Computer Science – Diploma programme subject outline – Group 4: Experimental sciences			
School name	International School of Lausanne	**School code**	001079
DP subject	Computer Science		
Level	✓ Higher Level ✓ Standard Level		

Course background

Computer science requires an understanding of the fundamental concepts of computational thinking as well as knowledge of how computers and other digital devices operate.

The Diploma Programme computer science course is engaging, accessible, inspiring and rigorous. It has the following characteristics.

- draws on a wide spectrum of knowledge
- enables and empowers innovation, exploration and the acquisition of further knowledge
- interacts with and influences cultures, society and how individuals and societies behave
- raises ethical issues
- is underpinned by computational thinking.

Computational thinking involves the ability to:

- think procedurally, logically, concurrently, abstractly, recursively and think ahead
- utilize an experimental and inquiry-based approach to problem-solving
- develop algorithms and express them clearly
- appreciate how theoretical and practical limitations affect the extent to which problems can be solved computationally.

During the course the student will develop computational solutions. This will involve the ability to:

- identify a problem or unanswered question
- design, prototype and test a proposed solution
- liaise with clients to evaluate the success of the proposed solution and make recommendations for future developments.

Computer science has links with subjects outside of group 4, notably information technology in a global society (ITGS), but it should be noted that there are clear differences between the subjects.

Assessment in IB Diploma Computer Science

HL assessment		**Weighting**
External Paper 1 (2 hr 10 min)	Two compulsory sections. • Section A (30 minutes 25 marks) consists of several compulsory short answer questions. • Section B (100 minutes 75 marks) consists of three compulsory structured questions.	40% (100 marks)
External Paper 2 (1 hr 20 min)	Linked to the option studied. Between three and seven compulsory questions. SL/HL core questions 45 marks HL extension questions 20 marks	20% (65 marks)
External Paper 3 (1 hr)	Four compulsory questions based on a pre-seen case study.	20% (30 marks)
Internal Assessment Solution (3o h)	The development of a computational solution including • a cover page that follows the prescribed format • a product • supporting documentation (word limit 2,000 words).	20% (34 marks)
Internal Assessment Group 4 project (10 h)	To be assessed using the criterion Personal skills.	Completion mandatory but zero weighting

SL assessment		**Weighting**
External Paper 1 (1 hr 30 min)	Two compulsory sections. • Section A (30 minutes 25 marks) consists of several compulsory short answer questions. • Section B (60 minutes 45 marks) consists of three compulsory structured questions.	45% (70 marks)
External Paper 2 (1 hr)	Linked to the option studied. Between two and five compulsory questions.	25% (45 marks)
Internal Assessment Solution (3o h)	The development of a computational solution including • a cover page that follows the prescribed format • a product • supporting documentation (word limit 2,000 words).	30% (34 marks)
Internal Assessment Group 4 project (10 h)	To be assessed using the criterion Personal skills.	Completion mandatory but zero weighting

Note: calculators are not permitted in any computer science examination.

Nature and timing of Internal Assessment

The Internal Assessment will involve the planning, developing and evaluating of a Computer Science solution to a scenario. The IA has five phases / criterion:

- Criterion A – Planning
- Criterion B – Solution overview
- Criterion C – Development
- Criterion D – Functionality and extensibility
- Criterion E – Evaluation.

Due to the size and scope of the Internal Assessment, class time dedicated to it will be split into two stages. Criterion A and B will take place in May of Year 12, and criterion's C, D and E will occur in late October/November of Year 13. It will be assessed out of 34 raw marks.

Criterion A: Planning (6 marks)
Developing and documenting a scenario, rationale for the proposed solution, and success criteria

Criterion B: Solution overview (6 marks)
The student must provide a record of tasks and a design overview that includes an outline test plan. A *record of tasks* form will be provided and must be used.

Criterion C: Development (12 marks)
Development documentation to include

- The structure of the product and why it is appropriate
- The algorithmic thinking used in the development of the product
- The techniques used in the development of the product and reasons why they are appropriate to it (may include screenshots, exemplar data, reference to information in the appendix)
- Any existing tools that are used in the development of the product, such as code libraries, software packages, web hosting, security information or infrastructure issues

Criterion D: Functionality and extensibility of the project (4 marks)
Use of video to demonstrate the product functioning, maintainability of the product by third parties (appropriate folder/data structures, naming conventions, inline comments)

Criterion E: Evaluation (6 marks)
Evaluation of the product against the success criteria developed in the planning phase. Recommendations for the future development of the product.

Schedule

When	Topic / unit	Content	Allocated time	Assessment	Resources
Weeks 1 & 2	Unit 2: Computer architecture	• CPU, CU, ALU, instruction cycle • Primary & secondary memory; cache • Common OS and application systems • Binary number system • Logic gates and circuits	2 weeks		Computer Science Illuminated (Dale & Lewis); Core Computer Science: For the IB Diploma Program (Dimitriou etc); Advanced Computer Science: For the IB Diploma Program (Dimitriou etc); Object orientated programming with Java (U.Helski) pbaumgarten.com And others
	FIELD TRIPS				
Weeks 4 & 5	Unit 6: Resource management (HL)	• Role of an operating system • Resources the system needs to manage (CPU, memory, network, IO etc) • Resource management techniques (virtual memory, paging, interrupts, polling)	2 weeks	Unit 2 & 6 test	
Weeks 6 & 7	Unit 4.1: Computational thinking principles	• Thinking procedurally • Thinking logically • Thinking ahead • Thinking concurrently • Thinking abstractly	2 weeks		
	HALF TERM HOLIDAYS				
Weeks 8-10	Unit 4.3: Introduction to programming	• Features of a programming language • Variable types and their operations • Iteration with loops • Selection with if/else constructs • Writing sub programs & functions • One dimensional arrays • Collections	3 weeks		

Weeks 11-15	Unit 4.2: Computational thinking program design	• Flow charts • Pseudo code • Analyse an algorithm • Big O notation • Standard search & sort algorithms	5 weeks	Unit 4 test In class programming exercises	
	CHRISTMAS HOLIDAYS				
Week 16	Unit 1: system fundamentals	• Software development life cycle • Change management • Stakeholder analysis • Functional v non-functional requirements • Prototyping • Data migration issues • Usability & accessibility • User documentation & training • Backups • Deployment strategies • SaaS, IaaS, PaaS etc	1 week		
Week 17	Exam review	• Exam review	1 week		
	EXAMS				
Weeks 19-21	Unit 1: System fundamentals	• Continuation of unit 1 from above	3 weeks	Unit 1 test	
	HALF TERM HOLIDAYS				
Week 22	Unit D1: OOP concepts	• The object orientated programming paradigm • Classes as object templates • Object relationships & dependencies • Passing messages between objects • UML diagrams	1 week		

Week 23	Unit D2: OOP features	• Inheritance • Encapsulation • Polymorphism	1 week	Unit D test #1	
Week 24-27	Unit D3: OOP programming	• Object constructor; instantiation • Instance, parameter and local variables • Accessor methods: public, private, protected, default • Coding inheritance (extends) • Coding encapsulation (Getters and setters) • Coding polymorphism (override & overload)	4 weeks	Unit D test #2	
	EASTER HOLIDAYS				
Week 28-31	Unit 5: Abstract data structures	• Two dimensional arrays • Linked lists and array lists • Stacks • Queues • Binary trees	4 weeks		
Week 32	Exam review		1 week		
	EXAMS				
Week 35	Group 4 project	• As organized by Science dept	1 week		
Week 36, 37	Internal assessment: Part 1	• Introduction to IA • Self-managed use of time towards criteria A & B of the IA	2 weeks		

YEAR 13

Week 1 & 2	Case study: Introduction (HL)	• Initial dissection of case study • Allocation of research duties	2 weeks		
	FIELD TRIPS				
Weeks 4-7	Unit D3; Unit D4 (HL); & Unit 5 (HL)	• Revisit OOP programming methods and review exercises (2 weeks) • Introduce Recursion • Recursion exercises (2 weeks)	4 weeks	Unit 5 test	
	HALF TERM HOLIDAYS				
Weeks 8-10	As above	• Finish recursion; test • Focus on programming skills typically necessary for the IA (2 weeks)	3 weeks	Unit D4 test	
Weeks 11-15	Internal assessment	• Complete official IA draft by Christmas break for all 5 criteria	5 weeks		
	CHRISTMAS HOLIDAYS				
Week 16 & 17	Unit 3: Networks	• Types of networks; standards • OSI networking model • VPNs • Network protocols • Compression • Transmission media • Wireless networks	2 weeks		
	EXAMS				

Weeks 19-21	Unit 7: Control systems	• Microprocessors & sensors in control systems • Open v closed systems • Role of feedback in a control system • Central v distributed control systems • Autonomous agents	3 weeks	Unit 3 & 7 test	
	HALF TERM HOLIDAYS				
Week 22-25	Case study (HL)	• Share research conducted into case study • Peer teaching • Peer review/assessment	4 weeks	Paper 3 review test	
Weeks 26, 27	Review	• End of course review	2 weeks	Paper 1 review test Paper 2 review test	
	EASTER HOLIDAYS				
	DP EXAMS				

ToK concepts

Topic	ToK concept
Data	What is the difference between data, information, knowledge and wisdom? To what extent can computers store and impart data, information, knowledge and wisdom? If there are no consequences of data loss, why is it stored? There is no such thing as persistent storage.
Computational thinking	Computational thinking includes: procedure, logic, pre-planning (thinking ahead), concurrency, abstraction and recursion. To what extent are these ways of thinking distinct? To what extent can knowledge in different areas (mathematics, ethics, and so on) be analysed in these ways? To what extent does computational thinking challenge conventional concepts of reasoning?
Characteristics of computer memory	It has been said that human memory is more like an improvised performance than a movie on a DVD. What does this mean? How does human memory differ from computer memory? How does a computer language differ from a natural language?
Nature of computer languages Computer number systems	What are the differences between representing numbers in denary and in binary? In binary, 1 + 1 = 10. Does this tell us anything about the nature of mathematical truth? Does binary represent an example of a lingua franca?
Algorithms can have the appearance of intellect	A chess machine can beat the top human chess players. Does a machine therefore “know” how to play chess? How do we know if other humans feel emotions? Can a machine ever feel an emotion? How would we know? Does information and communication technology, like deduction, simply allow the knower to arrange existing knowledge in a different way, without adding anything, or is this arrangement itself knowledge in some sense? Reasoning as a form of decision-making.

Thinking logically Abstract data types	The map as an abstraction of the territory.

Global understanding

At ISL, global understanding is articulated through the following concepts: Conflict resolution, Social justice, Values and perceptions, Sustainable development, Interdependence, Human rights, Diversity and perspectives. Three levels of depth are considered.

- Understanding and appreciation of their own identity
- Understanding and appreciation of the perspectives of others and our interconnectedness
- Promotion of positive intercultural exchange and/or global interaction

Topic	Global understanding concept
Networks Centralised & distributed control systems Unicode/UTF8 and ASCII Open source software development Function of a computer;	**Interdependence** The exchange of information and ideas across national boundaries. The acceleration of the "globalisation" process that comes with data being accessible anywhere anytime. Remote systems may be in a different time zone and the impact this may have on using the system at both the local and remote end. Different file formats / date formats / currencies used internationally may cause compatibility issues / errors in data interpretation if software is not expecting it. **Values and perceptions & Human rights** Being able to control computers from afar and the risk of loss of empathy for those impacted **Diversity and perspectives** Early ASCII based computers were unable to cater for non-roman character sets. The development of Unicode allows compatibility of Asian language symbols for example. **Sustainable development** Collaborative global software development processes, with the expertise being donated to the greater good. **Conflict resolution** Computers are emotionally impartial but the programmers who code them are not. The way change is managed (such as introduction of new software) can have a significant impact on businesses and the

Social & ethical issues Change management	employees who use the system

IB Learner profile

Topic	Learner profile attribute	Contribution to the development of the attribute(s) of the IB learner profile
Design process Case study	Inquirers	Stake holder interviews for IA Case study research (HL only)
Computer programming	Knowledgeable	Understanding of how to write software programs, including the use of the Object Orientated method
Computational thinking	Thinkers	Computational thinking: Thinking procedurally, thinking logically, thinking abstractly, thinking ahead, thinking concurrently, thinking recursively (HL only) The quality of user documentation can affect the rate of implementation of the new system. Effective collaboration with stakeholders is essential to success
Design process Social & ethical	Communicators	Who bears ethical responsibility for the impact of software on peoples lives?
Algorithm design	Principled	Devising novel algorithms to solve problems Inadequate testing can reduce employee productivity and lead to end-user dissatisfaction.
System design	Open-minded Caring	The role of the end-user must be considered when planning a new system. Who are the stake holders?
Algorithm design	Risk-takers	Experimenting with novel algorithms
	Balanced	Analyse of algorithm results and consideration of refinements
	Reflective	Evaluation of IA product

Development of Approaches to Learning in IB Diploma Computer Science

Topic	ATL skill category (*Communication/ Social/ Self Management/ Research/ Thinking*)	Activities that support the development of these skills
IA – Crteria A & E	Communication skills (Give and receive meaningful feedback)	Students must find and interview a client to develop the requirements of their project, and then interview them again at the end of the project for formal feedback which is assessed.
Case study	Communication skills (Read critically and/or for comprehension)	Students are required to conduct independent research into a topic assigned annually by the IB.
IA – Criterion B	Self-management (Organisational skills)	The IA is an extended assessment involving 30 hours of class time plus at least the equivilant of out-of-class time. Students are required to independently plan and pursue their project through the efficient use of this time.
IA – Criterion C	Creative-thinking skills (Apply existing knowledge to generate new ideas, products or processes)	Students will draw upon their computer programming skills to generate a new software product intended to meet a need for an identified client.
IA – All criteria	Information literacy (Create references and citations, use footnotes/endnotes and construct a bibliography according to recognised conventions)	The IA document must be properly referenced, is verified through TurnItIn for possible submission to the IB.