SENIOR SCHOOL CURRICULUM DESIGN

GRADE 10

COMPUTER SCIENCE

KENYA INSTITUTE OF CURRICULUM DEVELOPMENT
2024

KENYA INSTITUTE OF CURRICULUM DEVELOPMENT

Nurturing Every Learner's Potential

SENIOR SCHOOL CURRICULUM DESIGN

GRADE 10

COMPUTER SCIENCE

JUNE, 2024

First Published in 2024

ISBN: 978-9914-52-909-8

Published and printed by Kenya Institute of Curriculum Development

TABLE OF CONTENTS

NATIONAL GOALS OF EDUCATION

Education in Kenya should:

1. **Foster nationalism and patriotism and promote national unity.**
   Kenya's people belong to different communities, races and religions, but these differences need not divide them. They must be able to live and interact as Kenyans. It is a paramount duty of education to help young people acquire this sense of nationhood by removing conflicts and promoting positive attitudes of mutual respect which enable them to live together in harmony and foster patriotism in order to make a positive contribution to the life of the nation.

2. **Promote the social, economic, technological and industrial needs for national development.**
   Education should prepare the youth of the country to play an effective and productive role in the life of the nation.
   a) **Social Needs**
      Education in Kenya must prepare children for changes in attitudes and relationships which are necessary for the smooth progress of a rapidly developing modern economy. There is bound to be a silent social revolution following in the wake of rapid modernization. Education should assist our youth to adapt to this change.
   b) **Economic Needs**
      Education in Kenya should produce citizens with the skills, knowledge, expertise and personal qualities that are required to support a growing economy. Kenya is building up a modern and independent economy which is in need of an adequate and relevant domestic workforce.
   c) **Technological and Industrial Needs**
      Education in Kenya should provide learners with the necessary skills and attitudes for industrial development. Kenya recognizes the rapid industrial and technological changes taking place, especially in the developed world. We can only be part of this development if our education system is deliberately focused on the knowledge, skills and attitudes that will prepare our young people for these changing global trends.

3. **Promote individual development and self-fulfilment**
   Education should provide opportunities for the fullest development of individual talents and personality. It should help children to develop their potential interests and abilities. A vital aspect of individual development is the building of character.

4. **Promote sound moral and religious values.**
   Education should provide for the development of knowledge, skills and attitudes that will enhance the acquisition of sound moral values and help children to grow up into self-disciplined, self-reliant and integrated citizens.

5. **Promote social equity and responsibility.**
   Education should promote social equality and foster a sense of social responsibility within an education system which provides equal educational opportunities for all. It should give all children varied and challenging opportunities for collective activities and corporate social service irrespective of gender, ability or geographical environment.

6. **Promote respect for and development of Kenya's rich and varied cultures.**
   Education should instill in the youth of Kenya an understanding of past and present cultures and their valid place in contemporary society. Children should be able to blend the best of traditional values with the changing requirements that must follow rapid development in order to build a stable and modern society.

7. **Promote international consciousness and foster positive attitudes towards other nations.**
   Kenya is part of the international community. It is part of the complicated and interdependent network of peoples and nations. Education should therefore lead the youth of the country to accept membership of this international community with all the obligations and responsibilities, rights and benefits that this membership entails.

8. **Promote positive attitudes towards good health and environmental protection.**
   Education should inculcate in young people the value of good health in order for them to avoid indulging in activities that will lead to physical or mental ill health. It should foster positive attitudes towards environmental development and conservation. It should lead the youth of Kenya to appreciate the need for a healthy environment.

LEARNING OUTCOMES FOR SENIOR SCHOOL

By the end of senior school, the learner should be able to:

1. Communicate effectively and utilise information and communication technology across varied contexts.
2. Apply mathematical, logical and critical thinking skills for problem solving.
3. Apply basic research and scientific skills to manipulate the environment and solve problems.
4. Exploit individual talents for leisure, self-fulfilment, career growth, further education and training.
5. Uphold national, moral and religious values and apply them in day to day life.
6. Apply and promote health care strategies in day to day life.
7. Protect, preserve and improve the environment for sustainability.
8. Demonstrate active local and global citizenship for harmonious co-existence.
9. Demonstrate appreciation of diversity in people and cultures.
10. Manage pertinent and contemporary issues responsibly.

THE SENIOR SCHOOL IN THE COMPETENCY BASED CURRICULUM (CBC)

Senior School is the forth level of Basic Education in the Competency Based Curriculum (CBC) that learners shall come to after the Pre-Primary, Primary and Junior School (JS). The essence of Senior School is to offer learners a Pre- University/ Pre-career experience where the learners have an opportunity to choose pathways where they have demonstrated interest and/or potential at the earlier levels. Senior school comprises three years of education for learners in the age bracket of **15 to 18 years** and lays the foundation for further education and training at the tertiary level and the world of work. In the CBC vision, learners exiting this level are expected to be ***engaged, empowered and ethical citizens*** ready to participate in the socio-economic development of the nation.

At this level, learners shall take **SEVEN (07) learning areas (LAs)** as recommended by the *Presidential Working Party on Educational Reforms* (PWPER). These shall comprise **Four Compulsory** learning areas, and Three learning areas opted for by the learner according to their choses Pathway. While English and Kiswahili are indicated as Compulsory, the learners who opt for these learning areas as their subjects of specialization shall go through a *differentiated curriculum* in terms of scope, experiences and assessment. Such learners shall; therefore, take *Advanced English* or *Kiswahili Kipevu* with additional two lessons. It is recommended that AT LEAST TWO learning areas should be from chosen Pathway. In exceptional cases, some learners may opt for ONE learning area from the chosen Pathway and a maximum of TWO learning areas from any of the three pathways; depending on the learner's career projections and with guidance by the principals at Senior School.

PROPOSED LIST OF SUBJECTS AT SENIOR SCHOOL

Compulsory Subjects	**Science, Technology, Engineering & Mathematics (STEM)**	**Social Sciences**	**Arts & Sports Science**
1. English 2. Kiswahili/KSL 3. Community Service Learning 4. Physical Education ***NB: ICT skills will be offered to all students to facilitate learning and enjoyment***	5. Mathematics/Advanced Mathematics 6. Biology 7. Chemistry 8. Physics 9. General Science 10. Agriculture 11. Computer Studies 12. Home Science 13. Drawing and Design 14. Aviation Technology 15. Building and Construction 16. Electrical Technology 17. Metal Technology 18. Power Mechanics 19. Wood Technology 20. Media Technology* **21.** Marine and Fisheries Technology*	22. Advanced English 23. Literature in English 24. Indigenous Language 25. Kiswahili Kipevu/Kenya Sign Language 26. Fasihi ya Kiswahili 27. Sign Language 28. Arabic 29. French 30. German 31. Mandarin Chinese 32. History and Citizenship 33. Geography 34. Christian Religious Education/ Islamic Religious Education/Hindu Religious Education 35. Business Studies	36. Sports and Recreation 37. *Physical Education (C)* 38. Music and Dance 39. Theatre and Film 40. Fine Arts

LESSON DISTRIBUTION AT SENIOR SCHOOL

The number of lessons in each of the compulsory learning areas shall be 4; while the optional areas shall be 6 lessons each. A lesson shall be 40 minutes. The "free" lessons shall be used for development of ICT skills, Pastoral Instruction Programme (PPI), projects, collaborative study and further reading.

ESSENCE STATEMENT

The Computer Science curriculum at Senior School aims to equip a learner with competencies, skills and attitudes in computing principles, concepts and techniques. The competencies acquired will enable a learner to design, implement and manage computing systems that provide social, economic, industrial and technological solutions for national development. The curriculum design encompasses theoretical and practical aspects of foundations of computer science, computer networking and software development. These aspects will equip a learner with problem-solving, critical thinking and analytical skills through learner centred, experiential and inquiry-based learning approaches. The curriculum design is intended for a learner who would like to pursue a professional career in Computer Science or related disciplines.

This learning area is anchored on National Goals of Education No. 2 which aims to provide the learners with the necessary skills and attitudes for industrial development, Kenya Vision 2030 on making education responsive to education needs, Sessional Paper No 1 of 2019, which recommended the promotion of technical and vocational education with an emphasis on Science, Technology, and Innovation (ST&I) in the school curriculum. It is also informed by the second pillar of Africa Agenda 2063 and the National ICT Policy of Kenya 2016 (revised 2020), which emphasises on use of ICT as a foundation for the creation of a more robust economy.

GENERAL SUBJECT LEARNING OUTCOMES

By the end of Senior Secondary School, the learner should be able to:

a) Use knowledge, skills and values acquired in computer science to perform daily life activities.
b) Practise ethical and safe use of computer technology in society.

c) Develop computer applications for solving real world problems.
d) Configure and manage computer networks for effective resource sharing.
e) Initiate career and entrepreneurship opportunities in the field of computer science.
f) Adapt to the dynamic digital world to cope with the changing needs of the society.

SUMMARY OF STRANDS AND SUB STRANDS

Strands	Sub Strands	Suggested Number of Lessons
1.0 Foundation of Computer Science (56 Lessons)	1.1 Evolution of Computers	6
	1.2 Computer Architecture	4
	1.3 Input/Output (I/O) Devices	12
	1.4 Computer storage	6
	1.5 Central Processing Unit (CPU)	6
	1.6 Operating System (OS)	12
	1.7 Computer setup	10
2.0 Computer Networking (32 Lessons)	2.1 Data communication	6
	2.2 Data Transmission Media	8
	2.3 Computer Network Elements	10
	2.4 Network Topologies	8
3.0: Software Development (92 Lessons)	3.1 Computer Programming Concepts	14
	3.2 Program development	15
	3.3 Identifiers and Operators	18
	3.4 Control Structures	17
	3.5 Containers	14
	3.6 Functions	14
Total Number of Lessons		**180**

Note: The suggested number of lessons per sub strand may be more or less depending on the context.

STRAND 1.0: FOUNDATION OF COMPUTER SCIENCE

Strand	Sub Strand	Specific Learning Outcomes	Suggested Learning Experiences	Key Inquiry Questions
1.0 Foundation of Computer Science	**1.1 Evolution and Development of Computers** (6 lessons)	By the end of the sub strand the learner should be able to: a) Identify early computing devices and how they relate to evolution of electronic computers b) describe principle technologies that defined development of computers, c) Relate the principle technologies to respective computer generation, d) appreciate technological advancement in the development of computers.	The learner is guided to: ● use available resources to search for information on the ancient and medieval computing devices that necessitated the development of electronic computers (*Abacus, Napier's Bone, Pascaline, Slide Rule, Difference Engine, Analytic Engine, Jacquard loom),* ● use available resources to search for information on the principle technologies that defined development of computers *(vacuum tubes, transistors, integrated circuits (IC), large scale IC, very large scale IC)* and present findings, ● discuss various generations of computer (*first, second,*	1. What is the importance of principles and technologies in the development of computers? 2. How have computers evolved over generation?

			third, fourth, and fifth generation), ● match different generations of computers to their respective principal technologies, ● share ideas on the technological advancement in the development of computers.	

Core competencies to be developed:

- Communication and collaboration: learner listens keenly and actively participates in discussion of various generations of computer
- Self-efficacy: learner clearly follows the correct procedure to match different generations of computers to their respective principal technologies

Values:

- Unity: learner shares available resources amicably as they search for information on the principle technologies that defined development of computers.
- Responsibility: learner engages in assigned roles and duties as they present findings on principle technologies that defined development of computers

Pertinent and Contemporary Issues (PCIs):

Citizenship Education: equity and non-discrimination as a learner participate in group discussion regardless of their backgrounds.

Strand	Sub Strand	Specific Learning Outcomes	Suggested Learning Experiences	Key Inquiry Questions
1.0 Foundation of Computer Science	**1.2 Computer Organisation and Architecture** (4 lessons)	By the end of the sub strand the learner should be able to: a) describe the functional organisation and architecture of a von Neumann computer, b) analyse the relationships among functional elements of von Neumann computer, c) create a model of a computer architecture depicting the structural elements of a von Neumann computer, d) use binary, octal and hexadecimal number systems to represent data in a von Neumann's digital computer,	The learner is guided to: ● use available resources to search for information on the general structure of von Neumann computer architecture (*input, processing, output, storage, buses, instruction sets*) ● brainstorm on the differences between Reduced Instruction Set Computer (RISC) and Complex Instruction Set Computer (CISC) architectures. ● illustrate organisation among functional elements of computer architecture (input, output, control signals, interfaces and ports, CPU, storage) ● discuss how the elements of computer architecture communicate,	What is the role of computer architecture on performance of computers?

		e) appreciate the importance of computer architecture in computing.	• watch a video of fetch-execute cycle, • make a model computer architecture with all the structural elements, use locally available materials to, • convert numbers from base ten to binary, octal and hexadecimal systems used to represent data in a modern digital computer base to another • discuss the importance of the fetch-execute cycle in computer architecture.	

Core competencies to be developed:

- Critical Thinking and Problem solving: learner interprets and illustrates the interrelationship among elements of computer architecture.
- Digital literacy: learner uses digital devices to watch a video of fetch-execute cycle.

Values:

- Respect: learner understands and appreciates others' opinions as they discuss how the elements of computer architecture communicate.
- Unity: learner collaborates with others as they share available resources amicably to search for information on structural elements of von neumann computer architecture.

Pertinent and Contemporary Issues (PCIs):

Life Skills: Effective Communication as a learner discusses e elements of computer architecture.

Strand	Sub Strand	Specific Learning Outcomes	Suggested Learning Experiences	Key Inquiry Questions
1.0 Foundation of Computer Science	**1.3 Input/Output (I/O) Devices** (12 lessons)	By the end of the sub strand the learner should be able to: a) describe types of input and output devices used in computer systems, b) examine criteria used in selecting input and output devices, c) use input and output devices to perform tasks, d) appreciate the advancement of input and output devices used in computer systems.	The learner is guided to: ● discuss types of input devices (*keying devices, pointing devices, scanning devices, voice input devices, touch screen, digitizer, digital cameras, quick response (QR) code, digital camera, touch screens (resistive, capacitive and infra-red), two-dimensional (2D) and three-dimensional (3D) scanners),* ● use available resources to search for information on output devices (*printers, monitors, speakers, projectors, plotters, actuator),* ● discuss methods used by input devices to enter data into a computer. ● discuss methods used to output information from a computer,	1. How do input and output devices operate? 2. What are the technological advancements in input/output devices?

			• brainstorm on factors to consider when acquiring input and output devices, • create a quick response (QR) code and share information, • discuss advantages and disadvantages of QR codes and traditional barcodes, • scan a document and share with peers, • visit a computer user environment while observing safety practices, to identify the input/output devices, • match input and output devices to their uses, • share experience on how to reuse input/output devices which are still in good working condition to minimise wastage.	

Core competencies to be developed:

- Creativity and Imagination: learner generates new ideas to create a quick response (QR) code and share information.
- Learning to Learn: learner shares experience on how to reuse input and output devices.

Values:

- Peace: learner respects self and others while discussing advantages and disadvantages of QR codes and traditional barcodes.
- Unity: the leaner respects other opinions while brainstorming factors to be considered when acquiring input and output devices.

Pertinent and Contemporary Issues (PCIs):

Social Economic and Environmental Issues: learner observes safety when visiting a computer user environment to identify the input/output devices.

Strand	Sub Strand	Specific Learning Outcomes	Suggested Learning Experiences	Key Inquiry Questions
1.0 Foundation of Computer Science	**1.4 Computer storage** (6 lessons)	By the end of the sub strand the learner should be able to: a) identify types of storage used in computer systems, b) categorise types of storage used in computer systems, c) read data from a computer storage, d) write data to a computer storage, e) establish criteria used to select computer storage, f) acknowledge safety of data in computer storage media.	The learner is guided to: ● share experiences on types of storage devices they have interacted with, ● discuss the categories of computer storage *(primary and secondary)* ● discuss types of primary memory (*RAM and ROM*) ● discuss types of RAM technology (*dynamic RAM (DRAM) and static RAM (SRAM)*), ● differentiate between DRAM and SRAM, ● discuss features of Read-only memory (ROM) chips technology, ● discuss the differences between RAM and ROM chips, ● consult a resource person to discuss and demonstrate how ROM and RAM chips are	How is the safety of data in remote computer storage guaranteed?

			used in other devices (*microwave oven, refrigerator, remote-controlled toy car and aeroplane*), ● discuss and classify secondary storage into Internal secondary storage ((*Hard disk drive (HDD), Solid state drive (SSD)*) and External secondary storage *((DVD/CD and DVD-RAM, Blu-ray disc, USB memory stick/ flash memory, Removable hard drive)*), ● discuss advantages and disadvantages of different types of secondary storage, ● brainstorm the meaning of remote storage ● discuss examples of remote storage ● save data on a computer storage, ● transfer data using a computer storage,	

			• retrieve data from a computer storage, • store data in a remote storage, • retrieve data from remote storage, • carry out a case study to compare types of computer storage in terms of capacities, portability, robustness and data security, • debate on the benefits of remote storage versus local computer storage, • practice safety precautions when handling computer storage, • sensitises community members on how to reuse computer storage e-waste.	
Core competencies to be developed: • Communication and collaboration: learner speak engagingly as they share experiences on types of storage devices they have interacted with. • Self-efficacy: learner communicate effectively as they debate on the benefits of remote storage versus local computer storage.				
Values:				

- Responsibility: learner engages in assigned roles and duties as they carry out a case study to compare types of computer storage.
- Patriotism: learner serves the community when sensitising community members reuse computer storage e-waste.

Pertinent and Contemporary Issues (PCIs):
Socio- Economic and Environmental Issues: Safety in the computer user environment as a learner practises safety precautions when handling computer storage.

Strand	Sub Strand	Specific Learning Outcomes	Suggested Learning Experiences	Key Inquiry Questions
1.0 Foundation of Computer Science	**1.5 Central Processing Unit (CPU)** (6 lessons)	By the end of the sub strand the learner should be able to: a) describe structural elements of the CPU of a computer system, b) relate structural elements of the CPU to their functions, c) examine types of CPUs in computing devices, d) appreciate the role of CPU in computing.	The learner is guided to: ● brainstorm on the structural elements of the CPU, ● watch a video simulation of the structural elements of the CPU, ● discuss the functions of the structural elements of the CPU (*arithmetic and logic unit, control unit, registers, buses*), ● draw a diagram to show the fetch-decode-execute cycle and the interaction between CPU, registers and memory, ● compare types of CPU in computing devices according to instruction set (RISC vs CISC); word length (16-bit, 32-bit, 64-bit); core design (*single-core, dual-core, quad-core, hexa-core, octa-core, deca-core)*,	1. What are the technological advancements in the development of CPUs? 2. How does the CPU relate to the human brain?

			● discuss factors to be considered in selecting a CPU.	

Core competencies to be developed:

- Critical Thinking and Problem solving: learner demonstrates the fetch-decode-execute cycle interaction with the CPU.
- Creativity and Imagination: learner gains new knowledge as they discuss the functions of the structural elements of the CPU.

Values:

- Unity: learner appreciates efforts of others when drawing a diagram to show the fetch-decode-execute cycle and the interaction between CPU, registers and memory.
- Respect: learner understands and appreciates other opinions as they brainstorm on the structural elements of the CPU.

Pertinent and Contemporary Issues (PCIs):

Life Skills: learner develops effective communication as they discuss factors to be considered in selecting a CPU

Strand	Sub Strand	Specific Learning Outcomes	Suggested Learning Experiences	Key Inquiry Questions
1.0 Foundation of Computer Science	**1.6 Operating System (OS)** (12 lessons)	By the end of the sub strand the learner should be able to: a) describe functions of an operating system, b) classify operating system according to different attributes, c) install an operating system in a computer, d) use an operating system to perform a task, e) acknowledge the importance of operating systems in computing.	The learner is guided to: ● use available resources to search for the definition of the term operating system and give examples *(Microsoft Windows, MacOS, Linux, Android, iOS)*, ● use available resources to search for different functions of operating system and present findings, ● discuss functions of an operating system (*booting, resource management, user interface or command interpreter, memory management, input / output (I/O) management, file management, process management, user management*), ● visit computer user environment to observe and identify available operating system (*Windows operating system, Macintosh Operating System (MacOS), Linux, iOS, Android, Chrome OS),*	How does an operating system communicate with computer hardware?

			• discuss and classify operating system according to tasks (*single-tasking, multitasking*), users (*single-user, multi-user*), and user interface (*graphical user interface (GUI), command line, menu driven)*, • discuss factors to consider when choosing operating system, • select appropriate operating system for a given situation (*tasks*, *users*, *interface*), • write the procedure to be followed to install operating system in a computer, • install different types of OS and application software in computing devices in a virtual environment *(Virtual Box, Virtual Machine)*, • use operating systems to manipulate files and folders (create, rename, delete, restore, copy, move, backup) on computers, • watch a video on how the OS handles error and interrupts, manages devices, allocates tasks, • use of OS in error handling, managing input/output and storage devices,	

			● share experiences on installation and use of Operating Systems.	

Core competencies to be developed:

- Critical Thinking and Problem Solving: learner analyses situations and selects an appropriate operating system for a given situation.
- Learning to learn: learner reflects on own learning as they share experiences on installation and use of Operating Systems.

Values:

- Respect: learner is open minded as they discuss functions of an operating system.
- Integrity: learner applies laid down procedures as they use OS in error handling and managing input, output and storage devices.

Pertinent and Contemporary Issues (PCIs):

Life Skills: self-esteem as a learner install different types of OS and application software in computing devices in a virtual environment.

Strand	Sub Strand	Specific Learning Outcomes	Suggested Learning Experiences	Key Inquiry Questions
1.0 Foundation of Computer Science	**1.7 Computer setup** (10 Lessons)	By the end of the sub strand the learner should be able to: a) explain types of ports and cables used in computers, b) relate cables to their corresponding ports in a computer, c) set up a computer for use, d) appreciate following the correct procedure when setting up a computer.	The learner is guided to: ● search for information on different cables and ports used in computer systems, ● discuss the types of cables and ports used in computer systems, ● take turns to match ports to their corresponding cables, ● assemble tools needed in setting up a computer, ● discuss safety precautions in setting up a computer, ● connect cables to their corresponding ports in computers, ● connect all parts of a computer and use it to perform a task, ● share experiences on setting up computers, ● reuse computer e-waste that is still in good working condition to set up computers.	1. How are computers interfaced with the external environment? 2. Why is it important to practise safety when setting up computers?

Core competencies to be developed: ● Digital literacy: learner identifies and connects parts of a computer to perform a task. ● Self efficacy: learner follow the correct procedure to connect parts of a computer and use it to perform a task.
Values: ● Integrity: learner applies laid down procedure when connecting all parts of a computer and use it to perform a task. ● Responsibility: learner observes safety precautions as they connect cables to their corresponding ports in computers.
Pertinent and Contemporary Issues (PCIs): Socio-Economic and Environmental Issues: waste management as learner reuses computer e-waste that is still in good working condition to set up computers.

Suggested Assessment Rubric

Level / Indicator	Exceeds expectation	Meets expectation	Approaches expectations	Below expectations
Ability to relate the principle technologies to respective computer generation	Relates 5 principle technologies to respective computer generation and gives relevant examples	Relates 4 principle technologies to respective computer generation	Relates 3 principle technologies to respective computer generation	Relate at most 2 principle technologies to respective computer generation
Ability to use input and output devices to perform tasks	Uses input and output devices to perform tasks proficiently	Uses input and output devices to perform tasks	Uses some input and output devices to perform tasks	Uses some input and output devices to perform tasks, with limited challenge

Level / Indicator	Exceeds expectation	Meets expectation	Approaches expectations	Below expectations
Ability to read and write data to a computer storage device	Reads and writes data to a computer storage device using different methods	Reads and writes data to a computer storage device	Reads and writes data to a computer storage device, with prompts	Reads and writes data to a computer storage device, with prompts and assistance
Ability to use an operating system to perform a task	Uses an operating system to perform complex computing tasks	Uses an operating system to perform a tasks	Uses an operating system to perform basic tasks, with prompts	Uses an operating system to perform simple tasks, with prompts and assistance
Ability to set up a computer for use	Sets up a computer systematically and skillfully for use	Sets up a computer for use	Sets up a computer for use, with prompts	Sets up a computer for use, with prompts and assistance

STRAND2.0: COMPUTER NETWORKING

Strand	Sub Strand	Specific Learning Outcomes	Suggested Learning Experiences	Key Inquiry Questions
2.0 Computer Networking	**2.1 Data communication** (6 lessons)	By the end of the sub strand the learner should be able to: a) define basic data communications concepts b) describe characteristics of data communication in computer networking, c) analyse the components of data communication system, d) simulate modes of data flow in communication systems, e) acknowledge the significance of data communication systems in networking.	The learner is guided to: ● use available resources to search for definition of basic concepts used in data communication (*data, signals, communication channel, network, transmission media, TCP/IP, Open System Interconnection (OSI) model, protocols*) ● discuss characteristics of data communication, ● create a simulation to show interaction among data communication components (*sender, message, medium, protocol, receiver*), ● brainstorm the meaning of the terms (*data transmission, Analog signals, digital signal data rate, baud rate, band width*),	How have data communication systems evolved?

			● illustrate data communication modes in different situations *(simplex, half-duplex, full-duplex),* ● discuss advantages and disadvantages of data communication modes.	
Core competencies to be developed: ● Self Efficacy: learner follows the correct procedure in creating a simulation to show interaction among data communication components. ● Communication and Collaboration: learner contributes to group decision making by recognising and acknowledging other ideas as the discuss characteristics of data communication.				
Values: ● Patriotism: learner respects fellow learners as they discuss characteristics of data communication. ● Respect: learner appreciate diverse opinions as they discuss advantages and disadvantages of data communication modes.				
Pertinent and Contemporary Issues (PCIs): ● Life Skills: effective communication as a learner discusses advantages and disadvantages of data communication modes.				

Strand	Sub Strand	Specific Learning Outcomes	Suggested Learning Experiences	Key Inquiry Questions
2.0 Computer Networking	**2.2 Data Transmission Media** (8 lessons)	By the end of the sub strand the learner should be able to: a) define basic concepts used in data transmission b) describe types of transmission media used in computer networks, c) connect digital devices used in data communication, d) establish factors that affect communication over a computer network, e) appreciate the role of transmission media in computer networking.	The learner is guided to: ● use available resources to search for definition of basic data communication concepts (*transmission, transmission media, encoding, decoding, modulation, demodulation, multiplexing, demultiplexing*) ● brainstorm on the transmission media in computer networks, ● discuss the types of transmission media used in computer networks, ● draw illustrations of data transmission signals, multiplexing and demultiplexing. ● connect digital devices for communication using transmission media, while observing safety rules, ● share resources through connected digital devices through transmission media,	How is data transmitted in a transmission media?

			● discuss factors that lead to transmission impairment, ● brainstorm on ways of overcoming transmission impairments in the transmission medium, ● share experiences on use of transmission media in sharing resources.	

Core competencies to be developed:

- Communication and Collaboration: learner speak engagingly as they share experiences on use of transmission media in sharing resources.
- Digital Literacy: learner connects digital devices for communication using transmission media.

Values:

- Unity: learner collaborates with as they share experiences on use of transmission media in sharing resources.
- Peace: learner respect self and others as they discuss factors that lead to transmission impairment.

Pertinent and Contemporary Issues (PCIs):

- Life Skills: effective communication s learner discuss the types of transmission media used in computer networks.

Strand	Sub Strand	Specific Learning Outcomes	Suggested Learning Experiences	Key Inquiry Questions
2.0 Computer Networking	**2.3 Computer Network Elements** (10 lessons)	By the end of the sub strand the learner should be able to: a) identify different types of computer networks b) describe elements of a computer network, c) evaluate criteria of a computer network, d) connect a computing device to available network, e) appreciate the role of computer networks in communication.	The learner is guided to: ● use available resources to search for different types of computer networks *(LAN, MAN, WAN, PAN, WLAN)* and present to peers, ● visit a computer user environment, identify the type of network in use and list components that make up the network *(transmission media, DTE, DCE, network software),* ● discuss the functions of each network device *(routers, NICs, gateways, hubs, MODEMS, bridges, repeaters)* and network software (*network OS, web browsers, search engines, email clients…*) ● discuss standards (protocols) that establish communication over a network *(TCP/IP, SMTP, HTTPS…),*	How are computer networks formed?

			• share data through different computer network setups, • discuss qualities a computer network should have *(performance, security, scalability, reliability, availability),* • discuss methods of connecting to a computer network *(point to point connection, multi-point connection)* • connect a digital device to an available computer network (*wired, wireless)* and share data, • observe online safety rules when sharing resources through a computer network, • use locally available materials to model a computer network, • share experiences on the use of computer networks.	

Core competencies to be developed: ● Citizenship: learner observe online ethical and moral practices as they share resources through a computer network ● Digital Literacy: learner connects a digital device to an available computer network and shares data.
Values: ● Respect: learner share resources equitably as they use locally available materials to model a computer network. ● Social justice: learner obeys laws and regulations as they share resources through a computer network
Pertinent and Contemporary Issues (PCIs): ● Citizenship Education: social cohesion as learners share experiences on the use of computer networks regardless of their backgrounds.

Strand	Sub Strand	Specific Learning Outcomes	Suggested Learning Experiences	Key Inquiry Questions
2.0 Computer Networking	**2.4 Network Topologies** (8 lessons)	By the end of the sub strand the learner should be able to: a) differentiate between physical and logical network topologies b) describe types of logical and physical topologies in computer networking, c) create a physical network topology for a computer network, d) appreciate the use of network topologies in computer networks.	The learner is guided to: ● search for information on the definition of network topology, ● identify the differences between physical and logical computer network topologies, ● draw illustrations of different types of physical topologies *(star, ring, bus, mesh, tree, hybrid),* ● simulate a physical network topology using available devices, ● brainstorm on the advantages and disadvantages of physical network topologies, ● select an appropriate physical network topology for a given situation, ● share experiences on use of different physical network topologies.	1. How does a network topology affect network performance? 2. What is the importance of topologies in networking?

Core competencies to be developed:

- Critical Thinking and Problem Solving: learner brainstorms and discusses advantages and disadvantages of physical network topologies.
- Creativity and Imagination: learner generate new ideas on how to simulate a physical network topology.

Values:

- Unity: learner collaborates with others as they discuss types of computer network physical topologies.
- Love: learner respects others opinion as they discuss advantages and disadvantages of physical network topologies.

Pertinent and Contemporary Issues (PCIs):

Life skills: Effective communication as learner share experiences on use of different physical network topologies.

Suggested Assessment Rubric

Level / Indicator	Exceeds expectation	Meets expectation	Approaches expectations	Below expectations
Ability to simulate modes of data flow in communication	Simulates the three modes of data flow in communication proficiently	Simulates the three modes of data flow in communication	Simulates at most two modes of data flow in communication	Simulates at most one mode of data flow in communication
Ability to connect digital devices used in data communication	Connects digital devices used in data communication systematically and proficiently	Connects digital devices used in data communication	Connects digital devices used in data communication with prompts	Connects digital devices used in data communication with prompts and assistance
Ability to connect a computer device to available network	Connects a computer device to available network using different methods	Connects a computer device to available network	Connects computer device to available network with prompts	Connects computer device to available network with prompts and assistance
Ability to create a physical network topology for a computer network	Creates a physical network topology for a computer network systematically and proficiently	Creates a physical network topology for a computer network	Creates a physical network topology for a computer network with prompts	Creates a physical network topology for a computer network with prompts and assistance

STRAND 3.0: SOFTWARE DEVELOPMENT

Strand	Sub Strand	Specific Learning Outcomes	Suggested Learning Experiences	Suggested Key Inquiry Question(s)
3.0 Software development	**3.1 Computer Programming Concepts** (14 lessons)	By the end of the sub strand the learner should be able to: a) explain the terminologies used in programming languages, b) describe evolution of programming languages in software development, c) categorise the programming languages according to the paradigms, d) create simple instructions to simulate low level programming languages, e) acknowledge the evolution of programming languages in software development.	The learner is guided to: • discuss the terminologies used in programming languages (*programming, programming language, assembler, compiler, interpreter, syntax, source code, machine code, Integrated Development Environment, software development lifecycle),* • use available resources to search for information on evolution of programming languages, and present findings, • discuss levels of programming languages in computing *(low level and high level),* • use available resources to search for information about programming paradigms (*structured, procedural, object oriented, functional, logic, event driven*) and share findings,	1. Why are programming languages important in computing? 2. How are programming languages executed?

			• prepare charts showing evolution of programming languages timelines and display in the learning environment, using locally available materials, • write simple instructions to show how the machine language and assembly languages operate, • share experiences on the use of different levels of programming languages in software development.	

Core competencies to be developed:

- Communication and Collaboration: learner contributes to group by discussing levels of programming languages in computing.
- Learning to Learn: learner shares experiences on the use of programming languages in computing.

Values:

- Unity: learner collaborates with others as they discuss levels of programming languages in computing.
- Respect: learner appreciates others opinion as they discuss levels of programming languages in computing.

Pertinent and Contemporary Issues (PCIs):

Life Skills: self-esteem as learner searches for information on evolution of programming languages and present findings.

Strand	Sub Strand	Specific Learning Outcomes	Suggested Learning Experiences	Suggested Key Inquiry Question(s)
3.0 Software development	**3.2 Program development** (15 lessons)	By the end of the sub strand the learner should be able to: a) describe stages of program development in computer programming, b) write a pseudocode to illustrate the logical flow of an algorithm, c) represent the logical flow of an algorithm using a flowchart, d) design an algorithm to solve a real life problem, e) appreciate the importance of using algorithms in problem solving	The learner is guided to: ● discuss the stages of program development cycle (*problem definition, program design, coding, testing, implementation, documentation, maintenance),* ● discuss characteristics of an algorithm *(input, output, finite, definite, effective),* ● discuss the keywords (*start, variables, input, processing, output, end*) used in pseudocodes and demonstrate how they are used to represent algorithms, ● use the keywords (*start, variables, input, processing, output, end*) to write a pseudocode that illustrates the logical flow of an algorithm, ● discuss the standard symbols (*start, input, processing, decision, repetition, continuation, output, end*) used in flowcharts and demonstrate how they are used to represent an algorithm,	1. What is the essence of the program development cycle? 2. Why are algorithms designed?

			• draw a flowchart to illustrate the logical flow of an algorithm, • use pseudocodes and flowcharts to solve a real life problem, • share experiences on the significance of algorithms in problem solving.	
Core competencies to be developed: ● Critical Thinking and Problem Solving: learner seeks alternatives that may solve a problem when discussing characteristics of an algorithm. ● Creativity and Imagination: learner undertakes tasks as they work on a problem to find a solution when constructing an algorithm using a flowchart and pseudocode.				
Values: ● Unity: learner display team spirit as they draw and discuss symbols of a program flowchart. ● Respect: learner understands and appreciates others as they share experiences on the significance of algorithms in problem solving.				
Pertinent and Contemporary Issues (PCIs): Socio-Economic and Environmental Issues: safety in the school as learner observe safety practices in the computer user environment as they use digital devices when constructing algorithms.				

Strand	Sub Strand	Specific Learning Outcomes	Suggested Learning Experiences	Suggested Key Inquiry Question(s)
3.0 Software development	**3.3 Identifiers and Operators** (18 lessons)	By the end of the sub strand the learner should be able to: a) describe the elementary elements of a computer program, b) declare variables and constants in a programming language, c) use input and output statements in a programming language, d) use operators in a programming language, e) appreciate the role of identifiers and operators in programming.	The learner is guided to: ● discuss the structural elements of a programming language *(structure, syntax, errors)*, ● brainstorm on reserved words, identifiers and data types used in programming, ● write computer programs using a high-level language (*Python, C. C++, C#, Java, Ruby, Scratch, Swift, Go*), ● watch a video clip on declaration of variables and constants in a programming language, ● use variables and constants in a program, ● assign data types to variables in a program, ● use standard libraries in programming language, ● write and execute programs that accept input and display output.	1. What is the importance of structured programming language? 2. How are variables declared in programming languages?

			● use available resources to search for programming operators (*arithmetic, assignment, increment and decrement, relational, logical*), ● discuss categories of operators used in programming (*arithmetic, assignment, increment and decrement, relational, logical*), ● watch a video demonstrating the order of precedence in the evaluation of operators, ● write and execute programs that involve use of operators, ● share experiences on the role of identifiers and operators in programming.	

Core competencies to be developed:

- Creativity and Imagination: learner writes a program that accepts user inputs and displays outputs.
- Learning to Learn: learner watches a video on order of precedence of the operators used in programming, then write and execute programs that involve use of operators.

Values:

- Unity: learner collaborates with others as they brainstorm on reserved words, identifiers and data types used in programming.
- Respect: learner appreciate others' effort as they discuss the structure of a programming language.

Pertinent and Contemporary Issues (PCIs):

Citizenship Education: social cohesion as learner share experiences on writing programs.

Strand	Sub Strand	Specific Learning Outcomes	Suggested Learning Experiences	Suggested Key Inquiry Question(s)
3.0 Software development	**3.4 Control Structures** (17 lessons)	By the end of the sub strand the learner should be able to: a) describe control structures in programming, b) select program control structure for a given situation c) use control structures in programming, d) appreciate the application of control structures in programming.	The learner is guided to: ● use available resources to search for information on program control structures and their importance in programming *(sequential, iteration, selection)*, ● watch a video that shows how to use decision statements in solving problems, ● write and execute programs involving use of decision statements to perform tasks, ● write and execute programs with break and continue statements in control structures, ● share experiences on the application of control structures in programming.	1. How are Boolean expressions used in programming? 2. What is the importance of control structures in programming?
Core competencies to be developed: ● Critical Thinking and Problem solving: learner writes and executes programs involving use of decision statements to perform tasks. ● Self-Efficacy: learner successfully writes and executes programs with break and continue statements in control structures.				
Values: ● Respect: learner is open minded as they share experiences on use of control structures in programming.				

• Social justice: learner shares resources equitably as they use available resources to search for information on boolean and compound expressions.
Pertinent and Contemporary Issues (PCIs): Life Skills: self-esteem as learners present findings on boolean and compound expressions.

Strand	Sub Strand	Specific Learning Outcomes	Suggested Learning Experiences	Suggested Key Inquiry Question(s)
3.0 Software development	**3.5 Data Structures** (14 lessons)	By the end of the sub strand the learner should be able to: a) describe types of containers in programming, b) use containers in programming, c) apply sorting and searching techniques in data structures, d) embrace the use of containers in programming.	The learner is guided to: ● brainstorm the meaning and use of data structures *(lists, arrays, dictionaries, sets, tuples, classes, abstract data types),* ● watch a video clip on declaration and initialisation of single and two dimensional arrays or lists ● discuss the syntax of data structures in programming, ● write examples of lists, dictionary, sets and tuples as used in programming, ● write and execute a program that involves the use of data structures to solve a problem, ● demonstrate how to find the location of an element in a list using sequential search and binary search ● demonstrate how to sort elements in a list using insertion sort, selection sort, bubble sort and quick sort. ● share experiences on the use of containers in programming.	Why are containers used in programming?

Core competencies to be developed:

- Digital Literacy: learner uses a digital device to watch a video clip on declaration and initialisation of single and two dimensional containers.
- Learning to Learn: learner reflect on their own learning as they share experiences on the use of containers to solve problems in programming.

Values:

- Peace: learner respect self and others as they brainstorm the meaning and use of containers.
- Respect: learner appreciates diverse opinions as they discuss the syntax of containers in programming.

Pertinent and Contemporary Issues (PCIs):

Citizenship Education: national values as learner fosters fairness and justice among peers as they discuss and write examples of lists, dictionaries, sets and tuples in turns.

Strand	Sub Strand	Specific Learning Outcomes	Suggested Learning Experiences	Suggested Key Inquiry Question(s)
3.0 Software development	**3.6 Functions** (14 lessons)	By the end of the sub strand the learner should be able to: a) discuss types of function used in modular programming, b) use built-in and user-defined functions to create a modular program, c) discuss the scope of variables and parameter passing between functions d) appreciate the importance of modularity in programming.	The learner is guided to: ● search for the meaning and importance of modular programming, ● discuss the differences between user defined and inbuilt functions, ● discuss the general syntax of a function (*return type, function name, parameter list, function body*), ● user available resource to search for information on how functions communicate through parameter passing *(formal parameters, actual parameters, return types, functions prototype/signature),* ● discuss and present the scope of variables in a modular program *(local, global),* ● watch a video that demonstrates how to use built-in and user defined functions	What are the roles of functions in programming?

			● write and execute programs involving use of parameters in built-in functions, ● write and execute programs that involve use of parameters in user-defined functions ● share experiences on the importance of modular programming.	
Core competencies to be developed: ● Self-Efficacy: learner writes and executes programs that involve use of user defined functions. ● Digital Literacy: learner watches a video on how to design a user defined functions.				
Values: ● Unity: learner collaborates with others as they write and execute programs involving use of inbuilt functions. ● Peace: learner respects self and others as they share experiences on the use of function in solving problems in programming.				
Pertinent and Contemporary Issues (PCIs): ● Life Skills: effective communication as a learner discusses comparisons between functions and writing separate modules in programming.				

Suggested Assessment Rubric

Level / Indicator	Exceeds expectations	Meets expectations	Approaches expectations	Below expectations
Ability to categorise the programming languages according to the paradigms	Categorises programming languages systematically according to paradigms and gives detailed description and examples of each paradigm	Categorises programming languages according to paradigms.	Categorises some programming languages according to paradigms.	Categorises some programming languages according to paradigms with assistance.
Ability to design an algorithm to solve a real-life problem	Creatively and proficiently designs an algorithm to solve a real-life problem and provides documentation	Designs an algorithm to solve real-life problem	Designs an algorithm to solve real-life problem with prompts	Designs an algorithm to solve real-life problem with prompts and assistance
Ability to declare variables and constants in a programming language	Declares variables and constants proficiently in a programming language and includes appropriate comments	Declares variables and constants in a programming language	Declares variables and constants in a programming language with prompts	Declares variables and constants in a programming language with prompts and assistance

Level / Indicator	Exceeds expectations	Meets expectations	Approaches expectations	Below expectations
Ability to use input and output statements in programming language	Uses input and output statements in programming language proficiently and adds extra features	Uses input and output statements in programming language	Uses input and output statements in programming language with prompts	Uses input and output statements in programming language with prompts and assistance
Ability to use operators in a programming language	Uses operators in a programming language skillfully	Uses operators in a programming language	Uses operators in a programming language with prompts	Uses operators in a programming language with prompts and assistance
Ability to use control structures in a programming language	Uses control structures in a programming language skillfully in different situations	Uses control structure in a programming language	Uses control structure in a programming language with prompts	Uses control structure in a programming language with prompts and assistance
Ability to use containers in programming	Uses containers in a programming language skillfully in different situations	Uses containers in programming	Uses containers in programming with prompts	Uses containers in programming with prompts and assistance
Ability to use functions in programming	Uses functions in programming skillfully in different situations	Uses functions in programming	Uses functions in programming with prompts	Uses functions in programming with prompts and assistance

APPENDIX: LIST OF ASSESSMENT METHODS, LEARNING RESOURCES AND NON-FORMAL ACTIVITIES

Strand	Sub Strand	Suggested Assessment Methods	Suggested Learning Resources	Suggested Non-Formal Activities
1.0 Foundation of Computer Science	**1.1 Evolution of Computers**	Observation schedules, checklists, rating scales, rubrics, questionnaires, projects, journals, portfolios, oral questions, aural questions, interview schedules, learner's profile, written tests, anecdotal records.	• Reference materials (learners' books, teachers' guide, Newspapers, journals, magazines) • Audio resources (radio talk, audio tapes, DVDs/CDs) • Visual resources (photos, charts, posters, graphs, manilla papers, flash cards) • ICT resources Television, internet, QR codes, computer hardware, productivity tools, visual programming tools, computer software (OS, Utility software and Application programs digital materials and equipment) • Community resources (field trips, computer science exhibition/fair, computer science laboratory, computer science club) • Human Resources (teachers, computer laboratory technician)	Educate community members on the technological advancement in the development of computers

Strand	Sub Strand	Suggested Assessment Methods	Suggested Learning Resources	Suggested Non-Formal Activities
	1.6 Operating System (OS)	Observation schedules, checklists, rating scales, rubrics, questionnaires, projects, journals, portfolios, oral questions, aural questions, interview schedules, learner's profile, written tests, anecdotal records.	• Reference materials (learners' books, teachers' guide, Newspapers, journals, magazines) • Audio resources (radio talk, audio tapes, DVDs/CDs) • Visual resources (photos, charts, posters, graphs, manilla papers, flash cards) • ICT resources Television, internet, computer hardware, productivity tools, visual programming tools, computer software (OS, Utility software and Application programs digital materials and equipment) • Community resources (field trips, computer science exhibition/fair, computer science laboratory, computer science club) • Human Resources (teachers, computer laboratory technician)	Educate community members how to install a software

Strand	Sub Strand	Suggested Assessment Methods	Suggested Learning Resources	Suggested Non-Formal Activities
	1.7 Computer setup	rubrics, questionnaires, projects, journals, portfolios, oral questions, aural questions, interview schedules, learner's profile, written tests, anecdotal records, observation schedules, checklists, rating scales,.	• Reference materials (learners' books, teachers' guide, Newspapers, journals, magazines) • Audio resources (radio talk, audio tapes, DVDs/CDs) • Visual resources (photos, charts, posters, graphs, manilla papers, flash cards) • ICT resources Television, internet, computer hardware, productivity tools, visual programming tools, computer software (OS, Utility software and Application programs digital materials and equipment) • Community resources (field trips, computer science exhibition/fair, computer science laboratory, computer science club) • Human Resources (teachers, computer laboratory technician)	Demonstrate to how Computer setup during social gatherings

2.0 Computer Networking	2.1 Data communication	Observation schedules, checklists, rating scales, rubrics, questionnaires, projects, journals, portfolios, oral questions, aural questions, interview schedules, learner's profile, written tests, anecdotal records.	• Reference materials (learners' books, teachers' guide, Newspapers, journals, magazines) • Audio resources (radio talk, audio tapes, DVDs/CDs) • Visual resources (photos, charts, posters, graphs, manilla papers, flash cards) • ICT resources Television, internet, computer hardware, productivity tools, visual programming tools, computer software (OS, Utility software and Application programs digital materials and equipment) • Community resources (field trips, computer science exhibition/fair, computer science laboratory, computer science club) • Human Resources (teachers, computer laboratory technician)	Debate on advantages and disadvantages of data communication modes

	2.2 Data Transmission Medium	Observation schedules, checklists, rating scales, rubrics, questionnaires, projects, journals, portfolios, oral questions, aural questions, interview schedules, learner's profile, written tests, anecdotal records.	• Reference materials (learners' books, teachers' guide, Newspapers, journals, magazines) • Audio resources (radio talk, audio tapes, DVDs/CDs) • Visual resources (photos, charts, posters, graphs, manilla papers, flash cards) • ICT resources Television, internet, computer hardware, productivity tools, visual programming tools, computer software (OS, Utility software and Application programs digital materials and equipment) • Community resources (field trips, computer science exhibition/fair, computer science laboratory, computer science club) • Human Resources (teachers, computer laboratory technician)	Share ideas with peers on how to connect computer devices using data transmission media

	2.3 Computer Network elements	Observation schedules, checklists, rating scales, rubrics, questionnaires, projects, journals, portfolios, oral questions, aural questions, interview schedules, learner's profile, written tests, anecdotal records.	• Reference materials (learners' books, teachers' guide, Newspapers, journals, magazines) • Audio resources (radio talk, audio tapes, DVDs/CDs) • Visual resources (photos, charts, posters, graphs, manilla papers, flash cards) • ICT resources Television, internet, computer hardware, productivity tools, visual programming tools, computer software (OS, Utility software and Application programs digital materials and equipment) • Community resources (field trips, computer science exhibition/fair, computer science laboratory, computer science club) • Human Resources (teachers, computer laboratory technician)	Demonstrate to community members how to connect devices to a computer network

	2.4 Network Topologies	Observation schedules, checklists, rating scales, rubrics, questionnaires, projects, journals, portfolios, oral questions, aural questions, interview schedules, learner's profile, written tests, anecdotal records.	• Reference materials (learners' books, teachers' guide, Newspapers, journals, magazines) • Audio resources (radio talk, audio tapes, DVDs/CDs) • Visual resources (photos, charts, posters, graphs, manilla papers, flash cards) • ICT resources Television, internet, computer hardware, productivity tools, visual programming tools, computer software (OS, Utility software and Application programs digital materials and equipment) • Community resources (field trips, computer science	During a social gathering, simulate a physical network topology using available devices

3.0 Software development	3.1 Computer Programming concepts	Observation schedules, checklists, rating scales, rubrics, questionnaires, projects, journals, portfolios, oral questions, aural questions, interview schedules, learner's profile, written tests, anecdotal records.	• Reference materials (learners' books, teachers' guide, Newspapers, journals, magazines) • Audio resources (radio talk, audio tapes, DVDs/CDs) • Visual resources (photos, charts, posters, graphs, manilla papers, flash cards) • ICT resources Television, internet, computer hardware, productivity tools, visual programming tools, computer software (OS, Utility software and Application programs digital materials and equipment) • Community resources (field trips, computer science exhibition/fair, computer science laboratory, computer science club) • Human Resources (teachers, computer laboratory technician)	Educate community members the use of programming languages

	3.2 Program development	Observation schedules, checklists, rating scales, rubrics, questionnaires, projects, journals, portfolios, oral questions, aural questions, interview schedules, learner's profile, written tests, anecdotal records.	• Reference materials (learners' books, teachers' guide, Newspapers, journals, magazines) • Audio resources (radio talk, audio tapes, DVDs/CDs) • Visual resources (photos, charts, posters, graphs, manilla papers, flash cards) • ICT resources Television, internet, computer hardware, productivity tools, visual programming tools, computer software (OS, Utility software and Application programs digital materials and equipment) • Community resources (field trips, computer science exhibition/fair, computer science laboratory, computer science club) • Human Resources (teachers, computer laboratory technician)	During clubs, share experiences on the significance of algorithms in problem solving

	3.3 Identifiers and Operators	Observation schedules, checklists, rating scales, rubrics, questionnaires, projects, journals, portfolios, oral questions, aural questions, interview schedules, learner's profile, written tests, anecdotal records.	• Reference materials (learners' books, teachers' guide, Newspapers, journals, magazines) • Audio resources (radio talk, audio tapes, DVDs/CDs) • Visual resources (photos, charts, posters, graphs, manilla papers, flash cards) • ICT resources Television, internet, computer hardware, productivity tools, visual programming tools, computer software (OS, Utility software and Application programs digital materials and equipment) • Community resources (field trips, computer science exhibition/fair, computer science laboratory, computer science club) • Human Resources (teachers, computer laboratory technician)	Demonstrate to club and society members how to use variables and constants in a program

	3.4 Control Structures	Observation schedules, checklists, rating scales, rubrics, questionnaires, projects, journals, portfolios, oral questions, aural questions, interview schedules, learner's profile, written tests, anecdotal records.	• Reference materials (learners' books, teachers' guide, Newspapers, journals, magazines) • Audio resources (radio talk, audio tapes, DVDs/CDs) • Visual resources (photos, charts, posters, graphs, manilla papers, flash cards) • ICT resources Television, internet, computer hardware, productivity tools, visual programming tools, computer software (OS, Utility software and Application programs digital materials and equipment) • Community resources (field trips, computer science exhibition/fair, computer science laboratory, computer science club) • Human Resources (teachers, computer laboratory technician)	Educate peers on the use of control structures in programming Share ideas on the importance of control structures in programming

	3.5 Containers	Observation schedules, checklists, rating scales, rubrics, questionnaires, projects, journals, portfolios, oral questions, aural questions, interview schedules, learner's profile, written tests, anecdotal records.	• Reference materials (learners' books, teachers' guide, Newspapers, journals, magazines) • Audio resources (radio talk, audio tapes, DVDs/CDs) • Visual resources (photos, charts, posters, graphs, manilla papers, flash cards) • ICT resources Television, internet, computer hardware, productivity tools, visual programming tools, computer software (OS, Utility software and Application programs digital materials and equipment) • Community resources (field trips, computer science exhibition/fair, computer science laboratory, computer science club) • Human Resources (teachers, computer laboratory technician)	Share with peers the experiences on the use of containers in problems solving

	3.6 Functions	Observation schedules, checklists, rating scales, rubrics, questionnaires, projects, journals, portfolios, oral questions, aural questions, interview schedules, learner's profile, written tests, anecdotal records.	• Reference materials (learners' books, teachers' guide, Newspapers, journals, magazines) • Audio resources (radio talk, audio tapes, DVDs/CDs) • Visual resources (photos, charts, posters, graphs, manilla papers, flash cards) • ICT resources Television, internet, computer hardware, productivity tools, visual programming tools, computer software (OS, Utility software and Application programs digital materials and equipment) • Community resources (field trips, computer science exhibition/fair, computer science laboratory, computer science club) • Human Resources (teachers, computer laboratory technician)	Share a video simulation of the components of function during computer club activities

KENYA INSTITUTE OF CURRICULUM DEVELOPMENT

Desai Road, off Murang'a Road.
P.O.Box 30231-00100 Nairobi, Kenya.
Telephone: +254(020)3749900-9,3748204,3747994
Fax:+254(020)3639130
Email: info@kicd.ac.ke, Website: www.kicd.ac.ke