



State-Schools Computing Curriculum

A 3-year Computing curricular plan for State Schools:
Preparing students to build a strong foundation in
computational thinking, programming skills, and essential
computer science concepts for the MASTEC Examination

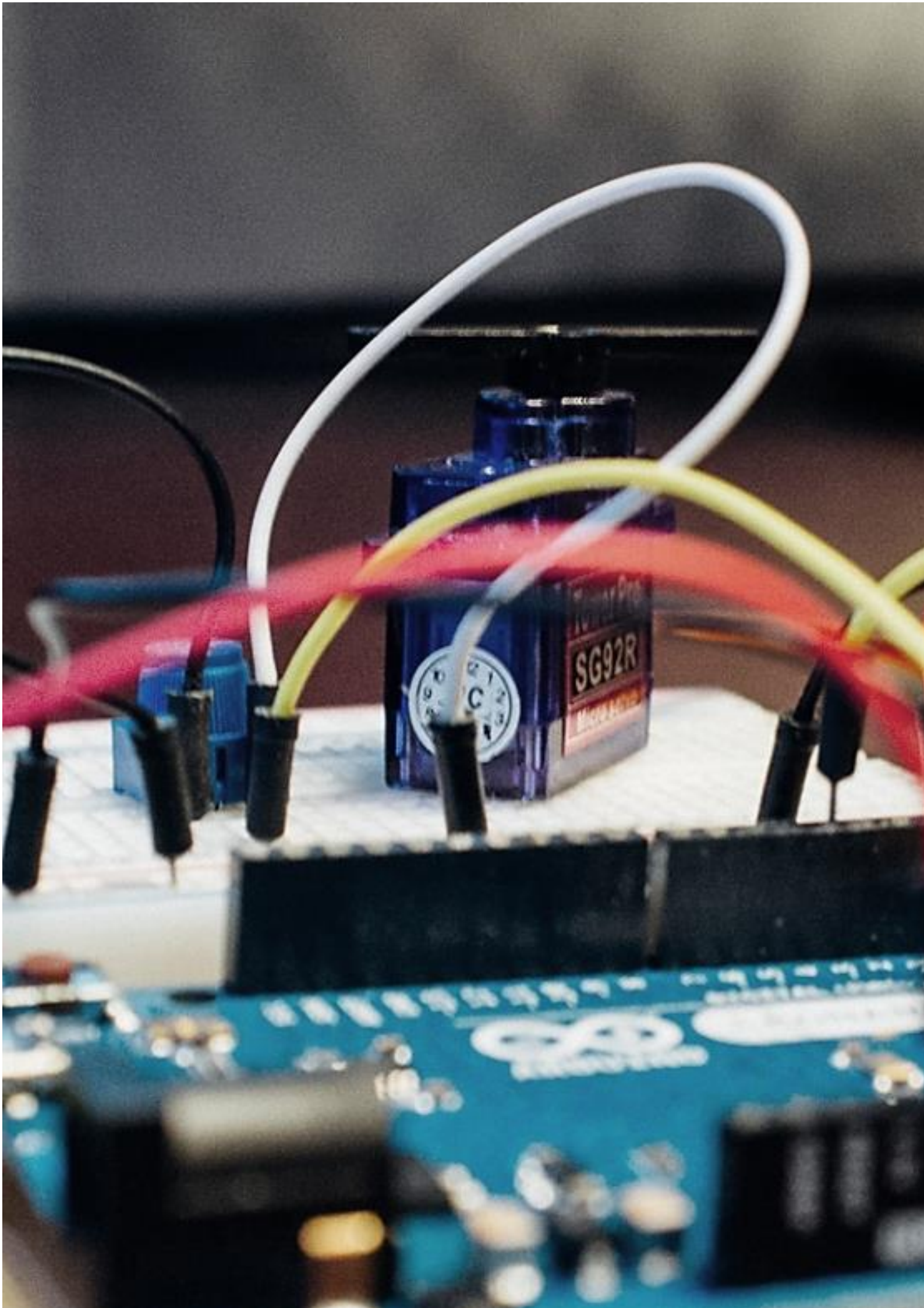
SEC 2027 ONWARDS



GOVERNMENT OF MALTA
MINISTRY FOR EDUCATION, SPORT, YOUTH
RESEARCH AND INNOVATION

Table of Contents

Preface	4
Year 9.....	6
List of Learning Outcomes	6
List of Broad Learning Outcomes for Assessment Reporting	7
Alignment of Broad Learning Outcomes with Syllabus Content	8
Year 10.....	13
List of Learning Outcomes	14
List of Broad Learning Outcomes for Assessment Reporting	14
Alignment of Broad Learning Outcomes with Syllabus Content	15
Year 11	21
List of Learning Outcomes	22
List of Broad Learning Outcomes for Assessment Reporting	22
Alignment of Broad Learning Outcomes with Syllabus Content	23
Assessment Modes.....	27
School Based Assessment	28
How will the SBA mark contribute to the students’ performance at SEC level?	28
How to formulate the SBA mark	28
Components excluded from the SBA mark.....	30
How will the SBA mark contribute to the students’ performance at school level? .	31
Annual Examination Papers.....	32
Year 9 & Year 10 annual examination papers	32
Year 11 annual examination paper	33
Reporting of Marks.....	33
Conclusion.....	34



Preface

Students in the twenty first century classroom need to develop the skills to be employable and re-employable in a constantly evolving job market. Hence, the success of our new Computing curriculum pivots on the development of a methodology that will help shape our students to be:

- *Critical thinkers:* capable of facing information, technology, and situations with a critical outlook.
- *Problem-Solvers:* capable of using knowledge and experience as tools for new problems, and become proficient at applying concepts learned in class to tackle unfamiliar scenarios,
- *Flexible:* capable of reinventing themselves to remain relevant in an ever-changing world with a mindset for lifelong learning,
- *Communicators:* in possession of the “people skills” to render them capable of communication and leadership within a social situation or working environment.

Our curriculum invites a methodology that develops these skills not only through content but also through a range of assessment methods. Students will benefit best from this curriculum if it is delivered with a methodology that is:

- *Student-centred:* teaching content by presenting questions and problems to be solved rather than telling.
- *Careful to employ assessment as a tool for learning rather than exclusively as a measure of learning.*
- *Amenable to differentiated learning:* recognising different abilities and inviting students to present their work in a variety of media.
- *Soft skills enhancing:* giving students the space to develop skills for group work, online communication, viva presentation etc.

In the light of the above, this document presents a complete three-year curricular plan for the subject of Computing for state schools, designed to prepare students for the MASTEC examination. This curriculum prioritizes the progressive development of the Learning Outcomes (LOs) throughout Years 9, 10, and 11 for students who will sit for the SEC 2027 and onwards. It delineates the mandatory content to be covered in each year, provides details related to the Broad Learning Outcomes (BLOs) that group related

assessment criteria, and sets a clear scheme of assessment, including the School-Based Assessment (SBA) and the annual exam papers.

A brief outline of this three-year plan is given hereunder.

Year 9: Establishing Foundational Knowledge

Students will learn about digital components, data representation, input/output devices, memory vs. storage, software, and basic algorithm and dedicated systems development.

Year 10: Building Programming Proficiency

Students will learn binary arithmetic, explore relational databases, networking concepts, machine logic, and developing simple text-based applications using Python programming language.

Year 11: Preparing for the MATSEC Examination

By Year 11, students will have covered the entire SEC Computing syllabus and should be well-prepared for the MATSEC exam. Students will explore CPU specifications, Operating Systems, and the characteristics of Computer Languages and Translators; learn how to develop basic Graphical User Interfaces using Python programming language; and review core concepts for the MATSEC exam.

To supplement this three-year curriculum, teachers are strongly encouraged to refer to the official SEC09 Computing syllabus. The syllabus provides valuable insights into the expected level of detail for each LO, and other relevant details that can be crucial for the smooth and effective delivery of the course.

Following this introduction, a detailed breakdown for Years 9, 10, and 11 is presented in the next sections.

Last updated on September 2024

Josmar Borg (EO Computing)

Anthony Farrugia (HoD)

Marlene Galea (HoD)

Sam Cefai (HoD)

Year 9

In Year 9, teachers will lay the groundwork towards a solid foundation in the field of computer science. Students will explore the basic architecture of modern digital devices, taking into consideration devices powered by processors, micro controllers and System-on-Chip (SoC) technology. They'll delve into how data is represented in digital devices, understand input and output concepts, learn about memory components and storage devices, and about both offline and online software including the implications of software licensing.

Besides, in Year 9 students should be well-acquainted with problem-solving skills, via pseudocode and flowcharts, and in programming simple dedicated systems using sensors and other input and output components. While state-school teachers are likely to use Arduino single-board systems for constructing and programming dedicated systems, the SEC syllabus provides flexibility for teachers and schools to choose other platforms, such as ESP, Micro:Bit or Raspberry Pi. At this stage, students are introduced to the basic programming concepts using a text-based programming language, including sequential, decision and iteration constructs. This foundational knowledge is designed to facilitate the teaching process and enhance the programming learning experience for students as they progress into Year 10.

List of Learning Outcomes

By the end of year 9, teachers should cover the below Learning Outcomes:

(LOs taken directly from the SEC09 Syllabus):

- LO1: Identify the function of the main components in a digital device to process both digital and analogue data.
- LO2: Represent data using different number systems and perform basic binary arithmetic operations. *
** The '[...]perform basic binary arithmetic operations' part of this LO will be covered in Year 10.*
- LO3: Describe the use and the quality of input and output devices.
- LO4: Distinguish between memory and storage in digital devices and describe the implications of the different technologies used.
- LO5: Distinguish and identify the importance of different types of software and software licenses.
- LO11: Produce algorithms, by applying problem solving concepts, and develop simple dedicated systems using boards powered by microcontrollers or System on Chip (SoC) technology.

List of Broad Learning Outcomes for Assessment Reporting

In year 9, educators are expected to report the students' abilities and performance by ticking the appropriate level of ability for the below Broad Learning Outcomes (BLOs).

- 1.1 I can identify the function of the main components in a digital device and the interrelation between these devices.
- 1.2 I can distinguish between analogue and digital data and analyse how sampling rate affects the quality of information in the context of a given scenario.
- 2.1 I can convert between different number systems and explain the use of character coding systems to represent binary bit patterns.
- 3.1 I can describe the use and the quality of input and output devices.
- 4.1 I can distinguish between memory and storage and their use in different digital devices.
- 4.2 I can describe the implications of the different storage technologies used and of cloud storage in different contexts.
- 5.1 I can distinguish and identify the importance of different types of software, software services, and software licences.
- 11.1 I can modify or develop algorithms that include sequence, decision, and iteration constructs, and represent them using pseudocode or flowchart.
- 11.2 I can program a dedicated system to solve a given scenario including the use of 1) digital and analogue input component/s, 2) digital and/or analogue output component/s, and 3) the use of external libraries.

Alignment of Broad Learning Outcomes with Syllabus Content

LO1: Identify the function of the main components in a digital device to process both digital and analogue data.		
BLO 1.1: I can identify the function of the main components in a digital device and the interrelation between these devices.		
1.1a Define the components/devices in a computer system.		
1.1b Identify the components/devices in a computer system.	1.2b Explain the data flow between the main components/devices in a computer system.	
1.1h Define a microcontroller.		
1.1i Identify devices powered by a microcontroller.	1.2i List devices that are powered by a microcontroller.	
1.1j Define System on Chip (SoC) technology.		
1.1k Identify devices using SoC technology.	1.2k List devices that use SoC technology.	
	1.2l Distinguish between microcontroller and SoC.	1.3l Justify the appropriateness of SoC and/or microcontroller technology in a given scenario.
BLO 1.2: I can distinguish between analogue and digital data and analyse how sampling rate affects the quality of information in the context of a given scenario.		
1.1c Define data and/or information.	1.2c Distinguish between data and information.	
1.1d Identify examples of data and information according to a given scenario.	1.2d Recommend examples of data and information according to a given scenario.	1.3d Justify the difference between data and information in a given scenario.
1.1e Define binary units of measurement.	1.2e Justify why digital devices process data in binary.	1.3e Solve problems related to binary units of measurement.
1.1f Define analogue and/or digital data	1.2f Distinguish between analogue and digital data.	
1.1g Define analogue-to-digital (ADC) and/or digital-to-analogue (DAC) conversions.	1.2g Explain the need for ADC and DAC conversions.	1.3g Evaluate the implications of the sampling level used during ADC.

LO2: Represent data using different number systems and perform basic binary arithmetic operations.

BLO 2.1: I can convert between different number systems and explain the use of character coding systems to represent binary bit patterns.

2.1a Recognise different number systems.	2.2a Distinguish between the use of different number.	2.3a Solve problems related to different number systems in specific scenarios.
	2.2b Outline the use of a bit pattern according to a given scenario.	
	2.2c Convert binary numbers into decimal and vice versa.	
	2.2d Convert hexadecimal numbers into binary and vice versa.	
	2.2e Convert hexadecimal numbers into decimal and vice versa.	
2.1j Define standard character coding systems.	2.2j Identify the need of different standard character coding systems.	

LO3: Describe the use and the quality of input and output devices.

BLO 3.1: I can describe the use and the quality of input and output devices.

3.1a Define the concept of the input of data into digital devices.		
3.1b Define the concept of the output of information from digital devices.		
3.1c Define peripheral devices.	3.2c Categorise components in digital devices as being input, output or both.	3.3c Evaluate the use of different components in a specific scenario.
	3.2d List different input and/or output and/or input/output components in a specific scenario.	3.3d Recommend different component/s according to a specific scenario.

		3.3e Justify the choice of different component/s according to a specific scenario.
3.1f Define serial and/or parallel data transfer between peripheral devices and digital systems.	3.2f Distinguish between the concepts of serial and parallel data transfer.	3.3f Explain advantages and/or disadvantages of serial and/or parallel data transfer in a given scenario.
3.1g Identify the different characteristics which determine output quality.	3.2g Explain the different characteristics which determine output quality.	3.3g Evaluate the suitability of resources based on the characteristics which determine output quality in a given scenario.
	3.2h Find solutions that determine the characteristics of output quality.	3.3h Justify the suitability of resources based on the characteristics which determine output quality for a given scenario.

LO4: Distinguish between memory and storage in digital devices and describe the implications of the different technologies used.

BLO 4.1: I can distinguish between memory and storage and their use in different digital devices.

4.1a Define primary and/or secondary storage.		
4.1b Define the memory components used in digital devices.	4.2b List different memory components used in digital devices.	
4.1c Identify memory components used in digital devices.	4.2c Distinguish between memory components used in digital devices in terms of volatility, speed and use.	4.3c Justify the choice of memory components for specific scenarios
4.1d Identify devices that are used for secondary storage.		
4.1e List devices that are used for secondary storage.		

BLO 4.2: I can describe the implications of the different storage technologies used and of cloud storage in different contexts.

	4.2d Distinguish between different storage devices in terms of speed, cost and portability.	
--	---	--

	4.2e Recommend storage media for specific scenarios.	4.3e Justify the choice of storage media for specific scenarios.
	4.2f Distinguish between memory components and storage components.	
4.1g Define cloud storage.	4.2g Describe advantages and disadvantages of cloud storage.	4.3g Justify the use of cloud storage in specific scenarios.

LO5: Distinguish and identify the importance of different types of software and software licenses.

BLO 5.1: I can distinguish and identify the importance of different types of software, software services, and software licences.

5.1a Define the different types of software.		
5.1b Identify the different types of software.	5.2b Distinguish between the different types of software.	5.3b Justify the type of software chosen for a given scenario.
5.1c Define the concept of a software license.		
5.1d Define the software licenses.		
5.1e List different types of software licenses.	5.2e Distinguish between the different types of software licenses.	
	5.2f Recommend a type of license for a given scenario.	5.3f Justify the choice of license for a given scenario.
5.1g Define Software as a Service (SaaS).	5.2g Identify examples of SaaS in a given scenario.	5.3g Justify the use of SaaS in a given scenario.

LO11: Produce algorithms, by applying problem solving concepts, and develop simple dedicated systems using boards powered by microcontrollers or System on Chip (SoC) technology.

BLO 11.1: I can modify or develop algorithms that include sequence, decision, and iteration constructs, and represent them using pseudocode or flowchart.

11.1a Define the term algorithm.		
11.1b Define pseudocode and/or flowcharts.	11.2b Distinguish between pseudocode and flowcharts.	
11.1c Identify symbols used in flowcharts.		
11.1d Define symbols used in flowcharts.		

11.1e Define the basic programming constructs.	11.2e Complete the missing parts of a given algorithm.	11.3e Interpret a given algorithm.
		11.3f Construct an algorithm for a given scenario.
11.1g Define syntax and/or logical and/or runtime errors.	11.2g Distinguish between syntax, logical and runtime errors.	
	11.2h Explain the effect of an error on program compilation and/or runtime.	
BLO 11.2: I can program a dedicated system to solve a given scenario including the use of 1) digital and analogue input component/s, 2) digital and/or analogue output component/s, and 3) the use of external libraries.		
11.1i Develop a program for a dedicated system that includes output and/or input constructs using digital components only.	11.2i Develop a program for a dedicated system that includes analogue output using digital components only.	11.3i Develop a program for a dedicated system that includes output and/or input constructs using both digital and/or analogue components.
	11.2j Develop a program for a dedicated system that includes arithmetic and assignment operators.	
	11.2k Develop a program for a dedicated system that includes selection constructs.	11.3k Develop a program for a dedicated system that includes nested selection constructs.
	11.2l Develop a program for a dedicated system that includes iteration constructs.	11.3l Develop a program for a dedicated system that includes nested iteration constructs.
		11.3m Develop a program for a dedicated system that includes the use of libraries.
		11.3n Trace values of analogue sensors in real time.

Year 10

In Year 10, students build upon the foundations laid in Year 9. For instance, they will not only understand how to represent data in Binary, Decimal, and Hexadecimal but also learn the basics of binary arithmetic, including addition, subtraction and bit-shift operations. Besides, teachers must also focus on advancing the students' programming abilities acquired in Year 9 by using the Python programming language. Programming should be at the heart of most lessons to ensure students grasp the core concepts of software coding. Teachers should emphasize the significance of cultivating good programming habits from the outset, including the incorporation of inline documentation, the creation of readable code, the promotion of code reusability, the implementation of code refactoring techniques, and the adoption of modularity.

For a comprehensive understanding of modern IT systems, students should also comprehend the importance, and the use, of Databases. Teachers will guide students in interpreting and creating simple database queries using SQL, enabling them to effectively interact with databases. Moreover, teachers should ensure that students grasp the role of databases as back-ends for IT applications.

Year 10 also introduces students to the principles of machine logic. They will engage in hands-on activities to design logic circuits and create truth tables and Boolean expressions using the NOT, AND, and OR gates. The concept of the NAND and NOR gates, as universal gates, and their implications in digital devices will also be introduced.

Teachers will also provide students with a fundamental understanding of networking concepts, emphasizing how devices communicate with each other. This foundational knowledge of computer networks will enhance students' overall understanding of IT systems.

List of Learning Outcomes

By the end of year 10, teachers should cover the below Learning Outcomes:

(LOs taken directly from the SEC09 Syllabus):

LO2: Represent data using different number systems and perform basic binary arithmetic operations.

**The 'Represent data using different number systems [...]' part of this LO would have already been covered in Year 9.*

LO8: Understand principles of machine logic in general and produce logic circuits, truth tables & Boolean expressions using the NOT, AND, and OR gates.

LO9: Show an understanding of a relational database, its structure and use.

LO10: Outline networking concepts of how devices communicate.

LO13: Develop programs using Python programming language that includes textual and graphical interface.

**In Year 10 the focus is on text-based interface.*

List of Broad Learning Outcomes for Assessment Reporting

In year 10, teachers are expected to report the students' abilities and performance by ticking the appropriate level of ability for the below Broad Learning Outcomes (BLOs).

2.2 I can perform binary arithmetic operations, including addition, subtraction, and bitwise operations.

8.1 Understand principles of machine logic in general and produce logic circuits, truth tables and Boolean expressions using the NOT, AND, and OR gates.

8.2 Represent and understand the implications of universal gates and the use of logic gates in digital devices.

9.1 Identify the different aspects of a DB structure and represent its schema using ERD.

9.2 Analyse a DB and suggest suitable validation methods.

9.3 Use SQL to develop DB queries.

10.1 Define and distinguish between the different types of networks (LAN, WAN, PAN), connectivity (wired or wireless), the client-server architecture, and the different network components.

10.2 Explain the need of protocols for devices to communicate in a secured network.

- 13.1 Interpret and develop programs using input, output, decisions, and iterative constructs.
- 13.2 Interpret and develop programs using ready-made modules, data structures, and user-defined functions.

Alignment of Broad Learning Outcomes with Syllabus Content

LO2: Represent data using different number systems and perform basic binary arithmetic operations.		
BLO 2.2: I can perform binary arithmetic operations, including addition, subtraction, and bitwise operations.		
	2.2f Represent positive and negative numbers in binary using two's complementation method.	
	2.2g Carry out binary addition and/or subtraction.	2.3g Justify the use of two's complement representation.
2.1h Relate arithmetic bit shift operations with multiplication and division operations.	2.2h Carry out binary multiplication and division.	
	2.2i Solve problems related to range of values in a given register.	2.3i Justify the occurrence of numerical overflow.

LO8: Understand principles of machine logic in general and produce logic circuits, truth tables & Boolean expressions using the NOT, AND, and OR gates.		
BLO 8.1: Understand principles of machine logic in general and produce logic circuits, truth tables and Boolean expressions using the NOT, AND, and OR gates.		
8.1a Define a logic gate and/or a logic circuit and/or truth table.		
8.1b Represent the AND, OR and NOT gates using standard symbols.		
8.1c Draw the truth table of the AND and/or OR and/or NOT Gate.	8.2c Draw the truth table for a given logic circuit and/or Boolean expression.	8.3c Derive the truth table to represent the solution for a given scenario.
8.1d Represent the AND and/or OR and/or NOT gates using Boolean expressions.	8.2d Draw the logic circuit for a given truth table and/or Boolean expression.	8.3d Derive the logic circuit to represent the solution for a given scenario.

	8.2e Express the Boolean expression from a given logic circuit and/or truth table.	8.3e Derive the Boolean expression to represent the solution for a given scenario.
	8.2f Complete a given logic circuit and/or truth table and/or Boolean expression.	8.3f Interpret a given logic circuit and/or truth table and/or Boolean expression.
8.1i Identify logic gates from truth tables.		
BLO 8.2: Represent and understand the implications of universal gates and the use of logic gates in digital devices.		
8.1g Define half adder.	8.2g Construct the truth table and/or logic circuit and/or Boolean expression of the half adder.	8.3g Solve a given scenario using concepts of the half adder.
8.1h Represent the NAND and/or NOR gate using standard symbols and/or truth table.	8.2h Describe the use of the NAND and/or NOR Gates as universal gates.	8.3h Explain the economical and manufacturing implications of the NAND and/or NOR gate as universal gates.

LO9: Show an understanding of a relational database, its structure and use.

BLO 9.1: Identify the different aspects of a DB structure and represent its schema using ERD.

9.1a Define a database.	9.2a Distinguish between manual record keeping and electronic databases.	
	9.2b Describe advantages and/or disadvantages of manual record keeping and electronic databases.	9.3b Explain the suitability of an electronic database in a given scenario.
9.1c Define a Database Management System (DBMS).	9.2c Distinguish between a DBMS and a database.	
9.1d Identify the key elements of a typical structure of a database.	9.2d Describe the key elements of a typical structure of a relational database.	
9.1e Identify a key element of a typical structure of a relational database according to a given scenario.	9.2e Recommend a key element of a typical structure of a relational database according to a given scenario.	9.3e Justify the suitability of a key element of a typical structure of a relational database.
	9.2f Interpret an ERD Diagram.	9.3f Construct an ERD diagram for a given scenario
	9.2l Justify the role of a database as a back-end to IT applications.	

BLO 9.2: Analyse a DB and come up with suitable validation methods.		
9.1g Define data validation and/or integrity.	9.2g Explain the importance of data validation and/or integrity.	9.3g Recommend data validation rules and/or field sizes for specific fields.
		9.3h Justify the choice of a data validation rule and/or field size for specific fields.
BLO 9.3: Use SQL to develop DB queries.		
9.1i Define a query.	9.2i Describe the use of Structured Query Language (SQL) in databases.	
	9.2j Interpret a query.	
	9.2k Modify a query.	9.3k Construct a query to search for specific records by using SQL.

LO10: Outline networking concepts of how devices communicate.		
BLO 10.1: Define and distinguish between the different types of networks (LAN, WAN, PAN), connectivity (wired or wireless), the client-server architecture, and the different network components.		
10.1a Define a network.	10.2a List advantages for networked devices as opposed to stand-alone devices.	
10.1b Identify Internet of Things (IoT) devices.	10.2b Describe the concept of the IoT.	10.3b Recommend examples of IoT devices.
10.1c Classify between wired and wireless connections.	10.2c Describe advantages and/or disadvantages between wired and wireless connections.	10.3c Justify a suitable connection (wired or wireless) for a given scenario.
	10.2d Recommend a suitable use according to the different networking connections of a device in relation to a given scenario.	10.3d Justify the choice of a suitable use according to the different networking connections of a device in relation to a given scenario.
	10.2e Define the role of the client and/or server in network communication.	
10.1k List different hardware components which are needed by devices to communicate.	10.2k Recommend suitable network components required for a given scenario.	10.3k Justify the use of suitable network components required for a given scenario.
10.1l List different types of networks.	10.2l Distinguish between the different types of networks.	

10.1m Define different types of networks.	10.2m Recommend the use of a type of network for a given scenario.	10.3m Justify the use of a type of network for a given scenario.
BLO 10.2: Explain the need of protocols for devices to communicate in a secured network.		
10.1f Describe the need for a device to be uniquely identified in a network.	10.2f Distinguish between IP and MAC address.	
	10.2g Distinguish between IPv4 and IPv6.	10.3g Justify the need for IPv6.
10.1h Define the need for a network protocol as a method of communication between different devices.	10.2h Justify the need of host names and/or translation service between host names and IPs.	
10.1i Define HTTP and/or FTP protocol.	10.2i Distinguish between FTP and HTTP.	
10.1j Define the term firewall.	10.2j Explain the need for security in networking.	10.3j Justify the need for security in a given scenario.

LO13: Develop programs using Python programming language that includes textual and graphical interface.		
BLO 13.1: Interpret and develop programs using input, output, decisions, and iterative constructs.		
13.1a Interpret a program snippet that includes output statements.		13.3a Interpret a program snippet that includes output and /or formatted output statements.
13.1b Develop a program using output statements according to a given scenario.		13.3b Develop a program snippet that includes output and/or formatted output statements according to a given scenario.
13.3b Develop a program snippet that includes output and/or formatted output statements according to a given scenario.		
13.1e Interpret a program snippet that includes the use of variables and/or input statements.	13.2e Interpret a program snippet that includes the use of variables and/or type	

	conversion functions and/or input statements.	
13.1f Develop a program that includes the use of variables and/or input statements.	13.2f Develop a program using variables and/or type conversion functions and/or input statements.	
	13.2i Interpret a program snippet that includes decision statements.	13.3i Interpret a program snippet that includes nested decision statements.
	13.2j Develop a program using decision statements.	13.3j Develop a program using nested decision statements.
	13.2k Interpret a program snippet that includes iteration statements.	13.3k Interpret a program snippet that includes nested iteration statements.
	13.2l Develop a program using iteration statements.	13.3l Develop a program using nested iteration statements.
		13.2y Implement good programming practices.
BLO 13.2: Interpret and develop programs using ready-made modules, data structures, and user-defined functions.		
		13.3o Interpret a program snippet that includes user-defined functions.
		13.3p Develop a program using user-defined functions.
	13.2q Interpret a program snippet that includes tuples and/or lists as data structures.	13.3q Interpret a program snippet that includes dictionaries as data structure.
	13.2r Develop a program using tuples and/or lists as data structures.	13.3r Develop a program using dictionaries as data structures.
	13.2t Interpret a program snippet that includes the Random module.	13.3t Interpret a program snippet that includes the Random module to access and modify items in data structures.
	13.2u Develop a program using the Random module.	13.3u Develop a program using the Random module to access and/or modify items in data structures.
		13.2y Implement good programming practices.

The assessment criteria below may not necessarily be included when evaluating the students' work related to the BLOs of LO13. However, these can be assessed in the both the Annual and SEC exam papers.

13.1c Define a variable.		
13.1d Identify between different types of variables for a given scenario.		
13.1g Identify arithmetic operations in a given scenario.	13.2g Interpret a program snippet that includes arithmetic operations.	13.3g Interpret a program snippet that includes assignment operators.
	13.2h Develop a program using arithmetic operations.	13.3h Develop a program using assignment operators.
13.1m Define the term function and/or module.	13.2m Distinguish between function and module.	
	13.2n Distinguish between built-in function and user-defined function.	
	13.2o Interpret a program snippet that includes String functions.	
	13.2p Develop a program using String functions.	
	13.2s Interpret the statement import and/or from-import and/or from-import-as.	
	13.2w Identify syntax and/or logical and/or runtime errors in a given algorithm.	13.3w Solve syntax and/or logical and/or runtime errors in a given algorithm.
13.1x List good programming practices.	13.2x Describe the importance of good programming practices.	

The assessment criteria below will be covered in Year 11.

	13.2v Develop, with assistance, a program that includes a graphical interface using any readymade module.	13.3v Develop a program that includes a graphical interface using any readymade module.
--	---	---

Year 11

Building upon the foundational knowledge and skills established in prior years, the Year 11 curriculum delves deeper into the core principles of computing. Teachers will facilitate the development of students' ability to distinguish between low-level and high-level programming languages, enabling them to understand the appropriate usage and contexts of each. Students should recognize the importance of language translators in transforming code into executable programs. Teachers will guide students in distinguishing between assemblers, compilers, and interpreters, highlighting their unique functions in software development.

Teachers will also guide students through the internal workings of the CPU, the significance of CPU specifications and how they are tailored for various computing scenarios. The knowledge about low-level languages, already gained during this year, will facilitate the teaching and learning process when it comes to the instruction set architecture (ISA), the CPU's native language that dictates how a processor communicates and performs tasks. By understanding the ISA, students acquire an understanding of the CPU's fetch-execute cycle, which is fundamental to the operation of all modern CPUs, and hence into how computers process instructions at the most basic level.

Furthermore, the curriculum emphasizes the critical role of Operating Systems (OS) in managing hardware and software resources effectively. Students will gain a basic understanding of the core functionalities performed by an OS and become familiar with various system utilities that help maintain the health and efficiency of a computer system.

To solidify their programming proficiency that has been acquired in previous years, students will have the opportunity to develop programs incorporating graphical user interfaces using readily available libraries such as Tkinter, PyGame, or PyQt. This practical application reinforces coding concepts and encourages creativity in software development.

As the final year of studies, students should be well prepared for the final SEC Examination paper; in the sense that the theoretical component of this syllabus should be fully covered whilst also allocating time for revision.

List of Learning Outcomes

By the end of year 11, teachers should cover the below Learning Outcomes:

(LOs taken directly from the SEC 09 Syllabus):

- LO6: Assess the suitability of different CPU specifications for different scenarios.
- LO7: Describe the role and the basic functions of an Operating System in different digital systems.
- LO12: Distinguish between low and high-level languages.
- LO13: Develop programs using Python programming language that includes textual and graphical interface.

**** In Year 11, the focus is on graphical user interface.***

List of Broad Learning Outcomes for Assessment Reporting

In year 11, teachers are expected to report the students' abilities and performance by ticking the appropriate level of ability for the below Broad Learning Outcomes (BLOs).

- 6.1 Explain the purpose of a CPU, the CPU components, the system bus, and the relevance of different CPU specifications for different scenarios.
- 6.2 Explain the purpose of the CPU instruction set and how computers process instructions at the most basic level, through the fetch-execute cycle.
- 7.1 Describe the role of an Operating System (OS) in different digital systems.
- 7.2 Explain the main functions performed by an OS.
- 7.3 Describe the use of different system utilities.
- 12.1 Distinguish between low- and high-level languages based on their characteristics.
- 12.2 Identify the need for language translation and the difference between translators.
- 13.3 Develop simple programs using Python programming language that includes graphics-based interface.

Alignment of Broad Learning Outcomes with Syllabus Content

LO6: Assess the suitability of different CPU specifications for different scenarios.		
BLO 6.1: Explain the purpose of a CPU, the CPU components, the system bus, and the relevance of different CPU specifications for different scenarios.		
6.1a Define the term CPU and/or 'CPU cycle'.	6.2a Describe the relationship between the CPU cycle and the system clock.	
6.1b Identify the basic components of the CPU.	6.2b Describe the basic components of the CPU.	6.3b Distinguish the function of the basic components of the CPU.
6.1c List the basic components of the CPU.		
6.1d Define a single core and/or a multi-core CPU.	6.2d Distinguish between a single core and a multi-core CPU.	6.3d Relate the system clock and/or the number of CPU cores to the overall system performance.
6.1h List the three main types of the system bus.	6.2h Describe the purpose of the system bus.	
6.1i Classify the different types of buses.	6.2i Define the address and/or data and/or control bus.	
6.1l Define address space.	6.2l Describe the relation between the address bus and the address space.	
6.1m Define word length.	6.2m Describe the relation between the data bus and the word length.	6.3m Explain how the address space and/or word length impact system performance.
BLO 6.2: Explain the purpose of the CPU instruction set and how computers process instructions at the most basic level, through the fetch-execute cycle.		
	6.2e Define machine instruction.	
6.1f Identify the structure of a machine instruction.	6.2f Distinguish between opcode and operand.	6.3f Describe the function of the instruction set.
		6.3g Interpret a snippet of machine instructions.
6.1j Define the Fetch and Execute Cycle.		6.3j Outline the underlying concept of the Fetch and Execute CPU cycle (machine cycle).
		6.3k Explain the function of the system bus in terms of read and write cycles.

LO7: Describe the role and the basic functions of an Operating System in different digital systems.		
BLO 7.1: Describe the role of an Operating System (OS) in different digital systems.		
7.1a Define the purpose of an Operating System (OS).		
7.1b Define a General-Purpose OS and/or Embedded OS.	7.2b Distinguish between General-Purpose OS and Embedded OS.	7.3b Recommend the type of OS (General-Purpose and/or Embedded OS) appropriate for a given scenario.
		7.3c Justify the choice of OS type (General-Purpose and/or Embedded OS) for a given scenario.
7.1d Define single and/or multitasking OS.	7.2d Distinguish between single and multitasking OS.	
		7.3t Explain compatibility issues between the OS, CPU and application software.
BLO 7.2: Explain the main functions performed by an OS.		
7.1e List the basic functions of an OS.	7.2e Explain the need for the basic functions of the OS.	
7.1f Define User Interface (UI).		
7.1g List different types of UI.		
7.1h Identify different types of UI.		
7.1i Define the different types of UI.	7.2i Distinguish between the different types of UI.	7.3i Recommend a UI for a given scenario.
		7.3j Justify the choice of UI for a given scenario.
7.1k Define a filing system.		
7.1l List different filing systems.	7.2l Describe the use of filing systems.	
	7.2m Describe the use of virtual memory.	
	7.2n Describe the use of device drivers.	7.3n Justify the use of buffering to deal with devices with different speeds.
7.1o Define CPU process scheduling.	7.2o Describe Round Robin and/or Priority scheduling.	

	7.2p Distinguish between Round Robin scheduling and Priority scheduling.	
BLO 7:3: Describe the use of different system utilities.		
7.1q Define a system utility.		
7.1r List different system utilities.	7.2r Describe the use of different system utilities.	7.3r Recommend different system utilities according to a given scenario.
		7.3s Justify the use of different system utilities according to a given scenario.

LO12: Distinguish between low and high-level languages.		
BLO 12.1: Distinguish between low- and high-level languages based on their characteristics.		
12.1a Define Low-Level Language (LLL) and/or High-Level Language (HLL).	12.2a Distinguish between LLL and/or HLL.	12.3a Justify the use of a specific type of programming language for a given scenario.
12.1b Classify languages in LLL and/or HLL.		
BLO 12.2: Identify the need for language translation and the difference between translators.		
12.1c Define source code and/or executable code.	12.2c Distinguish between byte code and executable code.	12.3c Explain the need for code translation, including the relation between the source code, byte code and executable code.
12.1d List the different types of language translators.	12.2d Distinguish between the different types of language translators.	12.3d Recommend a language translator for a given scenario.
		12.3e Justify the choice of language translator for a given scenario.

LO13: Develop programs using Python programming language that includes textual and graphical interface. In Year 11, the focus is on graphical user interface.		
BLO 13.3: Develop simple programs using Python programming language that includes graphics-based interface.		
	13.2v Develop, with assistance, a program that includes a graphical interface using any readymade module.	13.3v Develop a program that includes a graphical interface using any readymade module.



Assessment Modes

In the dynamic landscape of education, the evaluation of student performance has undergone significant transformations. Traditionally, assessments were confined to periodic tests and final exams, providing only a snapshot of a student's abilities at specific moments. However, as we progress further into the 21st century, educators recognize the need for a more holistic approach, one that considers students' continuous efforts and progress throughout their academic journey.

For several years, teachers have diligently provided a continuous assessment mark for each student attained from various assessment modes, such as assignments, projects, quizzes, participation in class and more. Whilst these marks allowed educators to track student progress and identify areas needing improvement, they were primarily used for reporting purposes. We are now seeing a shift towards evaluating students' proficiency by means of quantifying the continuous assessment marks alongside the exam marks.

While traditional methods like exams are crucial, in isolation these will overshadow the ongoing learning journey. Therefore, it is very important that both types of assessment work in tandem. The continuous assessment weight acknowledges the ongoing effort, while the exam weight reflects the culmination of knowledge at a specific point.

As we implement this new approach, students will benefit from knowing that their everyday contributions matter and that their sustained efforts are essential. This change not only promotes a more balanced and inclusive assessment system but also encourages students to stay committed and motivated throughout their educational journey. Teachers, too, will find this system more reflective of their students' true capabilities, enabling them to provide more meaningful feedback and support.

School Based Assessment

The School-Based Assessment (SBA) is an approach to students' evaluation that emphasises continuous learning and application of knowledge. This framework moves beyond traditional, one-off assessments, by incorporating a diverse range of assessment modes, such as classwork, homework, presentations, projects, portfolios, group work, performance tests and more. These varied methods allow teachers to tailor assessments to specific learning objectives and student age groups and provide a more accurate picture of students' progress.

How will the SBA mark contribute to the students' performance at SEC level?

In the MATSEC system, student assessment involves two components, the SEC Controlled Assessment (Exam Paper) and the SBA. Here's how they contribute to students' overall grades:

1. **SEC Controlled Assessment (Exam Paper):** The SEC Controlled Assessment is a written exam that carries a total of 100 marks. Students can achieve grades ranging from 1 (highest achievable grade) to 8 (lowest achievable grade) based on the Controlled Assessment mark. These grades align with the grading structure used in previous years, with the recent addition of grade 8 for attainment at Level 1.
2. **School-Based Assessment:** The SBA contributes to the MATSEC level achieved by students (Level 1, 2, or 3). The assessment ratio is 30% from SBA and 70% from the Controlled Assessment exam paper. To attain the expected Level 3, students must satisfy the requirements of the examiners in both the SBA and the Controlled Assessment components.

MATSEC collects students' SBA marks from schools in Year 11 as three separate marks (out of 100), representing the SBA achievements in Year 9, 10, and 11. It is important to note that MATSEC will not be carrying out any moderation of the SBA.

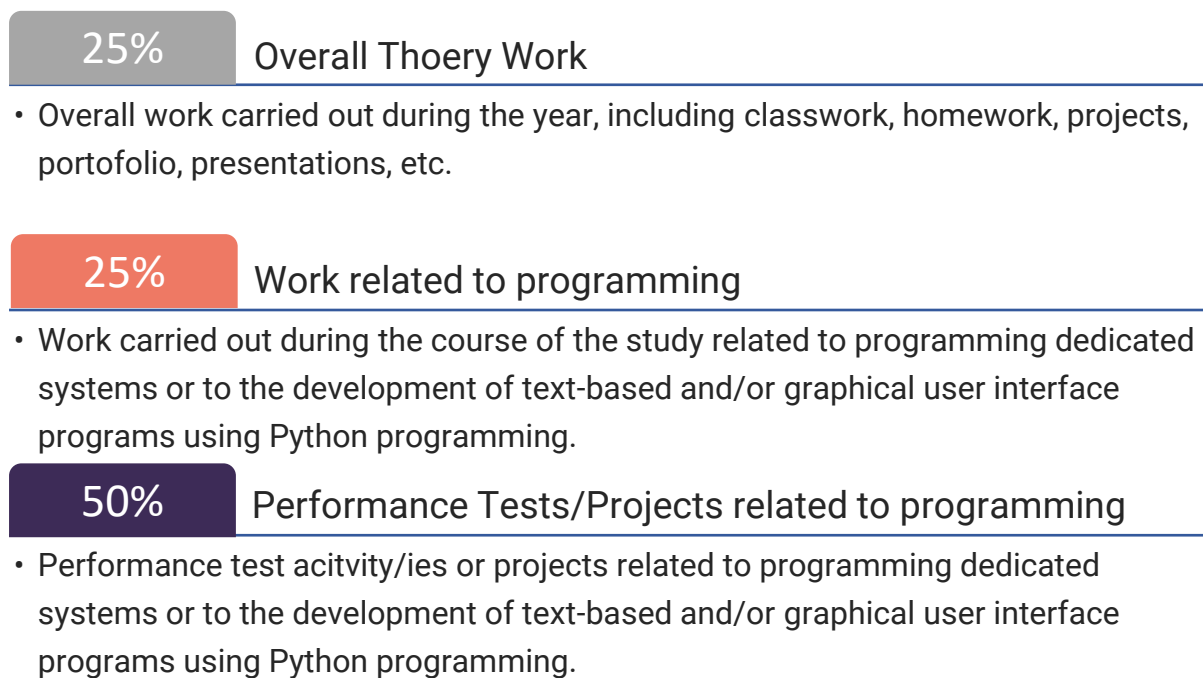
How to formulate the SBA mark

The SBA framework empowers schools, together with the subject teachers, with flexibility in assessment design and implementation. Schools can choose the format of the assessment, as well as the specific learning objectives to be evaluated. Additionally, they have the option to adopt the available pre-designed exemplars provided by MATSEC

or develop their own entirely. This flexibility allows schools to tailor assessments to their unique teaching approach and student needs.

However, it is important to consider the specific requirements of the SEC Computing syllabus in which certain assessment criteria **cannot be assessed in the exam paper**. Particularly, assessment criteria related to programming of dedicated systems (in LO11), and the development of programs using Python (in LO13). Since these LOs encompass at least 50% of the entire teaching time, mostly to be done in Year 9 and Year 10, it is logical for teachers to allocate a significant portion of the SBA mark to assessment modes that effectively gauge student proficiency in these areas.

While prioritizing assessment in these crucial areas is necessary, focusing solely on LOs 11 and 13 could downplay the importance of student performance in other LOs that contribute to the continuous assessment throughout the course. To address this, the following diagram suggests a way to balance the importance of continuous assessment, based on overall student performance, with the specific needs of the SEC Computing syllabus.



Whilst the above diagram is only a suggestion, it is very important that the SBA mark is not obtained only from one or two tests per year as this defeats the concept of the continuous assessment approach.

For the Computing Education Officer (EO) to fully support the teacher and ensure accountability, it is crucial to discuss with and inform the EO of the chosen assessment model for the SBA whenever the need arises or upon request by the EO.

Components excluded from the SBA mark

The aim of the SBA is to evaluate students' performance against the specific learning objectives and assessment criteria outlined in the Computing syllabus. Therefore, it should not consider factors outside the scope of the syllabus, such as student behaviour, attendance, or skills/knowledge not explicitly mentioned.

For example, while teaching programming for dedicated systems (LO 11), the SBA should not include assessments on circuit design and development. Although building a circuit using a breadboard, wires, and input/output components might precede programming it, circuit design and development do not feature in the syllabus.

Similarly, while designing a database schema (LO 9) and interpreting existing schemas are part of the syllabus, the SBA should not assess the creation of a functional database using DBMS like MS Access. While this might be a practical teaching approach, the syllabus focuses on understanding database design principles, not database creation tools.

When teaching binary arithmetic (2.2g), the concept of One's Complement is important, but the 'Sign and Magnitude' concept must be excluded from the SBA mark. Similarly, cloud-based services assessment should focus solely on Software as a Service (SaaS), with Infrastructure as a Service (IaaS) and Platform as a Service (PaaS) left out.

Memory components (LO4) are limited to RAM, ROM, and Cache. Hence, while the teacher may mention EPROM, EEPROM, and other variants as part of the teaching process, these cannot be included in the SBA. Similarly, while teaching storage devices, understanding the different technologies used (magnetic, optical, or electronic) is crucial for students to grasp the impact these technologies have in terms of speed, cost, and portability in different storage devices. However, the assessment cannot include the mechanical or electronic specifics of these technologies. For example, teachers can assess the students' ability to recommend storage devices based on their characteristics (linked to the technology) for a particular scenario (4.2e), but they cannot directly assess the students' ability to categorize storage devices by technology as this is nowhere included in the syllabus.

For CPU components (6.1b), mentioning cache memory (referencing section 4.1a) as an internal component is sufficient, without delving into L1 and L2 cache differences. In process management (7.1e), referring to a single job as a process or program is enough without the need to go into Threads, Parallel Threads, etc. Describing the use of filing systems like FAT32, NTFS, APFS, and EXT4 (7.2l) is adequate, and assessing the technical differences between them must not be done.

Process priority scheduling (7.2o) should focus on non-preemptive scenarios only, leaving out pre-emptive scheduling due to its additional complexities. Scheduling tables that include process arrival time, burst time, completion time and other details, as well as Gantt Charts, can be used for demonstration purposes only and must not be assessed. Students should grasp the concept without these intricacies.

This principle also applies to classroom participation. Some students may not be naturally inclined to actively participate in class but could still have a strong understanding of the subject matter. Participation grades can be subjective, leading to potential bias, and may disadvantage students with anxiety, shyness, cultural differences, or certain disabilities. Additionally, different learning styles and external factors, such as family responsibilities, can impact participation. Emphasising participation may also prioritise quantity over quality and undermine intrinsic motivation. Due to these reasons, **it is best to leave participation out of the SBA**, so that all students have equal opportunities to demonstrate their knowledge without being disadvantaged by their natural inclinations or other factors.

In essence, the SBA should be a clear reflection of student proficiency in the specific skills and knowledge defined within the Computing syllabus. Excluding elements outside this scope ensures a fair and focused assessment.

How will the SBA mark contribute to the students' performance at school level?

In Year 9 and Year 10, the SBA mark will contribute to 30% of the final exam mark. Therefore, to calculate the final exam mark, both the SBA and the annual examination paper will be considered. The SBA will account for 30 marks, while the Annual Examination Paper will contribute 70 marks. This approach is further explained in the next section regarding the annual examination papers.

In Year 11, the SBA mark will not be included in the final exam mark but must still be reported in the **mySchool** system in the Continuous Assessment column (out of 100).

It is important to note that besides quantifying the students' efforts throughout the course of the study, educators should also use the SBA as an opportunity to give feedback to students about their work, showing them where they are doing well and where and how they can improve. This ongoing feedback is crucial in helping students understand their progress and areas for development, making the SBA a valuable tool for learning and growth.

Annual Examination Papers

The annual examination papers will be developed internally by the department and issued by the Educational Assessment Unit. Exam papers will be set in English and students are expected to answer in English. Students may use flowchart stencils, but calculators are not allowed.

Year 9 & Year 10 annual examination papers

The Year 9 and Year 10 annual examination papers will be structured as a 1-hour and 30-minute assessment, covering key learning outcomes specific to each respective year. The Year 10 paper may also include questions that require fundamental concepts from Year 9 content.

The examination paper format will mirror the SEC Controlled Assessment exam paper, with a slight modification in the allocation of marks. The annual papers for Years 9 and 10 will be valued at 70 marks and will consist of two sections:

- **Section A:** Up to four questions, which may be scenario-based, totalling 30 marks.
- **Section B:** Two scenario-based questions, totalling 40 marks.

To determine the final examination percentage, the mark obtained by the student in the annual examination paper (which is out of 70) will be converted to a percentage (out of 100).

A sample excerpt from an exam paper front cover is provided below for a visual representation of this adjustment.

QUESTION	Section A	Section B	Exam Mark Obtained	Final Mark
MAXIMUM	Up to four questions (30 Marks)	Two questions (40 Marks)	70	100
MARK				

Year 11 annual examination paper

The Year 11 annual examination paper will be structured as a 1-hour and 45-minute assessment, mostly covering learning outcomes covered in Year 11. However, consistent with past practices, it may also incorporate questions that assess knowledge and application of skills from Years 9 and 10 content.

The examination paper will be out of 100 and the format will be the same as that of the SEC Controlled Assessment exam paper. It will therefore consist of two sections:

- **Section A:** Up to five questions, which may be scenario-based, totalling 40 marks.
- **Section B:** Three scenario-based questions, totalling 60 marks.

A sample excerpt of a Year 11 exam paper front cover is provided below for a visual representation.

QUESTION	Section A	Section B	Final Mark
MAXIMUM	Up to 5 questions (40 marks)	3 questions (60 Marks)	100
MARK			

Reporting of Marks

For reporting purposes, at end of the year educators will need to submit two marks in the **mySchool** portal in the following manner:

- **Continuous Assessment:** This mark represents the SBA performance and should be submitted out of 100. If a student does not submit any work throughout the scholastic year that contributes to the continuous assessment mark, it should be recorded as “NP” (Nil Presented).
- **Annual Exam*:** This mark should be submitted out of 100 or recorded as "A" if the student is absent.

** In case of Year 9 and Year 10, this mark must be calculated as a percentage of the mark obtained (out of 70) in the annual examination paper.*

Let's consider an example of a Year 9 or Year 10 student who achieved 75 out of 100 marks for the SBA and 48 out of 70 marks for the annual examination. The final reporting should be as follows:

- **Continuous Assessment:** The mark reported should be **75**, reflecting the SBA.
- **Annual Exam:** The mark for the annual examination will be reported as **69**. This is calculated as a percentage of the mark obtained then rounded accordingly; i.e. $(48 / 70) \times 100 \approx 68.6$ (69%)
- **Global Mark:** This will be automatically calculated and generated by the mySchool system by weighting the Exam Mark at 70% and the Continuous Assessment Mark at 30%. In this case the global mark would be **71**. It is important to note that if the Continuous Assessment is reported as “NP” (Nil Presented) and/or the Annual Exam mark is reported as “A” (Absent), the global mark will not be generated and left as nil.

Conclusion

The new curriculum approach is designed to provide a complete and balanced educational experience for our students. The dual focus on continuous assessment and final exams ensures that both ongoing effort and accumulated knowledge are recognized and rewarded.

The curriculum emphasises the importance of teaching concepts rather than facts as we aim to equip our students with the skills and knowledge they need to succeed in the ever-evolving world of technology. We believe that this approach will not only enhance their learning experience but also prepare them for future challenges both in their academic journey and IT industry.

GOVERNMENT OF MALTA
MINISTRY FOR EDUCATION, SPORT
YOUTH, RESEARCH AND INNOVATION

