International Baccalaureate Computer Science for teaching from Sept 2025

Course aims

The course enables students to:

1. develop conceptual understanding that allows connections to be made between different areas of the subject, and to other DP subjects
2. acquire and apply a body of knowledge, methods, tools and techniques that characterize computer science
3. analyse and evaluate solutions developed through computational thinking in a range of contexts
4. approach unfamiliar situations with creativity and resilience
5. use computational thinking to design and implement solutions to local and global problems
6. develop an appreciation of the possibilities and limitations of computer science
7. evaluate the impact of emerging technologies in computer science
8. communicate and collaborate effectively
9. develop awareness of the environmental, economic, cultural and social impact of computer science, its applications and ethical implications.

The syllabus

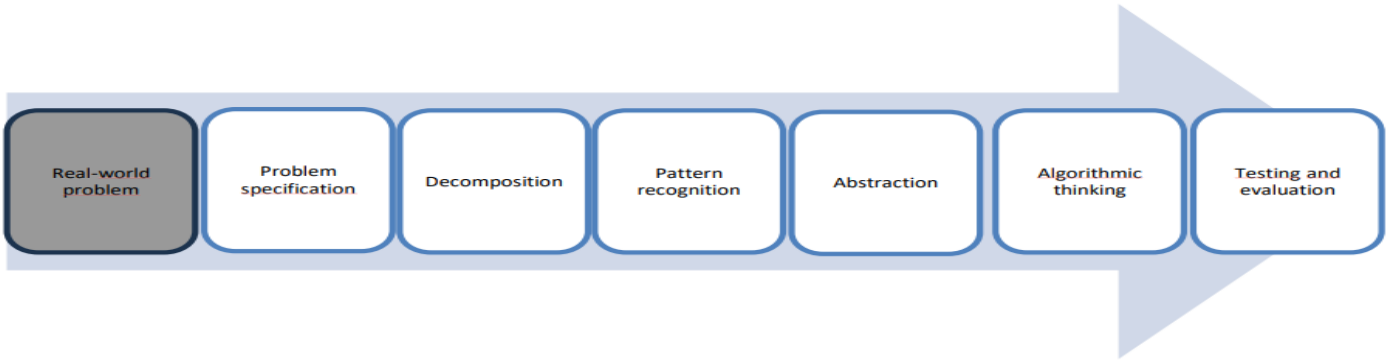
The Computer Science course is the study of computational systems, and the use and programming of computers to solve real world problems. It studies the theory, design, and architecture, of each of these.

Theme A: Concepts of computer science	Theme B: Computational thinking and problem-solving
A1 Computer fundamentals	B1 Computational thinking
A2 Networks	B2 Programming
A3 Databases	B3 Object Oriented Programming (OOP)
A4 Machine learning	B4 Abstract data types (ADT)—HL only

Theme B can be studied in either Python or Java. Students' understanding of emerging technologies will be assessed through a case study

Curriculum framework

The concept of computational thinking frames the DP computer science course. The diagram below illustrates one model for computational thinking which students will follow closely in their internal assessment.



This framework encourages students to explore how skills and knowledge can be used to solve real-world problems. Skills in computer science are understood as practical and personal and relate to the approaches to learning.

Assessment

Component	Content	**HL:** Length	Value	**SL:** Length	Value
Paper 1	Questions focused on the four topics in Theme A: Concepts of computer science and questions related to the case study	Four theme A questions Three case study questions 2 hours	40%	Four theme A questions Three case study questions 1 hour 15 minutes	40%
Paper 2	Questions on Theme B: Computational thinking and problem solving. Students can answers questions in either Java or Python	Questions on Theme B: Additional OOP questions 2 hours	40%	Questions on Theme B: 1 hour 15 minutes	30%
Internal Assessment Computational solution	Students develop a computational solution to a real-world problem of their choosing.	45 hours (approx..)	20%	45 hours (approx..)	30%

Paper 1

Assesses fundamental concepts of computer science, database development, networking, and machine learning. Section B focuses on evaluating knowledge and understanding of a pre-seen case study.

Paper 2

Evaluates computational thinking and the application of programming, both procedurally and using an object-oriented approach. Questions that require code will be presented both in Java and in Python, and students will be able to answer these questions using the coding language they have studied.

Internal assessment

The computational solution for SL and HL comprises the same task. This assessment focuses on computational thinking as a process. Students are asked to use this process to solve a problem of their choosing. The assessment criteria reflect the computational thinking process, as shown in table 3. The criteria also recognize the role that generative artificial intelligence can play in coding algorithms. Students are assessed not simply on the quality of the code they produce, but on their ability to use computational thinking. A total of 30 marks are available for the assessment, and the full set of criteria descriptors will be published with the subject guide.

Internal assessment criteria

Criterion	Marks	Focus
A—Problem specification	4	Problem scenario, computational context and success criteria
B—Planning	4	Decomposition and planning (including timing)
C—System overview	6	System model, component algorithms and testing strategy
D—Development	12	Functionality, techniques to code algorithms and testing
E—Evaluation	4	Effectiveness, improvements

The internal assessment has a recommended 35 hours of class time. The maximum word count for the project is 2,000 words, excluding diagrams and code. The documentation submitted for assessment should be accompanied by a video with a maximum length of five minutes. This video should demonstrate the functionality of the product developed for the solution and show examples of the testing strategy.