Dec 4	Monday Grade 12 Exams 3.13 Multiple-Choice Exercises - Objects and Classes 3.14 Unit 3 Test 3.15 Free Response Question (FRQ) 2 Class Design - CupcakeMachine Class Weekly CB Lab: Virtual Pet: Activity 4 A, B, & C
Week 9 (18) Dec 7 - Dec 11	Monday TBD Tuesday Foundation Day Mass 9:00, Cutting of the Cake Ceremony, Christmas Fair, (Whole day) Wednesday TBD Thursday Major Final Exams half day Friday Major Final Exams half day
<i>Dec 14 to Jan 1 Christmas Break</i>	

Third Quarter Tentative Course Content

Week & Date	Topics / Projects / Assessments
Week 1 (19) Jan 4 - Jan 8	Monday Record Day ~ Student Holiday 3.16 FRQ 2 Practice with Classes → 3.17 Free Response Question (FRQ) - StepTracker Class → 3.18 Free Response Question - Time CB Quiz Unit 3 MCQ Part A CB Quiz Unit 3 MCQ Part B CB Quiz Unit 3 FRQ Part B Weekly CB Lab: Virtual Pet: Activity 5 — Adding Features Friday New Year Mass at 8:00
Week 2 (20) Jan 11 - Jan 15	3.20 Design a Class for your Community Project (2P) Q3 Quiz 1 Unit 3.1 to 3.9 4.1 Ethical and Social Issues Around Data Collection 4.2 Data Sets 4.3 Array Creation and Access
Week 3 (21) Jan 18 - Jan 22	4.4 Array Traversals 4.5 Implementing Array Algorithms CB Quiz Unit 4 MCQ Part A CB Quiz Unit 4 FRQ Part A Q3 Quiz 2 Unit 4.1 to 4.5 Weekly CB Lab: Magpie Chatbot Lab Activity 5

Week 4 (22) Jan 25 - Jan 29	4.6 Using Text Files 4.7 Wrapper Classes - Integer and Double 4.8 ArrayList and its Methods 4.9 ArrayList Traversals 4.10 Implementing ArrayList Algorithms
Week 5 (23) Feb 1 - Feb 5	CB Quiz Unit 4 MCQ Part B CB Quiz Unit 4 FRQ Part B Q3 Quiz 2 Unit 4.1 to 4.10 Thursday Little New Year's Eve ~ Public Holiday Friday Lunar New Year ~ Public Holiday
February 8 to 12 Chinese New Year Holiday	
Week 6 (24) Feb 15 - Feb 19	4.11 2D Array Creation and Access 4.12 2D Array Traversals: Nested Loops 4.13 Implementing 2D Array Algorithms
Week 7 (25) Feb 22 - Feb 26	22-26 IOWA Assessments 4.14 Searching Algorithms 4.15 Sorting Algorithms 4.16 Recursion 4.17 Recursive Searching and Sorting 4.18 Arrays Summary
Week 8 (26) Mar 1 - Mar 5	Monday Memorial Day ~ Public Holiday Arrays Review → 4.19 Toggle Code Practice 4a Arrays (4.1-4.6) → 4.20 Arrays Code Practice → 4.21 More Code Practice with Arrays → 4.22 Arrays Multiple-Choice Exercises 4.23 Practice Exam for Arrays FRQ Practice Arrays → 4.24 Free Response - Horse Barn A → 4.25 Free Response - Horse Barn B → 4.26 Free Response - Self Divisor B → 4.27 Free Response - Sound A → 4.28 Free Response - Sound B T-W-Th Pre-Exam Days
Week 9 (27) Mar 8 - Mar 12	FRQ Practice Arrays → 4.29 Free Response - Number Cube A → 4.30 Free Response - Number Cube B CB Quiz Unit 4 MCQ Part C CB Quiz Unit 4 FRQ Part C ArrayList Practice → 4.31 ArrayList Summary

	→ 4.32 Toggle Code Practice 4b ArrayList (4.7-4.10) → 4.33 Code Practice with ArrayLists → 4.34 ArrayList Multiple-Choice Exercises
Week 10 (28) Mar 15 - Mar 19	FRQ Practice ArrayLists → 4.35 Free Response Question (FRQ) 3 - ItemInventory → 4.36 Free Response - Climbing Club A → 4.37 Free Response - Climbing Club B → 4.38 Free Response - Climbing Club C FRQ Practice ArrayLists → 4.39 Free Response - CookieOrder A → 4.40 Free Response - CookieOrder B → 4.41 Free Response - StringFormatter A → 4.42 Free Response - StringFormatter B FRQ Practice ArrayLists → 4.43 Free Response - Delimiters A → 4.44 Free Response - Delimiters B 4.45 Search/Sort Multiple-Choice Exercises Q3 Final Exam Units 1 to 4
Week 11 (29) Mar 22- Mar 26	Tuesday Major Final Exams half day Wednesday Major Final Exams half day Thurs - Fri Easter Break (Spring Break) ~ Public Holiday
<i>March 25 to April 2 Easter & Spring Break</i>	

Fourth Quarter Tentative Course Content

Week & Date	Topics / Projects / Assessments
Week 1 (30) Apr 5 - Apr 9	Monday Tomb Sweeping Day ~ Public Holiday Tuesday Children's Day ~ Public Holiday Q4 Quiz 1 AP MCQ Practice Test 4.46 2D Arrays Summary 4.47 Toggle Code Practice 4c 2D Arrays (4.11-4.13) 4.48 Code Practice with 2D Arrays 4.49 More Code Practice with 2D Arrays 4.50 2D Arrays Multiple-Choice Exercises
Week 2 (31) Apr 12 - Apr 16	AP FRQ Practice Test A FRQ Practice Two-dimensional Arrays → 4.51 Free Response (FRQ) 4 → 4.52 Free Response - Gray Image A → 4.53 Free Response - Gray Image B → 4.54 FRQ Route Cipher A and B

	4.55 Recursion Summary 4.56 Recursion Mixed Up Code Practice 4.57 Recursion More Mixed Up Code Practice 4.58 Code Practice for Recursion AP FRQ Practice Test B
Week 3 (32) Apr 19 - Apr 23	19 - 23 AP Mock Exams Once the schedule is set for students and their schedules with other AP Classes are understood, the AP review schedule will be shared for the remaining classes.
Week 4 (33) Apr 26 - Apr 30	T-W-Th Pre-Exam Days Friday Labor Day ~ Public Holiday
Week 5 (34) May 3 - May 7	3 - 7 Final Exams (K, Gr. 5, 8, & 12 Only) 3 - 14 AP Exams
Week 6 (35) May 10 - May 14	3 - 14 AP Exams Wednesday Major Final Exams Undergraduates half day Thursday Major Final Exams Undergraduates half day Friday Record Day ~ Student Holiday
Week 7 (36) May 17 - May 21	Monday Gr. 12 Recollection (Morning) Monday Baccalaureate Mass (Afternoon, Schoolwide) Wednesday House Sports Day Thursday High School Field Trip Friday House Culminating Activity All Week Student Clearance Days & Special Events
Week 8 (37) May 24 - May 28	Monday Middle & High School Sports Day Monday LS Field Trip Tuesday Kindergarten Graduation/Gr. 5 Promotion Wednesday Gr. 8 Graduation and Gr. 12 Graduation Thursday Pre-Kindergarten & Gr. 1 - 4, 6 & 7, 9-11 Recognition Friday Last Day of School for Students

“Kindling the Light of Faith, Hope, and Love: The Legacy of St. Dominic de Guzmán”

High School Computer Science Sequence 2026–2027

The table below outlines the High School Computer Science pathways available at DIST.

Classes are 40 minutes long, and most students should be able to complete the core lesson activities during class. Students who need additional practice, review, or support may occasionally require about 15-25 minutes outside class to satisfactorily complete the activities.

Students who wish to develop deeper mastery may choose to spend more time on enrichment activities included within or added to the lessons. These activities are not required for the course

grade, but they provide valuable opportunities for additional practice, self-assessment, and further exploration.

A student's effort and engagement will strongly influence their growth in computer science. Although completing enrichment activities does not automatically result in a higher grade, it can help students build stronger skills, greater confidence, and a deeper understanding of the course content.

Subject CS Sequence	
Grade 9 CS AI & Computing	Grade 10 CS Discoveries: Design & Data
CodeAI: AI Discoveries (AID) Unit 1 - Thinking Critically About AI Unit 2 - AI and Machine Learning Unit 4b - Interactive Animations and Games	CodeAI: AI Discoveries (AID) Unit 5 - The Design Process Unit 6 - Data and Society
G11 CS Exploring AI Foundations	G12 CS Building AI Foundations
CodeAI: AI Foundations Unit 1: Problem Solving with AI Unit 2: AI and the Systems That Power It Unit 5: Insights from Data and AI	CodeAI: AI Foundations Unit 6: AI-Generated Design Unit 7: AI and Algorithmic Decisions Unit 8: Building Data-Driven Systems with AI

AP CS Sequence	
G11 APCS A JAVA	G12 APCS Principles
CSAwesome AP Computer Science A is an introductory college-level computer science course. Students cultivate their understanding of coding through analyzing, writing, and testing code as they explore concepts like modularity, variables, and control structures. Unit 1: Using Objects and Methods Unit 2: Selection and Iteration Unit 3: Class Creation Unit 4: Data Collections Unit 5: Review & AP Exam Preparation	CS5oAP , CS5oPython, CodeAI: AI Foundations (AIF) AP Computer Science Principles introduces students to the breadth of the field of computer science. In this course, students will learn to design and evaluate solutions and to apply computer science to solve problems through the development of algorithms and programs. Unit 1: The impact of Computing Unit 2: Programming in Python Unit 3: Create Task Unit 4: SQL and Data Science Unit 5: Application Development Unit 6: The Fabric of the Internet and AI Unit 7: Cybersecurity, AI-Powered Threats and Defenses Unit 8: Final Project Unit 9: Review & AP Exam Preparation

High School CS Curriculum Overview

CS5oAP as the AP CS Capstone

For students in the AP CS Pathway, CS5oAP serves as the Grade 12 capstone following AP Computer Science A in Grade 11. It builds on students' Java programming experience while broadening their understanding of computer science through Python, databases, web development, cybersecurity, artificial intelligence, and the social impact of computing.

Beginning in 2026–2027, C was removed from the AP Computer Science Principles course. This provides more time for other parts of the curriculum and allows students to begin with Python at a higher level of abstraction. Students can therefore spend more time designing, building, and refining realistic applications rather than managing lower-level programming details.

Practical Application through Service Learning

Students in both the Subject CS Pathway and AP CS Pathway are encouraged to apply their knowledge through the school's Service Learning program.

The High School Computer Science department collaborates with Service Learning staff to help students identify opportunities to:

- Apply computer science skills in authentic community contexts

- Develop solutions for real users and needs
- Gain practical experience beyond the classroom
- Strengthen collaboration, communication, and project-management skills
- Understand how technology can be used to address real-world challenges

These experiences complement classroom learning and provide meaningful opportunities for students to demonstrate the practical value of their computer science knowledge.

Curriculum Review and Development

The High School Computer Science curriculum is reviewed regularly to ensure that it remains current, appropriately challenging, and responsive to student needs, academic expectations, and changes in technology.

Curriculum Evaluation

Potential additions and revisions are evaluated through classroom trials, student outcomes, teacher reflection, and stakeholder feedback. Previous trials have included CS50 SQL and Carnegie Mellon University college-level programming materials.

These evaluations showed that the existing AP CS Pathway provides substantial breadth, depth, and academic rigor while maintaining flexibility to adapt to evolving technologies and student interests. This supports a focused curriculum that allows students sufficient time to develop, apply, and reflect on the skills necessary for further study, future careers, and real-world problem solving.

Stakeholder Feedback

Feedback is encouraged from students, parents, alumni, teachers, administrators, and industry professionals. This helps ensure that the curriculum remains aligned with academic standards, university pathways, and practical applications of computer science.

Adaptive Planning and Collaboration

The curriculum continues to evolve in response to new technologies, student interests, and stakeholder input. Collaboration with the Service Learning program provides a practical way to connect classroom learning with industry-related skills and community engagement without unnecessarily adding further course content.

Through ongoing evaluation and collaboration, the High School Computer Science program aims to remain coherent, relevant, and sustainable while preparing students for further study, future careers, and responsible participation in a technology-rich society.