Grade 11 Computer Studies

Term 1 - 64 complete lesson plans

Ready to teach, in the KICD lesson-plan format | 40-minute lessons | PDF and editable Word

ABOUT THIS PACK

64 complete, ready-to-teach Grade 11 Computer Studies lesson plans for Term 1, covering data representation, binary arithmetic operations, logic gates, logic circuits and computer network standards and protocols.

TOPICS IN THIS PACK

Topic	Name	Lessons
1.1	Data Representation	18
1.2	Binary Arithmetic Operations	12
1.3	Logic Gates	12
1.4	Logic Circuits	12
2.1	Computer Network Standards and Protocols	10

INSIDE EVERY LESSON PLAN

- The specific learning outcomes, key inquiry question, core competencies, values and PCIs, and the resources needed.
- A timed lesson flow: what the teacher does and what the learners do, from introduction to summary.
- The key content (a worked example, notes or a practical with its expected results), then four practice questions with the answers for the teacher.
- The common mistake to watch for, and ideas for learners who need support and for those ready for more.
- A four-level CBC assessment rubric (EE, ME, AE, BE), homework, and a blank reflection space to fill in after teaching.

A TASTE OF LESSON 1

Why computers use binary: the bit

By the end of the lesson, the learner should be able to: a) explain why computers represent data in binary; b) define a bit

> **Everything is stored as 0s and 1s**
> A computer is built from millions of tiny electronic switches (transistors). Each switch has two states, on and off, so a computer works naturally with two symbols: 1 (on) and 0 (off).
> A BIT (binary digit) is the smallest unit of data: one 0 or one 1.

You also receive this pack as an editable Word file, so you can change any lesson to suit your class before you print it. Lessons follow the order of the KICD curriculum design.

CONTENTS

No.	Lesson	Topic	Page
1	Why computers use binary: the bit	1.1 Data Representation	5
2	Units of data: bit, nibble, byte and word	1.1 Data Representation	8
3	How units affect file size, storage and speed	1.1 Data Representation	11
4	Representing text and images	1.1 Data Representation	14
5	Representing audio and video	1.1 Data Representation	17
6	Number systems: radix and positional value	1.1 Data Representation	20
7	Real-world applications of number systems	1.1 Data Representation	23
8	Converting decimal to binary	1.1 Data Representation	26
9	Converting binary to decimal	1.1 Data Representation	29
10	Converting between decimal and octal	1.1 Data Representation	32
11	Converting between decimal and hexadecimal	1.1 Data Representation	35
12	Converting between binary and octal	1.1 Data Representation	38
13	Converting between binary and hexadecimal	1.1 Data Representation	41
14	Converting between octal and hexadecimal	1.1 Data Representation	44
15	Writing a program to convert number systems	1.1 Data Representation	47
16	Coding schemes: ASCII	1.1 Data Representation	50
17	Coding schemes: EBCDIC and BCD	1.1 Data Representation	53
18	Group project: a text-based communication system	1.1 Data Representation	56
19	Rules of binary addition	1.2 Binary Arithmetic Operations	59
20	Adding unsigned binary numbers	1.2 Binary Arithmetic Operations	62
21	Direct binary subtraction	1.2 Binary Arithmetic Operations	65
22	Unsigned and signed numbers: signed magnitude	1.2 Binary Arithmetic Operations	68
23	One's complement representation	1.2 Binary Arithmetic Operations	71
24	Two's complement representation	1.2 Binary Arithmetic Operations	74
25	Comparing signed number methods	1.2 Binary Arithmetic Operations	77
26	Subtraction using one's complement	1.2 Binary Arithmetic Operations	80
27	Subtraction using two's complement	1.2 Binary Arithmetic Operations	83
28	Adding signed numbers and overflow	1.2 Binary Arithmetic Operations	86
29	Binary arithmetic in the computer	1.2 Binary Arithmetic Operations	89
30	Sensitising peers and review of binary arithmetic	1.2 Binary Arithmetic Operations	92
31	Meaning of logic gates	1.3 Logic Gates	95
32	The AND gate	1.3 Logic Gates	98

Elimu254 Learn • Grow • Build Tomorrow

33	The OR gate	1.3 Logic Gates	101
34	The NOT gate	1.3 Logic Gates	104
35	NAND and NOR gates	1.3 Logic Gates	107
36	The XOR gate	1.3 Logic Gates	110
37	Venn diagrams and real-life logic	1.3 Logic Gates	113
38	Matching and modelling gate symbols	1.3 Logic Gates	116
39	Truth tables for gate combinations	1.3 Logic Gates	119
40	Analysing truth tables to predict outputs	1.3 Logic Gates	122
41	Deriving boolean expressions from truth tables	1.3 Logic Gates	125
42	Simulating gates in circuits; review	1.3 Logic Gates	128
43	Logic gates as the building blocks of logic circuits	1.4 Logic Circuits	131
44	Combining two gates	1.4 Logic Circuits	134
45	Circuits with three inputs	1.4 Logic Circuits	137
46	Generating boolean expressions from circuits	1.4 Logic Circuits	140
47	Generating circuits from boolean expressions	1.4 Logic Circuits	143
48	Showing two circuits are equivalent	1.4 Logic Circuits	146
49	The half adder: logic circuits that add	1.4 Logic Circuits	149
50	Burglar alarms and simple switches	1.4 Logic Circuits	152
51	Street lights and emergency response systems	1.4 Logic Circuits	155
52	Designing a circuit from a real problem	1.4 Logic Circuits	158
53	Resource person: applications of logic circuits	1.4 Logic Circuits	161
54	Review of logic circuits	1.4 Logic Circuits	164
55	Meaning and purpose of network standards and protocols	2.1 Computer Network Standards and Protocols	167
56	Examples of network standards and their bodies	2.1 Computer Network Standards and Protocols	170
57	The OSI model	2.1 Computer Network Standards and Protocols	173
58	The TCP/IP model compared with OSI	2.1 Computer Network Standards and Protocols	176
59	Application layer protocols: HTTP, SMTP, FTP and DNS	2.1 Computer Network Standards and Protocols	179
60	Transport layer protocols: TCP and UDP	2.1 Computer Network Standards and Protocols	182
61	Internet and network access protocols: IPv4, IPv6 and MAC	2.1 Computer Network Standards and Protocols	185

Elimu254 Learn • Grow • Build Tomorrow

62	Encapsulation and decapsulation	2.1 Computer Network Standards and Protocols	188
63	Creating a LAN and assigning IP addresses in a simulator	2.1 Computer Network Standards and Protocols	191
64	Connecting two LANs with a router; troubleshooting and review	2.1 Computer Network Standards and Protocols	194

Elimu254
Learn • Grow • Build Tomorrow

Why computers use binary: the bit

Strand 1.0 Foundation of Computer Studies > Sub-strand 1.1 Data Representation | Lesson 1 of 18 | 40 minutes

SCHOOL	TEACHER	DATE
CLASS / STREAM	TIME	ROLL (B / G)

SPECIFIC LEARNING OUTCOME(S)

By the end of the lesson, the learner should be able to: a) explain why computers represent data in binary; b) define a bit

KEY INQUIRY QUESTION(S)

- Why are different data units important in computing?
- How do different number systems function in computing?

CORE COMPETENCIES, VALUES AND PCIS IN THIS LESSON

COMPETENCY	**Critical thinking and problem solving**	Learners choose a method, check their answers and give reasons for their choices.
COMPETENCY	**Learning to learn**	Learners say what they found easy or hard and decide their next step.
VALUE	**Unity**	Learners share materials fairly and work as one team.
VALUE	**Respect**	Learners listen while others explain and accept correction politely.
PCI	**Social awareness skills**	Learners communicate respectfully and work well in groups.

LEARNING RESOURCES

Charts of place values and conversion steps | Flash cards with binary digits | ASCII and EBCDIC code tables | Notebooks, pencils and calculators

ORGANISATION OF LEARNING

Pairs for conversions; groups of three or four at a computer, taking turns at the keyboard; whole class for checking step-by-step working on the board.

KEY WORDS

bit | nibble | byte | word | pixel | sampling | number system | radix | positional value | decimal | binary | octal | hexadecimal | coding scheme | ASCII | EBCDIC | BCD

THE LESSON AT A GLANCE

Introduction	Teacher models	Guided practice	On their own	Wrap-up
6 min	9 min	10 min	9 min	6 min

LESSON FLOW

Stage	Minutes	The teacher	The learners
Introduction	6	Greet the class and ask: "A photo, a song and a text message all sit on the same phone. What do they have in common inside the phone?" Take two or three answers, link them to earlier work, then state today's outcome.	Answer the starter question, recall what they already know and read the outcome aloud.
Step 1: Teacher models	9	Take learners through "Everything is stored as 0s and 1s" (key ideas on the next page), explaining as you go. Use charts of place values and conversion steps, flash cards with binary digits, aSCII and EBCDIC code tables. Ask learners to predict what comes next before you show it.	Watch, predict the next step, ask questions and note the key steps in their exercise books.
Step 2: Guided practice	10	Give questions 1 and 2. Move round the groups, ask "How do you know?" and correct errors kindly on the spot.	Work in pairs or small groups, explain each step to a partner and check one another's answers.
Step 3: Independent practice	9	Set questions 3 and 4 for individual work. Mark as you walk round and note who needs help in the next lesson.	Work on their own, then swap books with a partner to check.
Summary and conclusion	6	Ask two or three learners to state the key idea in their own words. Give this exit question and check the answers: "What does 1 stand for in a computer circuit?"	Say the key idea in their own words and answer the exit question in the exercise book.

KEY IDEAS

Everything is stored as 0s and 1s

A computer is built from millions of tiny electronic switches (transistors). Each switch has two states, on and off, so a computer works naturally with two symbols: 1 (on) and 0 (off).

A BIT (binary digit) is the smallest unit of data: one 0 or one 1.

Two states are reliable: it is easy for a circuit to tell "high voltage" from "low voltage", even if the voltage changes slightly. Ten different levels would be much easier to confuse.

Activity: use a row of bottle tops, facing up for 1 and down for 0, and show that 3 tops can make 8 different patterns: $2 \times 2 \times 2 = 8$.

PRACTICE QUESTIONS - WITH ANSWERS FOR THE TEACHER

No.	Question	Answer
1	What does the word bit stand for?	**Binary digit.**
2	Why do computers use two states?	**Their switches are either on or off, and two states are reliable to detect.**
3	How many patterns can 4 bits make?	**$2 \times 2 \times 2 \times 2 = 16$ patterns.**
4	What does 1 stand for in a computer circuit?	**The on state (high voltage).**

WATCH OUT FOR

Learners think 0 means "nothing is stored". A 0 is information, just as a 1 is.

ASSESSMENT RUBRIC - THE FOUR CBC LEVELS

Level	What the learner can do
Exceeds expectations (EE)	Can explain why computers represent data in binary confidently and accurately, explain the method used, and apply the idea to new or unfamiliar problems.
Meets expectations (ME)	Can explain why computers represent data in binary accurately in familiar situations; notices and corrects slips without help.
Approaches expectations (AE)	Can partly explain why computers represent data in binary, or does so with prompts from the teacher or a partner; makes some errors.
Below expectations (BE)	Cannot yet explain why computers represent data in binary without direct help; needs the idea taught again with concrete objects.

SUPPORT AND EXTENSION

Learners who need support	Let learners make patterns with bottle tops and count them.
Learners ready for more	How many different patterns can 8 bits make? (Answer: 2^8 is 256 patterns)
Inclusion	Seat learners with low vision near the board and read aloud whatever you write; use large, clear, tactile materials; allow extra time; pair with a patient partner; use gestures or Kenyan Sign Language for the key words.

HOMEWORK AND FOLLOW-UP

Redo questions 3 and 4 neatly in the exercise book. Then tell someone at home what you learned about "why computers use binary: the bit" and bring one question of your own to ask next lesson.

TEACHER SELF-CHECK

- Did every learner get a turn to handle materials or explain their thinking?
- Did most learners meet the outcome? Write down the names of those who did not.
- Was the pace right — did any step take too long or too short a time?
- Which competency and value did I see learners showing?
- What will I change the next time I teach this lesson?

REFLECTION ON THE LESSON / SELF-REMARKS

Write this after teaching - what worked, what to change, which learners need follow-up.

Units of data: bit, nibble, byte and word

Strand 1.0 Foundation of Computer Studies > Sub-strand 1.1 Data Representation | Lesson 2 of 18 | 40 minutes

SCHOOL	TEACHER	DATE
CLASS / STREAM	TIME	ROLL (B / G)

SPECIFIC LEARNING OUTCOME(S)

By the end of the lesson, the learner should be able to: a) identify the units of data measurement; b) convert between storage units

KEY INQUIRY QUESTION(S)

- Why are different data units important in computing?
- How do different number systems function in computing?

CORE COMPETENCIES, VALUES AND PCIS IN THIS LESSON

COMPETENCY	**Critical thinking and problem solving**	Learners choose a method, check their answers and give reasons for their choices.
COMPETENCY	**Learning to learn**	Learners say what they found easy or hard and decide their next step.
VALUE	**Unity**	Learners share materials fairly and work as one team.
VALUE	**Respect**	Learners listen while others explain and accept correction politely.
PCI	**Social awareness skills**	Learners communicate respectfully and work well in groups.

LEARNING RESOURCES

Charts of place values and conversion steps | Flash cards with binary digits | ASCII and EBCDIC code tables | Notebooks, pencils and calculators

ORGANISATION OF LEARNING

Pairs for conversions; groups of three or four at a computer, taking turns at the keyboard; whole class for checking step-by-step working on the board.

KEY WORDS

bit | nibble | byte | word | pixel | sampling | number system | radix | positional value | decimal | binary | octal | hexadecimal | coding scheme | ASCII | EBCDIC | BCD

THE LESSON AT A GLANCE

Introduction	Teacher models	Guided practice	On their own	Wrap-up
6 min	9 min	10 min	9 min	6 min