ISBN: 978-9982-54-005-7

Printed by
Zambia Educational Publishing House

GRZ
NOT FOR SALE

Republic of Zambia

Ministry of Education

COMPUTER SCIENCE SYLLABI

SECONDARY EDUCATION ORDINARY LEVEL
FORM 1 – 4

DEVELOPED BY THE CURRICULUM DEVELOPMENT CENTRE
LUSAKA
2024

Republic of Zambia

MINISTRY OF EDUCATION

COMPUTER SCIENCE SYLLABI

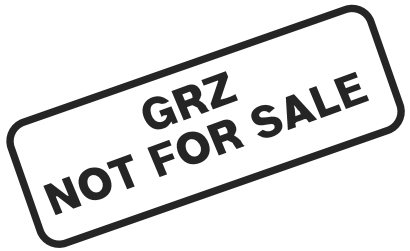
SECONDARY EDUCATION ORDINARY LEVEL

FORM 1-4

Developed by The Curriculum Development Centre

2024

ISBN: 978-9982-54-005-7



First Published 2024 by
Zambia Educational Publishing House
Light Industrial Area
Chishango Road
P. O. Box 32708
Lusaka, Zambia

Printed by:
Zambia Educational Publishing House (ZEPH)

TABLE OF CONTENTS

VISION ..... **vi**
PREFACE ..... **vii**
ACKNOWLEDGEMENT ..... **viii**
INTRODUCTION ..... **ix**

Rationale ix
Structure of the Syllabus ix
Teaching Methodology ix
Time Allocation xi
Assessment xi
Alternative to Practical Examinations xii

KEY COMPETENCES TO BE DEVELOPED ..... **xiv**

FORM 1 ..... **1**
1.1 Fundamentals of Computing 2
1.2 Productivity Tools: Word Processing 4
1.3 Data Representation 4
1.4 Computer Networks 5
1.5 Cybersecurity 5
1.6 Data Processing 6
1.7 Web Design 7
1.8 Digital Citizenship 7
1.9 Databases 8
1.10 Artificial Intelligence 8
1.11 Internet of Things (IOT) 9
1.12 Logic Gates 9

SUMMARY OF KEY COMPETENCES FOR FORM 1 ..... **10**

FORM 2..... 12

2.1 Productivity Tools..... 13
2.2 Data Representation..... 14
2.3 Introduction to Programming..... 14
2.4 Networking..... 15
2.5 Cybersecurity..... 15
2.6 Algorithms..... 15
2.7 Web Development 16
2.8 Database..... 17
2.9 Artificial Intelligence (AI)..... 17
2.10 Internet of Things..... 17
2.11 Robotics..... 17

SUMMARY OF KEY COMPETENCES FOR FORM 2..... 19

FORM 3..... 21

3.1 Productivity Tools: Presentation..... 22
3.2 Logic Gates..... 22
3.3 Networking..... 23
3.4. Cybersecurity 24
3.5 Programming..... 25
3.6 Web Development..... 25
3.6 Database..... 26
3.7 Artificial Intelligence..... 26
3.8 Internet of Things (IOT)..... 27
3.9 Mobile Application..... 27
3.10 Cloud Computing..... 28
3.11 Robotics..... 28

SUMMARY OF KEY COMPETENCES FOR FORM 3..... 30

FORM 4 ..... **32**
4.1 Computer Systems 33
4.2 Productivity Tools: Databases 33
4.3. Computer Networking 33
4.4 Programming 34
4.5 Mobile Application Development 35
4.6. Emerging Technologies 36
4.7 Cloud Computing 36

SUMMARY OF KEY COMPETENCES FOR FORM 4 38

APPENDICES 39
Scope and Sequence of Computer Science Syllabus (Form 1 to Form 4) 40

VISION

Quality, life long education for all which is accessible, inclusive and relevant to individual, national and global needs value systems.

PREFACE

It is with great enthusiasm and a profound sense of responsibility that I introduce this specialized Computer Science curriculum, designed for our secondary schools. As the Minister of Education, I am acutely aware of the pivotal role that technology plays in our rapidly evolving world. It is imperative that we equip our learners with the skills and knowledge necessary to thrive in the digital age.

The inclusion of Computer Science as a subject in our selected secondary schools marks a significant milestone in our education system. This initiative is driven by the recognition that computational thinking and digital literacy are essential competencies for the 21st century. Our aim is to foster a generation of innovators, problem-solvers, and critical thinkers who are adept at navigating and shaping the technological landscape.

This curriculum has been meticulously crafted to provide a comprehensive introduction to the principles and practices of Computer Science. Learners will delve into topics such as programming, data structures, algorithms, and cybersecurity. Additionally, they will explore the ethical and societal implications of technology, ensuring they become not only skilled technicians but also responsible digital citizens.

Our educators have been specially trained to deliver this curriculum, and our schools are equipped with the necessary resources to provide a stimulating and supportive learning environment. The interactive and hands-on nature of the course is designed to engage learners and spark their curiosity, encouraging them to pursue further studies and careers in the field of technology.

I am confident that this initiative will significantly enhance the quality of education in our schools and better prepare our learners for the challenges and opportunities of the future. I urge all learners to embrace this subject with enthusiasm and dedication, and I encourage parents and guardians to support their children's journey into the world of Computer Science.

Together, let us pave the way for a brighter, more technologically adept future for our nation.

Yours sincerely,

Joel Kamoko (Mr.)
Permanent Secretary – Educational Services
MINISTRY OF EDUCATION

ACKNOWLEDGEMENT

The development and introduction of the Computer Science curriculum for secondary schools represents a collaborative effort, and I am honored to acknowledge the contributions of all those involved in this significant undertaking. As the Director for Curriculum Development, I extend my deepest gratitude to everyone who has played a role in bringing this initiative to fruition.

First and foremost, I would like to express my sincere appreciation to the Minister of Education for their visionary leadership and unwavering support. Their commitment to enhancing the quality of education and preparing our learners for the future has been the driving force behind this project.

I am deeply grateful to the members of the Curriculum Development Committee, whose expertise and dedication have been instrumental in shaping this curriculum. Their extensive knowledge in the field of Computer Science and their commitment to educational excellence have ensured that this curriculum is both comprehensive and forward-thinking.

Special thanks are due to our team of educators and subject matter experts, who have worked tirelessly to design and refine the course content. Their insights and practical experience have been invaluable in creating a curriculum that is both engaging and relevant to our learners.

We also owe a debt of gratitude to our partner institutions and organizations, whose collaboration has provided essential resources and support. Their contributions have enabled us to equip our schools with the necessary tools and technology to deliver this curriculum effectively.

Furthermore, I would like to acknowledge the feedback and support from the pilot schools and their communities. The insights gained from their experiences have been crucial in fine-tuning the curriculum to meet the needs of our learners.

Finally, I extend my heartfelt thanks to the learners, parents, and guardians for their enthusiasm and support. It is your dedication to learning and your trust in our education system that inspire us to continually strive for excellence.

Together, we have embarked on a journey to empower our learners with the knowledge and skills needed to thrive in the digital age. I am confident that this curriculum will not only enhance their educational experience but also open doors to exciting opportunities in the world of technology.

Thank you all for your invaluable contributions and support.

Sincerely,

Charles Ndakala (Dr.)
Director - Curriculum Development
MINISTRY OF EDUCATION

INTRODUCTION

The Computer Science curriculum for secondary schools has been developed to equip learners with essential skills and knowledge in the field of technology. Recognizing the growing importance of digital literacy, this subject is designed to foster critical thinking, problem-solving, and creativity through a comprehensive exploration of computing principles and practices. While optional, this course is available to selected schools, providing learners with the opportunity to delve into the world of Computer Science and prepare for future technological advancements.

Rationale

Teaching computer science at the secondary school level is supported by several key rationales:

1. **Digital Literacy**: In an increasingly digital and technology-driven world, computer science education provides learners with essential digital literacy skills. Understanding the fundamentals of computing, programming, and technology empowers learners to navigate and participate in a technology-driven society.
2. **Problem-Solving Skills**: Computer science encourages the development of critical thinking and problem-solving skills. Programming and algorithmic thinking require learners to break down complex problems into manageable components, analyze them, and develop systematic solutions.
3. **Preparation for the Future Workforce:** As technology continues to advance, there is a growing demand for individuals with computer science skills in the workforce. Teaching computer science at the secondary level prepares learners for future careers in technology-related fields, which are diverse and ever-expanding.
4. **Promotion of Creativity**: Computer science fosters creativity by enabling learners to design and create their own digital solutions. Programming allows learners to express their creativity through the development of software, games, applications, and more.
5. **Understanding Technology's Impact**: Computer science education goes beyond coding; it includes understanding the societal impact of technology. Learners learn about ethical considerations, privacy issues, and the broader implications of technological advancements on society.
6. **Innovation and Entrepreneurship:** Computer science education equips learners with the skills needed to innovate and create new technologies. It also fosters an entrepreneurial mindset, encouraging learners to explore and develop their own tech-based ideas and solutions.
7. **Global Competitiveness**: In a globalized world, proficiency in computer science enhances a country's competitiveness. Countries with a strong emphasis on computer science education are better positioned to excel in technology-driven industries and contribute to innovation on a global scale.
8. **Cross-Disciplinary Learning**: Computer science is not limited to its own domain; it intersects with various other disciplines. Integrating computer science into secondary education provides opportunities for cross-disciplinary learning, where learners can apply computational thinking to solve problems in different subject areas.
9. **Equity and Inclusion:** Offering computer science education at the secondary level promotes equity by ensuring that learners from diverse backgrounds have access to the skills and knowledge necessary for participation in the digital age. This helps bridge the digital divide and fosters inclusivity in the tech industry.

10. **Preparation for Advanced Studies:** For learners interested in pursuing higher education in computer science or related fields, exposure to computer science at the secondary level provides a solid foundation. It helps learners make informed decisions about their academic and career paths.

In summary, teaching computer science at the secondary school level is essential for equipping learners with the skills, knowledge, and mindset needed to thrive in a technology-driven world and contribute meaningfully to society.

Structure of the Syllabus

The syllabus is structured to provide a balanced blend of theoretical knowledge and practical skills. It is divided into the following key modules:

- Foundations of Computer Science: Introduction to basic concepts, history of computing, and the evolution of technology.
- Programming: Learning programming languages and developing coding skills, with a focus on problem-solving and algorithmic thinking.
- Data Structures and Algorithms: Understanding how data is organized, processed, and optimized for efficient computation.
- Systems and Networks: Exploring computer hardware, software, and networking principles, including the fundamentals of operating systems and network configurations.
- Cybersecurity: Basics of protecting information and systems from digital threats, covering topics like encryption, threat detection, and secure communication.
- Ethical and Societal Implications: Discussing the impact of technology on society, ethical considerations in computing, and responsible digital citizenship.
- Web Technologies: Introduction to web development, including HTML, CSS and JavaScript
- Mobile Application Development: Fundamentals of creating mobile applications, focusing on user interface design, platform-specific development, and app testing.
- Artificial Intelligence (AI) and Machine Learning: Introduction to AI concepts, machine learning algorithms, and their applications in real-world scenarios.
- Internet of Things (IoT): Exploring IoT systems, devices, and their applications, including data management and communication protocols.
- Cloud Computing: Understanding cloud services, deployment models (IaaS, PaaS, SaaS), and cloud security practices.
- Database Management: Principles of database design, SQL query writing, and data management techniques, including relational and non-relational databases.
- Robotics: Introduction to robotics, covering the basics of robot design, programming, and the integration of sensors and actuators.
- Emerging Technologies: Exploration of the latest trends and advancements in technology, such as blockchain, quantum computing, and augmented/virtual reality.

Teaching Methodology

The teaching methodology emphasizes interactive and learner-centered learning approaches, including:

1. **Project-Based Learning (PBL):**
   - Design hands-on projects that require learners to apply programming and problem-solving skills.
   - Projects could include developing software applications, games, or websites.
2. **Collaborative Coding:**
   - Encourage collaborative coding exercises where learners work together on coding challenges.
   - Use version control systems to facilitate teamwork and code collaboration.
3. **Flipped Classroom:**
   - Provide instructional content, such as tutorials outside of class.
   - Use class time for hands-on activities, coding practice, and addressing learner questions.
4. **Interactive Coding Platforms:**
   - Utilize online platforms that allow learners to practice coding in an interactive and engaging environment.
   - Platforms can provide instant feedback and track learner progress.
5. **Real-World Applications:**
   - Integrate real-world scenarios and case studies into lessons.
   - Show how computer science is applied in various industries and professions.
6. **Guest Speakers and Industry Exposure:**
   - Invite guest speakers from the tech industry to share insights and experiences.
   - Arrange field trips or virtual visits to tech companies to expose learners to industry practices.

Time Allocation

The course is designed to be flexible, allowing schools to integrate it into their existing schedules. Recommended time allocation is:

- Weekly Classes: Three double periods per week, each period lasting 40 minutes.
- Practical Sessions: Additional laboratory or computer lab sessions of two triple periods to reinforce learning through hands-on experience.
- Project Work: Dedicated time for learners to work on individual or group projects running from form 2 to form 4.

Assessment

Assessment in the Computer Science curriculum is designed to evaluate both theoretical understanding and practical skills. Assessment methods include:
The following is an assessment overview for computer science

1. **Project Assessments:**
   - Evaluate learners based on the completion and quality of their programming projects.
   - Consider creativity, functionality, and adherence to best practices.
   - Assessment of individual and group projects based on creativity, functionality, and presentation.
2. **Coding Assignments and Tests:**
   - Regular coding assignments and tests to assess learners' understanding of programming concepts and syntax.
   - Assess both correctness and efficiency of code.
3. **Problem-Solving Exercises:**
   - Assess learners' ability to apply computational thinking and problem-solving skills to new challenges.
   - Include open-ended questions that require critical thinking.
   - Regular quizzes, assignments, and class participation.
4. **Class Participation and Collaboration:**
   - Assess engagement in class discussions, collaboration on coding exercises, and willingness to share knowledge with peers.
5. **Portfolio Assessment:**
   - Have learners maintain a portfolio showcasing their coding projects, solutions, and reflections on their learning journey.
6. **Final Project Presentation:**
   - Culminate the course with a final project presentation where learners demonstrate their skills, explain their project choices, and answer questions.
7. **Peer Review**:
   - Incorporate peer review in coding projects, encouraging learners to provide constructive feedback on each other's work.
8. **Written Examinations:**
   - Assess theoretical knowledge through written examinations covering topics such as algorithms, data structures, and computer science concepts.
   - Hands-on tests and exams to evaluate programming and problem-solving skills.
   - Final Examination: A comprehensive written final examination covering all modules of the syllabus.

Alternative to Practical Examinations

For schools that may not have adequate computers for all learners, alternative assessment methods can be employed to ensure that practical skills are still evaluated effectively. These alternatives include:

Simulation Software

Utilize simulation software that mimics real-world computing environments, allowing learners to practice coding, data management, and other COMPUTER SCIENCE skills on fewer computers or even on personal devices outside of school.

Paper-Based Practical Tasks

Design paper-based practical tasks where learners outline step-by-step processes for solving problems using Computer Science tools. These tasks can include writing code snippets, designing network diagrams, or planning data management strategies.

Project-Based Assessments

Assign project-based assessments where learners work on theoretical projects that require them to plan and design Computer Science solutions.

These projects can be assessed based on their understanding of concepts, problem-solving approaches, and creativity.

Oral Examinations

Conduct oral examinations where learners explain how they would perform certain practical tasks. This can include describing coding logic, explaining how to secure a network, or detailing steps for data analysis.

Peer Reviews and Group Work

Encourage peer reviews and group work to leverage collaborative learning. Learners can present their work to classmates, who then provide feedback based on set criteria. This method promotes understanding through teaching and discussion.

Use of Computer Science Labs in Shifts

If a limited number of computers are available, organize learners into shifts to ensure each learner gets adequate hands-on time. This can be supplemented with guided instruction during non-computer times.

Portfolio Assessments

Have learners compile a portfolio of their work throughout the course. This portfolio can include written assignments, project plans, and reflections on their learning process. It provides a comprehensive view of their understanding and progress.

By implementing these alternatives, schools can ensure that all learners are assessed fairly and comprehensively, even in environments with limited resources. These methods aim to maintain the integrity and effectiveness of the Computer Science curriculum while providing equitable opportunities for all learners.

KEY COMPETENCES TO BE DEVELOPED

The following is a summary of the key competences achieved at the end of the four-year Ordinary Level Computer Science course:

Competence	Descriptor
Technical Proficiency	Comprehensive understanding of computer components and their functions, and ability to perform basic and advanced troubleshooting and maintenance.
Programming Skills	Proficiency in writing, debugging, and optimizing programs using various programming languages, including Scratch, Python, and JavaScript, with an understanding of object-oriented programming concepts.
Data Structure Skills	Ability to create, manipulate, and apply basic and advanced data structures, such as arrays, linked lists, stacks, queues, dictionaries, sets, and trees, in problem-solving.
Algorithmic Thinking	Competence in designing, implementing, and analyzing both basic and advanced algorithms, including sorting, searching, recursion, and dynamic programming, with a focus on optimization and complexity analysis.
Web Development Skills	Proficiency in designing and developing web pages using HTML, CSS, and JavaScript, including the ability to create responsive designs and interactive web elements.
Database Management Skills	Comprehensive understanding of database concepts, ability to perform basic and advanced SQL queries, design and normalize database schemas, and develop database-driven applications.
Networking Knowledge	Knowledge of basic and advanced networking concepts, components, protocols, and configurations, with the ability to set up, manage, and troubleshoot computer networks.
Cybersecurity Awareness	Understanding of fundamental and advanced cybersecurity principles, ability to identify and mitigate various cyber threats, implement network security measures, and apply cybersecurity frameworks.
Digital Citizenship	Awareness of ethical and responsible use of technology, understanding of digital footprints and online privacy management, and promotion of safe and responsible digital behavior.
Application Software Skills	Proficiency in using word processing and spreadsheet software for various tasks, including creating and formatting documents, using advanced spreadsheet functions, and developing database-driven applications.
Critical Thinking and Problem-Solving	Enhanced problem-solving skills through algorithmic thinking, data structure manipulation, and practical application of programming concepts to solve complex problems.

Competence	Descriptor
Collaboration and Communication	Improved teamwork and communication skills through collaborative projects, peer reviews, and presentations, fostering an environment of cooperative learning and knowledge sharing.
Ethical and Societal Awareness	Awareness of the ethical implications of technology use and its societal impact, promoting the responsible and informed use of ICT.

The table provides a clear overview of the key competences that learners will achieve upon completing the four-year course, equipping them with a well-rounded skill set for further education and careers in computer science and related fields.

FORM 1

TOPIC	SUB-TOPIC	SPECIFIC COMPETENCES	LEARNING ACTIVITIES	EXPECTED STANDARDS
1.1 FUNDAMENTALS OF COMPUTING	**1.1.1 Computer Basics**	1.1.1.1 Demonstrate understanding of essential computer software and hardware	• Discussing history and evolution of computers • Classifying types of computers • Analyzing components of computer hardware • Classifying types of software • Operating essential computer software and hardware	• Understanding of essential computer hardware and software components demonstrated correctly
	1.1.2 Operating system and file management	1.1.2.1 Demonstrate proper management of operating system and files	• Discussing functions of an operating system • Demonstrating proper handling of operating system • Installing and updating the operating system • Applying file management tasks such as copying, pasting, renaming and deleting.	Proper management of operating system and files demonstrated correctly

TOPIC	SUB-TOPIC	SPECIFIC COMPETENCES	LEARNING ACTIVITIES	EXPECTED STANDARDS
			• Organizing files and folders using appropriate naming convention and directories structures • Utilizing search functions to locate files and programs efficiently.	
	1.1.3 Application Software	1.1.3.1 Operate Commonly used application software	• Discussing application software and their functions • Identifying commonly used application software • Demonstrating installation of application software (*VLC, Chrome, Adobe...)*	Commonly used application software appropriately operated.
	1.1.4 Basic trouble shooting	1.1.4.1 Diagnose and resolve common computer problems	• Analyzing computer error messages • Isolating common computer problems • Accessing help resources such as built-in system guides to resolve technical issues • Resolving identified problems • Recovering lost data or from corrupted/damaged storage devices	Common computer problems effectively resolved.

TOPIC	SUB-TOPIC	SPECIFIC COMPETENCES	LEARNING ACTIVITIES	EXPECTED STANDARDS
1.2 PRODUCTIVITY TOOLS: WORD PROCESSING	**1.2.1 Word Processing**	1.2.1.1 Create Documents using Word Processing Software	• Exploring the word processing interface. • Describing the basic functions of word processing application. • Demonstrating correct typing skills • Creating and formatting documents using text editing features such as font styles, paragraph alignment and spacing. • Inserting and manipulating visual elements like tables images and charts within a table. • Applying templets to design professional documents • Utilizing tools for reviewing and editing documents • Importing/ Exporting documents in different formats	Documents successfully created using word processing software
1.3 DATA REPRESENTATION	**1.3.1 Number Systems**	1.3.1.1 Application of number systems in computing	• Discussing the basic concepts of the number systems including (binary, Decimal) • Exploring the role of number systems in computer systems	• Number Systems correctly applied in computing

TOPIC	SUB-TOPIC	SPECIFIC COMPETENCES	LEARNING ACTIVITIES	EXPECTED STANDARDS
			• Converting of number systems • Analyzing binary coded to understand its representation in real-world application • Demonstrating the use of binary numbers in programming and hardware configurations.	
1.4 COMPUTER NETWORKS	**1.4.1 Network Design and Implementation**	1.4.1.1 Configure and Install Networks	• Discuss Basic concepts of computer networks • Identifying basic networking devices and software • Discussing types of connections • Configure a local area network (LAN) using basic networking devices such as routers and switches • Implementing basic network sharing setting for networking devices such as printers, files, folders … across the network	Computer networks correctly configured and installed
1.5 CYBERSECURITY	**1.5.1 Fundamentals of Cyber Security**	1.5.1.1 Apply online safety and security measures	• Discussing terminologies and key concepts in cyber-security • Identifying different types of cyber threats such as	Online Safety and Security measures successfully applied

TOPIC	SUB-TOPIC	SPECIFIC COMPETENCES	LEARNING ACTIVITIES	EXPECTED STANDARDS
			hacking, phishing, social engineering attacks… • Detecting potential cyber threats using monitoring tools and logs • Describing the importance of cyber security in the digital world. • Implementing strong password policies and multifactor authentication to protect user accounts	
1.6 DATA PROCESSING	**1.6.1 Data collection and cleaning**	1.6.1.1 Collect and clean data	• Discussing key concepts of data processing and cleaning • Exploring types of data in computing • Identifying data processing techniques • Discussing tools for capturing and cleaning data • Collecting data from multiple sources • Demonstrating how to clean and validate data • Designing visualization to present processed data *(ms power BI, tableau, google data studio ...)* • Developing testing protocols to ensure	Data effectively collected and cleaned

TOPIC	SUB-TOPIC	SPECIFIC COMPETENCES	LEARNING ACTIVITIES	EXPECTED STANDARDS
			accuracy and consistence of processed data	
1.7 WEB DESIGN	**1.7.1 Fundamentals of Web Design**	1.7.1.1 Design and Structure Web Pages using web design tools.	• Discussing the key concepts of Web Design • Identifying the components of a web page • Describing the roles of *(HTML, CSS, and JavaScripts ...)* in creating and designing web pages • Designing and structuring simple web pages using HTML5 and semantic tags	Simple web pages and structures successfully implemented
1.8 DIGITAL CITIZENSHIP	**1.8.1 Digital Ethics**	1.8.1.1 Demonstrate Ethical Behavior in a Digital Environment	• Discussing digital citizenship. • Demonstrating respectful communication in online interaction • Distinguish between fact, opinion, and bias in digital content • Demonstrate how to balance screen time with offline activities to maintain physical and mental well-being. • demonstrating a culture of inclusion, respect and fairness in online communities	Ethical Behavior in Digital Environment Demonstrated Appropriately

TOPIC	SUB-TOPIC	SPECIFIC COMPETENCES	LEARNING ACTIVITIES	EXPECTED STANDARDS
1.9 DATABASES	**1.9.1 Introduction to Databases**	1.9.1.1 Create and manage databases	• Defining databases • Discussing the key concepts of databases such as tables, records, fields and keys. • Differentiate types of databases • Describing the types of database models • Identifying popular DBMS software *(access, oracle, maria db ...)* • Create tables and define their structures using database tools.	Databases successfully created and managed
1.10 ARTIFICIAL INTELLIGENCE	**1.10.1 fundamentals of Artificial Intelligence (AI)**	1.10.1.1 Design Artificial Intelligence (AI) Models	• Defining AI • Discussing the core concepts of AI including machine learning, deep learning and natural language processing • Identifying the problems to be solved • Determining objectives of the model. • Collecting and preparing datasets for creating AI models • Using simple machine learning to create AI models.	Artificial Intelligence (AI) Models designed.

TOPIC	SUB-TOPIC	SPECIFIC COMPETENCES	LEARNING ACTIVITIES	EXPECTED STANDARDS
1.11 INTERNET OF THINGS (IOT)	**1.11.1 Basic Components of Internet Things (IoT)**	1.11.1.1 Demonstrate understanding of IoT components.	• Discussing the concepts of IoT • Identifying key components of IoT systems, such as sensors, actuators, microcontrollers and communication modules.	Understanding of IoT components demonstrated appropriately.
1.12 LOGIC GATES	**1.12.1 Introduction to Logic Gates**	1.12.1.1 Create Digital Control systems	• Discussing the concept of logic gates and their role in digital circuits • Identifying types of logic gates and their symbols • Analyzing combinational and sequential logic gates • Working with Truth Tables • Creating a simple digital control system such as alarm system, traffic lights…	• Digital control systems successfully created

SUMMARY OF KEY COMPETENCES FOR FORM 1

Competence	Descriptor
Fundamentals of Computing	Demonstrate an understanding of computer systems, their components, types, and software by identifying, classifying, and installing hardware and software components while explaining their roles and functions, maintain computer systems by identifying and resolving common hardware and software issues.
Productivity Tools: Word Processing	Effectively use a word processing application to create, format, and structure documents by applying text formatting, inserting objects, and utilizing proofreading and alignment tools.
Data Representation	Demonstrate numerical proficiency by identifying, converting between, and performing arithmetic operations on decimal and binary number systems.
Computer Networks	Utilize the internet effectively and safely by accessing basic services, navigating the web, managing an email account, and applying secure online practices.
Cybersecurity	Understand the concept and significance of cybersecurity by identifying its applications and emphasizing its importance in safeguarding the digital world.
Introduction to Programming	Develop interactive and dynamic programs in Scratch by navigating its interface, using motion and appearance controls, applying loops and decision-making blocks, and integrating events and triggers to animate sprites and enhance interactivity.
Data Processing	Using series of actions or operations on data to retrieve, transform, or classify it into a more usable form.
Web Design	Design and create basic web pages by understanding and applying foundational HTML concepts and tags.
Digital Citizenship	Exhibit responsible digital citizenship by practicing digital etiquette, understanding privacy rights and intellectual property, and addressing the impact of cyberbullying with preventive and responsive measures.
Databases	Understand and apply foundational database concepts by defining and explaining their purpose, identifying types and characteristics of databases, and demonstrating knowledge of data organization, management, and basic terminologies.
Artificial Intelligence	Demonstrate an understanding of Artificial Intelligence (AI) by explaining its principles, history, and evolution, recognizing its applications in everyday life, and analyzing its societal impact, while advocating for ethical AI practices and addressing potential risks.
Introduction to the Internet of Things (IoT)	Understand and explain the Internet of Things (IoT) by defining its significance, identifying its components, describing how IoT systems work, and recognizing its applications in daily life, while

	emphasizing the importance of connectivity, data transmission, and communication protocols in creating efficient and interconnected IoT ecosystem
Logic Gates	Demonstrate understanding of logic gates by defining their role, identifying their types and functions, constructing truth tables, deriving logical expressions, combining them to design circuits, and applying them to solve real-world problems.

FORM 2

TOPIC	SUB-TOPIC	SPECIFIC COMPETENCES	LEARNING ACTIVITIES	EXPECTED STANDARDS
2.1 PRODUCTIVITY TOOLS	**2.1.1 Spreadsheets**	2.1.1.1 Use spreadsheet for data entry and analysis	• Discussing the basic components of spreadsheets interface *(rows, cells, shortcuts…)* • Exploring the uses and benefits of spreadsheets in our daily lives • Performing arithmetic operations using basic formulas and functions • Filtering and sorting data to identify trends, patterns, or specific values. • Performing statistical analysis using functions *(countif, average, median ...)* • Creating charts to visualize data • Protecting cells, sheets, workbooks • Sharing spreadsheets using collaborative tools such as cloud-based tools • Creating and applying macros	• Entered and analyzed data correctly using spreadsheet.

TOPIC	SUB-TOPIC	SPECIFIC COMPETENCES	LEARNING ACTIVITIES	EXPECTED STANDARDS
			• Developing financial models, using *(what-if ...)* scenarios	
2.2 DATA REPRESENTATION	**2.2.1 Number Systems**	2.2.1.1 Apply Number System Operations	• Discussing concepts of Hexadecimal and Octal number systems • Converting numbers between systems • Converting text using ASCII	Number System Operations applied accordingly
2.3 INTRODUCTION TO PROGRAMMING	**2.3.1 Block based programing**	2.3.1.1 Construct programs using block-based software	• Exploring different types of block-based programming software *(Scratch ...)* • Describing the key features of block-based programming, such as drag-drop blocks and visual works • Identifying basic programing constructs, including loops, conditions, variables and events in block-based platforms. • Constructing simple programs by sequencing blocks to create specific outcomes including the use of loops, conditions, variables and events in block-based platforms.	Block-based programs constructed correctly.

TOPIC	SUB-TOPIC	SPECIFIC COMPETENCES	LEARNING ACTIVITIES	EXPECTED STANDARDS
2.4 NETWORKING	**2.4.1 Network Topologies**	2.4.1.1 Develop and Configure Network Topologies	• Discussing different types of Network topologies • Analyzing roles of network topology • Create network cables *(Ethernet, UTP, STP, FTP…)* • Designing network topologies. • Creating and configuring network topologies.	• Network topologies developed and Configured correctly
2.5 CYBER SECURITY	**2.5.1 Introduction to cybersecurity**	2.5.1.1 Utilize cybersecurity tools and technologies	• Discussing cyber security and its importance in digital systems • Investigating cyber vulnerabilities in operating system, software and hardware • Using cyber security tools and technologies • Developing policies to protect data, devices and networks from cyber threats.	Cybersecurity tools and technologies utilized accordingly
2.6 ALGORITHMS	**2.6.1 Fundamentals of Algorithms**	2.6.1.1 Design Algorithms	• Identifying different types of algorithms • Discussing characteristics of algorithms	Algorithms designed correctly

TOPIC	SUB-TOPIC	SPECIFIC COMPETENCES	LEARNING ACTIVITIES	EXPECTED STANDARDS
			• Designing flow charts and pseudocode • Using different types of algorithms design techniques	
2.7 WEB DEVELOPMENT	**2.7.1 Cascading Style Sheets (CSS) concepts**	2.7.1.1 Apply CSS concepts to websites	• Defining Cascading Style Sheets (CSS) and explain their purpose in web design. • Identifying the different types of CSS: inline, internal, and external. • Describing the basic structure of a CSS rule, including selectors, properties, and value • Using different CSS elements	CSS concepts applied to websites correctly
			• Using text properties to style fonts, including font-family, font-size, and color. • Applying background properties, such as background-color, background-image, and background-repeat. • Styling borders and spacing using properties like border, margin, and padding	

TOPIC	SUB-TOPIC	SPECIFIC COMPETENCES	LEARNING ACTIVITIES	EXPECTED STANDARDS
2.8 DATABASE	**2.8.1 Database Design and Creation**	2.8.1.1 Design and create databases using SQL	• Discussing Structured Query Language *(SQL)* concepts. • Writing SQL scripts to create and modify database structures	• Databases designed and created using SQL correctly
2.9 ARTIFICIAL INTELLIGENCE (AI)	**2.9.1 AI Models**	2.9.1.1 Develop AI Models	• Exploring the problems to be solved • Selecting suitable models • Designing and Developing the model • Evaluating the models	• AI models developed successfully
2.10 INTERNET OF THINGS	**2.10.1 IoT Model Development**	2.10.1.1 Develop an IoT Model	• Discussing different development platforms • Exploring the user interfaces • Creating codes for IoTs • Connecting IoT devices using wireless communication protocols like wifi, Bluetooth, Zigbee, or LoRaWAN. • Building a basic IoT system using sensors, microcontrollers (Arduino, Raspberry Pi), actuators, LEDs	• IoT Model developed accordingly.
2.11 ROBOTICS	**2.11.1 fundamentals of Robotics**	2.11.1.1 Design Simple Robots	• Discussing the basic concepts of Robotics	Simple Robots correctly designed

TOPIC	SUB-TOPIC	SPECIFIC COMPETENCES	LEARNING ACTIVITIES	EXPECTED STANDARDS
			including ethical use of robots. • Classifying types of robots • Identifying the core components of a robot mechanism in robotics • Designing a basic robotic system considering functionality, form, and purpose	

SUMMARY OF KEY COMPETENCES FOR FORM 2

Competences	Descriptor
Productivity Tools: Spreadsheets	Master the use of spreadsheet software by creating, editing, and formatting data, utilizing formulas and functions for calculations, analyzing data through sorting, filtering, and charting, and applying protection and sharing features to manage access and collaboration.
Data Representation	Demonstrate proficiency in number system conversions by accurately converting between number systems.
Networking	Understand and explain computer networks by defining their purpose, identifying key components, and differentiating between various types and topologies. Demonstrate knowledge of common networking devices, their roles, and the concept of IP addresses, including the differences between IPv4 and IPv6.
Cybersecurity	Identify and analyze various cyber threats, understand their impact on personal and organizational systems, and develop policies to protect data, devices and networks.
Algorithms	Apply problem-solving techniques and algorithms to real-world and computational problems by designing, interpreting, and optimizing algorithms, using flowcharts and pseudocode to represent solutions, and refining algorithms to improve efficiency and handle cases.
Programming	Understand and apply programming concepts in Python by installing necessary software, configuring IDEs, and executing scripts. Demonstrate proficiency in Python syntax, data types, and basic operations, while using conditional statements to implement decision-making in programs.
Web Development	Master web design principles using CSS by styling HTML elements through inline, internal, and external CSS, applying various selectors, and managing layout and spacing using the CSS box model. Demonstrate the ability to create responsive designs using media queries and flexible layouts, and design user-friendly navigation bars while understanding the importance of accessibility and usability in modern web development.
Database-Structured Query Language (SQL)	Demonstrate proficiency in using SQL to manage and organize data by writing queries involving JOIN, GROUP BY, and ORDER BY commands to retrieve and analyze datasets, while applying database security principles such as user authentication and data encryption; implement backup and recovery operations, understanding various types of backups; and apply deep learning concepts and big data analysis techniques for tasks like fraud detection and recommendation systems, including basic data preprocessing and visualization.

Artificial Intelligence (AI)	Understand and explain the different types and algorithms of AI, including their applications in machine learning and problem-solving. Demonstrate knowledge of how AI is applied in games and entertainment, particularly in enhancing user experiences through pathfinding, content personalization, and AI-generated content.
Internet of Things (IoT)	Understand and explain the components of IoT architecture, including devices, gateways, networks, and cloud platforms, and how they interact with the physical environment. Demonstrate knowledge of IoT data collection processes, communication protocols, and security measures to protect against threats, ensuring privacy and data protection in IoT systems.
Mobile Application	Understand the different types of mobile applications (native, web, hybrid) and their platforms (e.g., Android, iOS), and explain their core features and functionalities. Demonstrate knowledge of the mobile app development process, including ideation, development, testing, and deployment, while ensuring security, ethical practices, and user privacy. Apply an understanding of mobile application categories and their impact on addressing user needs, promoting ethical app usage and respect for intellectual property rights.
Cloud Computing	Understand the core concepts and models of cloud computing, including on-demand access, scalability, and resource pooling, and differentiate between cloud deployment models and service models. Demonstrate knowledge of the benefits of cloud computing, such as cost-efficiency, accessibility, and environmental impact, and explain the role of cloud storage in remote data access and sharing, highlighting key features like synchronization and data backup.
Robotics	Understand the significance of robotics in technology and society, recognizing its historical development and key milestones. Demonstrate knowledge of the basic components of a robot, including sensors, actuators, controllers, and power sources, and their roles in robotic functioning. Classify robots based on functionality, design, and applications, and differentiate between autonomous and remote-controlled robots. Gain practical experience by assembling a simple robot and programming it to execute basic movement tasks.

FORM 3

TOPIC	SUB-TOPIC	SPECIFIC COMPETENCES	LEARNING ACTIVITIES	EXPECTED STANDARDS
3.1 PRODUCTIVITY TOOLS: PRESENTATION	**3.1.1 Presentation Software**	3.1.1.1 Design and Create Presentations	• Exploring presentation software • Identifying different types of presentation software • Exploring the presentation software • Creating presentations using templets and themes • Applying design principals • Designing and laying-out presentations • Embed hyperlinks and other advanced interactive elements.	• Presentations designed and created correctly.
	3.1.2 Desktop Publishing software	3.1.2.1 Create and Format publications	• Discussing and identifying different types of publishing software *(MS publisher, Adobe, Canva ...)* • Exploring the interface for publishing software • Design and laying-out canvas or pages • Creating and formatting publications	• Publications Created and Formatted correctly.
3.2 LOGIC GATES	**3.2.1 Application of Logic Gates**	3.2.1.1 Design Digital Circuits	• Simulating logic gate circuits using digital design software *(Academo, digital works, circuit verse ...)*	• Digital circuits designed correctly.

TOPIC	SUB-TOPIC	SPECIFIC COMPETENCES	LEARNING ACTIVITIES	EXPECTED STANDARDS
			• Analyzing the output of logic circuits for various inputs • Implementing Boolean expressions using logic gates • Integrating logic gates into microcontroller-based systems for automation • Designing sequential circuits for time dependent applications *(timers, counters ...)*	
3.3 NETWORKING	**3.3.1 The Internet**	3.3.1.1 Create and Share Content online	• Exploring the internet and its key components *(search engines, servers, clients ...)* • Discussing internet services • Setting up accounts on different platforms for professional and personal use. • Engaging in video conferencing and webinars using platforms such as zoom, google meet … • Creating and sharing online content using different internet platforms.	Online content created and shared appropriately

TOPIC	SUB-TOPIC	SPECIFIC COMPETENCES	LEARNING ACTIVITIES	EXPECTED STANDARDS
3.4. CYBERSECURITY	**3.4.1. Data Protection and Privacy**	3.4.1.1. Mitigate and Respond to Cyber Threats	• Discussing data protection laws *(GDPR, CCPA ...)* in handling user information. • Analysing the impact of online actions on personal and professional reputation • Applying encryption techniques to secure sensitive data during storage and transmission *(symmetric and asymmetric encryption)*. • Employing Virtual Private Networks (VPNs) to secure remote connection. • Develop security policies and guidelines to ensure compliance with the best practices	Cyber threats Mitigated and Responded to correctly.

TOPIC	SUB-TOPIC	SPECIFIC COMPETENCES	LEARNING ACTIVITIES	EXPECTED STANDARDS
3.5 PROGRAMMING	**3.5.1 Python Programming**	3.5.1.1 Use Python to Code programs	• Identifying various Integrated Development Environments (IDEs) *(Jupiter, PyCharm ...)* • Exploring python syntax and control structures • Creating simple programs *(games...)*	Python used to code programs correctly
	3.5.2 Coding in Python	3.5.2.1 Develop Practical Python Applications	• Installing and configuring python, IDEs *(PyCharm, VS-Codes, Jupyter...)* • Writing Python scripts using conditional statements *(if, else, else if ...)* to manage decision-making in programs • Creating python projects *(task automation, data analysis, web development...)*	Practical Python Applications Developed accordingly.
3.6 WEB DEVELOPMENT	**3.6.1 Web Interactivity**	3.6.1.1 Develop Dynamic Interactive Web Pages	• Discussing Concepts of Java Script • Discussing the role of JavaScript in Web Development • Analyzing JavaScript and other web technologies • Integrate Object Oriented Programming (OOP) principles in Java Script	Dynamic Interactive Web Pages developed accordingly

TOPIC	SUB-TOPIC	SPECIFIC COMPETENCES	LEARNING ACTIVITIES	EXPECTED STANDARDS
			• Creating animations using JavaScript and CSS transitions	
3.7 DATABASE	**3.7.1 Database Management**	3.7.1.1 Manage Databases	• Discussing key components of database management • Configure user roles and permissions to control access to database resources • Upgrade and migrate databases to newer versions or platforms • Identifying and resolving database querying issues • Applying the first, second, and third normal forms (1NF, 2NF, 3NF) to organize data. • Analyzing well-designed and poorly designed database • Integrate database with programming languages *(python, Java, PHP…)*	• Database managed correctly.
3.8 ARTIFICIAL INTELLIGENCE	**3.8.1. AI Models**	3.8.1.1 Deploy AI Models	• Discussing deployment tools • Training and testing models	• AI models deployed correctly

TOPIC	SUB-TOPIC	SPECIFIC COMPETENCES	LEARNING ACTIVITIES	EXPECTED STANDARDS
			• Deploying models • Monitoring and maintaining models	
3.9 INTERNET OF THINGS (IOT)	**3.9.1 Application of Internet of Things (IoT)**	3.9.1.1 Create IoT solutions	• Developing IoT dashboards to visualize sensor data in real time • Exploring real-world IoT application *(Transportation, education, healthcare, agriculture, smart homes…)* • Developing IoT solutions for a chosen sector *(Transportation, education, healthcare, agriculture, smart homes…)*	• IoT solutions Created accordingly.
3.10 MOBILE APPLICATION	**3.10.1 Introduction to Mobile Applications**	3.10.1.1 Create Mobile Applications.	• Exploring mobile application and explaining its use in daily life. • Discussing native, web, and hybrid mobile applications. • Identifying core features of mobile applications • Discussing common tools and programming languages used in app development	• Mobile Applications Created Correctly.

TOPIC	SUB-TOPIC	SPECIFIC COMPETENCES	LEARNING ACTIVITIES	EXPECTED STANDARDS
			• Analyzing common security risks in mobile applications *(malware, data breaches ...)*. • Describe strategies for protecting personal data when using mobile apps. • Creating Mobile Apps	
3.11 CLOUD COMPUTING	**3.11.1 Fundamentals of Cloud computing**	3.11.1.1 Develop and Manage cloud computing models	• Defining the key concepts of cloud computing • Discussing the benefits and challenges of cloud computing for business and individuals • Designing cloud computing models • Developing cloud computing models • Managing three main cloud services using modeling tools such as AWS -EC2, google cloud tool, Microsoft 360, Azure…	• Cloud computing models correctly developed and managed
3.12 ROBOTICS	**3.12.1 Mechanical and Electronic Systems**	3.12.1.1 Implement control systems	• Designing robotic structures using CAD Software. • Installing and aligning robotic motors and gears • Setting up power supply systems for robots • Testing electronic circuits to ensure proper	• Control systems implemented accordingly.

TOPIC	SUB-TOPIC	SPECIFIC COMPETENCES	LEARNING ACTIVITIES	EXPECTED STANDARDS
			functioning of robotic components • Writing programs to control movement using languages *(python, C++...)* • Combining mechanical, electronic and software components into a functional robotic system • Integrating robotics with internet of things (IoT) systems for remote monitoring. • Inspecting robotic systems to identify and address potential issues	

SUMMARY OF KEY COMPETENCES FOR FORM 3

Competence	Descriptor
Computer Systems	Demonstrate the ability to analyze and differentiate between batch processing, real-time processing, and online transaction processing systems, evaluating their functionalities, applications, and industry use cases to select the most suitable system based on efficiency, speed, and accuracy, while proposing improvements for enhanced performance and adaptability.
Productivity Tools: Presentation	Demonstrate proficiency in designing and delivering impactful presentations by creating visually engaging slides with appropriate design elements, incorporating multimedia for enhanced engagement, and utilizing purposeful transitions and animations. Apply effective delivery techniques, manage audience interactions, and leverage collaboration tools for co-creation and version control to ensure professional and cohesive presentations.
Productivity tools: Publisher	Demonstrate proficiency in Microsoft Publisher by navigating its interface, creating and customizing publications, formatting text, images, and shapes, utilizing Master Pages for consistent design, and preparing publications for professional printing or digital distribution.
Logic Gates	Demonstrate the ability to apply logic gates in digital circuits and computing systems
Networking Basics	Demonstrate how the internet can be used in industries such as entertainment, education health, business, information …
Cybersecurity	Demonstrate the ability to implement fundamental cybersecurity practices, including encryption, wireless network security, digital footprint management, and Two-Factor Authentication (2FA), to safeguard personal and organizational data against common threats.
Programming in Python	Demonstrate proficiency in Python programming by effectively applying control structures, loops, functions, error handling, and data structures like lists and tuples to create efficient, reusable, and robust code solutions.
Web Design	Demonstrate the ability to Analyzing JavaScript and other web technologies, Integrate Object Oriented Programming (OOP) principles in Java Script, Creating simple animations using JavaScript and CSS transitions
Database Management	Demonstrate the ability to design and manage relational databases by applying normalization techniques, creating entity-relationship diagrams (ERDs), defining relationships, and using SQL to

	retrieve, manipulate, and maintain data efficiently, while leveraging constraints and indexing to ensure data integrity and optimize performance.
Artificial Intelligence	Demonstrate the ability to explain and apply the principles of Natural Language Processing (NLP) and Computer Vision in AI, analyze their roles in various AI applications, and develop basic AI programs to solve problems, while evaluating the effectiveness of these programs in achieving desired outcomes.
Internet of Things (IoT)	Demonstrate the ability to Develope IoT dashboards to visualize sensor data in real time, Exploring real-world IoT application such as transportation, education, healthcare, agriculture, smart home, Presenting IoT Projects to stakeholders
Mobile Application Development	Demonstrate the ability to Set up and Design environments using tools such as android studio, Xcode, VS-Code and Writing codes to implement app functionality using programming languages such as java, kotlin, Design and building responsive app interfaces that adapts to screen sizes and orientations, Integrating device features such as GPS, Camera, and sensors into mobile apps. Building offline capabilities using catching or local databases
Cloud Computing	Demonstrate how to Launch cloud-based application and services including databases, storage and computer resources such as DynmoDB, Firestore, Azure , Design cloud-native application using micro services architecture, Building APIs for cloud-based services to integrate with other systems, Integrate cloud services with Internet of Things (IoT) devices for real time data processing, Design and implement cloud migration strategies
Robotics	Demonstrate proficiency in robotics by understanding key terminology, identifying and explaining the role of various sensors in robotic systems, writing and debugging basic programs to control robots, and assembling and testing basic robots while analyzing the applications and implications of robotics in various fields.

FORM 4

TOPIC	SUB-TOPIC	SPECIFIC COMPETENCES	LEARNING ACTIVITIES	EXPECTED STANDARDS
4.1 COMPUTER SYSTEMS	**4.1.1 Fundamentals of Computer Systems**	4.1.1.1 Demonstrate understanding of computer systems	• Discussing the key components of computer systems • Exploring different types of computer systems and their functions *(expert system, batch processing system, control system ...)*	• Understanding of computer systems demonstrated accordingly
4.2 PRODUCTIVITY TOOLS: DATABASES	**4.2.1 Database Management Systems**	4.2.1.1 Create and Manage Databases	• Discussing the purpose and functions of a database management system (DBMS) • Identifying common DBMS software *(MS Access ...)* • Installing and configuring DBMS software *(MS Access ...)* • Creating databases • Inserting, updating, and deleting data in tables	• Databases created and managed accordingly.
4.3. COMPUTER NETWORKING	**4.3.1. Network Design and Configurations**	4.3.1.1 Design and Configure Computer Networks	• Discussing key networking concepts including OSI layers and TCP/IP models. • Explaining the role of DNS, IP addressing and MAC addressing in networking.	• Computer Networks correctly designed and configured

TOPIC	SUB-TOPIC	SPECIFIC COMPETENCES	LEARNING ACTIVITIES	EXPECTED STANDARDS
			• Discussing key networking protocols *(TCP, UDP , HTTP, FTP, DHCP ...)* • Implementing VLANs to segment network traffic for efficiency and security • Designing Subnets to optimize IP address allocation using subnet mask. • Configuring IP addresses (static and dynamic) on devices • Configuring Virtual Private Networks (VPNs) • Setting up Quality of Service (QoS) to prioritize specific network traffic. • Implementing encryption protocols such WPA3 for wireless networks	
4.4 PROGRAMMING	**4.4.1. Programming C++**	4.4.1.1 Develop C++ programs	• Discussing the main concepts of C++ programing languages • Describing the structures of C++ program *(headers, functions, main() ...)* • Discussing tools used to create C++ programming languages	• C++ Programs developed accordingly.

TOPIC	SUB-TOPIC	SPECIFIC COMPETENCES	LEARNING ACTIVITIES	EXPECTED STANDARDS
			• Writing simple programs using basic syntax and structures in C++ • Creating programs that uses loops • Developing programs that uses arrays and strings for data storage and manipulations • Defining Classes and objects in C++ to encapsulate data and behavior • Applying techniques to identify and fix errors • Writing C++ programs for object oriented applications like simulations and GUIs.	
4.5 MOBILE APPLICATION DEVELOPMENT	**4.5.1 Mobile Application Design**	4.5.1.1 Develop mobile applications .	• Setting up and Designing environments using tools (android studio, Xcode, VS-Code …) • Writing codes to implement app functionality using programming languages *(java, kotlin ...)* • Design and building responsive app interfaces that adapts to screen sizes and orientations.	• Mobile applications developed accordingly.

TOPIC	SUB-TOPIC	SPECIFIC COMPETENCES	LEARNING ACTIVITIES	EXPECTED STANDARDS
			• Integrating device features *(GPS, Camera, and sensors into mobile apps…)* • Building offline capabilities using catching or local databases	
4.6. EMERGING TECHNOLOGIES	**4.6.1. Fundamentals of Emerging Technologies**	4.6.1.1 Demonstrate understanding of Emerging Technologies	• Exploring emerging technologies and their role in transforming industries • Identifying key emerging technologies *(AI, IoT, AR, VR…)* • Applying Emerging Technologies into industries • Discussing the potential impact of emerging technologies on society, economy, and daily life. • Discussing the concept of 3D printing and its role in manufacturing and design.	• Understanding of Emerging Technologies demonstrated appropriately
4.7 CLOUD COMPUTING	**4.7.1 Cloud Deployment**	4.7.1.1 Deploy Cloud Platforms	• Launching cloud-based application and services including databases, storage and computer resources *(DynmoDB, Firestore, Azure…)* • Designing cloud-native application using micro services architecture.	• Cloud platforms deployed correctly

TOPIC	SUB-TOPIC	SPECIFIC COMPETENCES	LEARNING ACTIVITIES	EXPECTED STANDARDS
			• Using APIs for cloud-based services to integrate with other systems • Integrate cloud services with Internet of Things (IoT) devices for real time data processing. • Designing and implementing cloud migration strategies	

SUMMARY OF KEY COMPETENCES FOR FORM 4

Competence	Descriptor
Computer System	Demonstrate understanding of different types of computer systems, including expert systems, satellite navigation systems, Configuring multi-user environments, connecting computers to a domain or workgroup for centralized management, installing virtual machines to run multiple operating systems on a single computer
Productivity tools: Databases	Demonstrate the ability to create, manage, and secure a relational database by organizing data into tables, records, and fields, performing basic database operations such as adding, modifying, and deleting records, and ensuring data integrity and security through proper querying, filtering, and access control measures.
Computer Networking	Demonstrate the ability to explain and apply networking concepts such as OSI layers and TCP/IP models, VLANs for network segmentation, VPNs for secure remote access, QoS for performance optimization, and network security protocols like HTTPS and SSL/TLS, while also performing basic troubleshooting to diagnose network issues.
Programming	Demonstrate proficiency in Writing simple programs using basic syntax and structures in C and C++, creating programs that uses loops, developing programs that uses arrays and strings for data storage and manipulations, defining Classes and objects in C and C++ to encapsulate data and behavior, applying techniques to identify and fix errors
Web Design	Demonstrate the ability to Configure a domain name to point to the web hosting server using DNS settings, setting up a web hosting account using control panel tools like cPannel or file manager, configuring hosting services, m anaging domain name services
Databases	Demonstrate proficiency in identifying scenarios where databases are used in industries, designing a database schema to manage catalogs, orders, and customer data for an online store, developing a database in banking and financial system, develop databases in healthcare, education, transport industries…
Mobile Application Development	Demonstrate the ability to differentiate between User Interface (UI) and User Experience (UX) in mobile applications, design wireframes and prototypes applying UI/UX principles, implement data storage techniques such as local and cloud storage, understand networking including API integration, build apps using multi-platform frameworks like Flutter or React Native, evaluate multi-platform development advantages and challenges, and effectively debug mobile applications while documenting the process.
Emerging Technologies	Demonstrate the ability to identify, explain, and apply emerging technologies such as AR, VR, and 3D printing across various industries, understanding their impact on society and the economy.

This comprehensive syllabus and summary ensure that learners develop a robust set of competences across the four years, preparing them for further studies and careers in computer science and related fields.

APPENDICES

SCOPE AND SEQUENCE OF COMPUTER SCIENCE SYLLABUS (FORM 1 TO FORM 4)

Topic	Form 1	Form 2	Form 3	Form 4
Fundamentals in Computing	Computer Basics. Operating system and file management, Application Software, Basic trouble shooting			
Computer Systems				Fundamentals of Computer Systems Functions and Benefits of Different Types of Computer Systems.
Productivity Tools	Word Processing. Functions of a word processing application. Text in a document. Insert and manipulate objects in a document. Text alignment, paragraph spacing, and indentation Proofreading tools.	Spreadsheets. Basic Operations in Spreadsheets. Formulas and Functions. Data Organization and Analysis. Charts and Graphs. Spreadsheet Security and Sharing.	Presentations. Creating and Designing Slides. Using Multimedia Elements in Presentations. Presentation Delivery. Techniques Collaborating on Presentations. Microsoft Publisher. Creating a New Publication. Working with Text Boxes. Inserting and Formatting Images. Creating and Formatting Shapes. Working with Templates and Page Design. Create and Edit Tables. Working with Master Pages.	

Topic	Form 1	Form 2	Form 3	Form 4
			Preparing a Publication for Print or Digital Distribution.	
Data Representation	Fundamentals of data representation Convention of number systems (Decimal, Binary). Analyzing binary coded to understand its representation in real-world application Demonstrating the use of binary numbers in programming and hardware configurations	Convention of number systems. (Octal, Hexadecimal) Implement Number System Operations in Programing Languages Representing text using ASCII		
Computer Networks	Network Design and Implementation	Network Topologies. Implement and Configure Network Topologies, Create network cables such as Ethernet, UTP, STP,FTP	The Internet Create and Share Content online	Network Design and Configurations. Introduction to Virtual Local Area Networks (VLANs). Introduction to VPNs. Quality of Service (QoS). Introduction to Network Security Protocols
Cyber Security	Fundamentals of Cyber Security	Protect Data, Devices and Networks from Cyber Threats. Identifying common cyber vulnerabilities in operating system, software and hardware Ethical Hacking and Penetration Testing.	Prevent and Respond to Cyber Threats. Analyse the impact of online actions on personal and professional reputation. Applying encryption techniques to secure sensitive data during storage and transmission	

Topic	Form 1	Form 2	Form 3	Form 4
			such as symmetric and asymmetric encryption.	
Data processing	Data collection and cleaning			
Algorithms		Fundamental of Algorithms. Understanding Flowcharts and Pseudocode. Applying Algorithms to Solve Problems. Refining Algorithms.		
Programming		Introduction to programing Block based programing Such as Scratch	Introduction to Python Programming Concepts. Environment Setup. Coding in Python. Build Practical Python Applications	Programming in C++. Describing the structures of C++ program such as headers, functions, main(). Writing simple programs using basic syntax and structures in C++
Web Design	Fundamentals of Web Design	Introduction to CSS. Applying CSS to websites. Using different CSS elements.	Introduction to JavaScript and Its Role in Web Development. JavaScript Syntax and Basic Operations. Working with Variables, Data Types, and Operators. Control Structures: Loops and Conditionals. Basic DOM Manipulation.	
Digital Citizenship	Digital Ethics			

Topic	Form 1	Form 2	Form 3	Form 4
	Strategies to Prevent Unethical Practices. Privacy and Intellectual Property Online.			
Database	Introduction to Databases. Key concepts of Databases. Database Terminologies.	Database Design and Creation using SQL. Writing SQL scripts to create and modify database structures Creating databases using SQL	Database Management Implement and Maintain Databases Upgrade and migrate databases to newer versions or platforms. Integrate database with programming languages such as python, Java, PHP	
Artificial Intelligence (AI)	fundamentals of Artificial Intelligence (AI)	Design and Develop AI Models	Natural Language Processing (NLP) and Computer Vision. AI in Robotics and Healthcare. Introduction to AI Programming. Deep Learning and AI in Big Data. Reinforcement Learning and AI in Autonomous Vehicles. Ethical Challenges in AI.	
Internet Of Things (IoT)	Basic Components of Internet of Things (IoT)	Discussing different development platforms Exploring the user interfaces Creating codes for IoTs	Introduction to Internet of Things (IoT). IoT Communication Protocols. IoT in Real World.	

Topic	Form 1	Form 2	Form 3	Form 4
Mobile Applications			Introduction to Mobile Applications. Create mobile applications	Mobile Application Design for deployment Design and Build mobile applications
Cloud Computing			Fundamentals of Cloud computing, Develop and Manage cloud computing models	Cloud Deployment and Operations. Designing cloud-native application using micro services architecture. Building APIs for cloud-based services to integrate with other systems. Cloud Providers and Setting Up Cloud Accounts.
Logic Gates	Definition and Purpose of Logic Gates. Types of Logic Gates: AND, OR, NOT, NAND, NOR. Truth Tables and Logical Expressions for Basic Gates. Combining Logic Gates to Create Circuits. Practical Applications of Logic Gates.			
Robotics		Fundamentals of Robotics. Discussing the basic concepts of Robotics including ethical use of robots,	Mechanical and Electronic Systems. Implement control systems such as PID controllers to manage robot actions.	
Emerging Technologies				Introduction to Emerging Technologies.

Hardware	Recommendations
Laptop, Smartphone, Printer, scanner, Tablets, Headphones, Network tool kit, microcontrollers and Microphones (Processor)	• Minimum: 4 cores (Core i3 and above), 1.6 Gigahertz (GHz) or faster, 64-bit. • Recommended: (8 cores or more), 2 GHz or higher, 64-bit.
RAM and Speed	• Recommended: 8 GB and above. • Recommended: 2GB or higher for Smart phones • Speed 1.6HGhz or Higher
Hard disk space/ SSD	• Minimum: 120 GB and above. • Minimum 16 GB for Smart phones

Topic	Form 1	Form 2	Form 3	Form 4
				Augmented Reality (AR) and Virtual Reality (VR). Introduction to 3D Printing Identify and Apply Emerging Technologies into industries