

**CHRISTIAN SOCIAL SERVICES COMMISSION (CSSC)
NORTHERN ZONE JOINT EXAMINATION SYNDICATES (NZ-JES)**



**FORM TWO PRE- NATIONAL EXAMINATION AUGUST-2026
COMPUTER SCIENCE
MARKING SCHEME**

SECTION A (15 Marks)

1: Multiple Choice (10 Marks)

(i)	(ii)	(iii)	(iv)	(v)	(vi)	(vii)	(viii)	(ix)	(x)
B	C	B	C	C	C	C	A	B	B

QUESTION 2: Matching (5 Marks)

LIST A	(i)	(ii)	(iii)	(iv)	(v)
LIST B	E	D	C	F	B

SECTION B (70 Marks)

QUESTION 3: Computer Generations and Types (10 Marks)

a) ENIAC Generation and Disadvantage (2 Marks)

Criteria	Maximum Marks	Student's Response
Correct generation name	1	First Generation
One valid disadvantage	1	Very large in size / Consumed a lot of electricity / Generated excessive heat / Very expensive

Sample Answer:

ENIAC belongs to the **First Generation** of computers. One disadvantage of computers from this generation is that they were **very large in size**, occupying entire rooms, and they **consumed huge amounts of electricity**, making them expensive to operate.

b) Differentiate General-Purpose and Special-Purpose Computers (4 Marks)

Criteria	Maximum Marks	Student's Response
Correct definition of general-purpose	1	Can perform many different tasks / Can run various software applications
Correct example of general-purpose	1	Desktop computer / Laptop / Smartphone
Correct definition of special-purpose	1	Designed for one specific task only

Criteria	Maximum Marks	Student's Response
Correct example of special-purpose	1	EFD machine / ATM / Calculator / Traffic light controller

Sample Answer:

General-purpose computers are designed to perform many different tasks and can run various types of software. For example, a **desktop computer** at school can be used for typing documents, browsing the internet, playing games, and creating spreadsheets.

Special-purpose computers are designed to perform only one specific task. For example, an **Electronic Fiscal Device (EFD)** used in shops is designed only to record sales and print receipts for tax purposes. It cannot be used for browsing the internet or typing letters.

Daily Life Examples:

- General-purpose: A **laptop** used by a student for both studies and entertainment
- Special-purpose: A **calculator** used only for mathematical calculations

c) Why a Smartphone is Considered a Computer (2 Marks)

Criteria	Maximum Marks	Student's Response
Valid reason	2	It has a processor (CPU), memory (RAM), storage, and runs an operating system / It can process data, store information, and execute instructions

Sample Answer:

A smartphone is considered a computer because it contains all the essential components of a computer: a **processor (CPU)** that executes instructions, **memory (RAM)** for temporary data storage, **permanent storage** for files and apps, and an **operating system** (like Android or iOS) that manages hardware and software. It can process data, run applications, connect to the internet, and perform calculations just like a desktop computer.

d) Two Improvements of Fifth Generation over First Generation (2 Marks)

Criteria	Maximum Marks	Student's Response
First valid improvement	1	Fifth Generation computers are much smaller / Use microprocessors / Are more powerful
Second valid improvement	1	Fifth Generation computers use Artificial Intelligence / Are more energy efficient / Have faster processing speeds

Sample Answer:

1. **Size and Speed:** Fifth Generation computers are **much smaller** and **thousands of times faster** than First Generation computers. While ENIAC filled a large room, modern computers fit in our pockets.
2. **Artificial Intelligence:** Fifth Generation computers can **learn and make decisions** using Artificial Intelligence (AI), whereas First Generation computers could only perform basic calculations using machine language.

QUESTION 4: Computer Maintenance and Safety (10 Marks)

a) How Dust Causes Overheating and Restarting (2 Marks)

Criteria	Maximum Marks	Student's Response
Explanation of dust blocking airflow	1	Dust blocks ventilation holes and fans, preventing proper airflow
Explanation of overheating effect	1	Overheating triggers thermal protection, causing automatic shutdown/restart

Sample Answer:

Dust accumulates on computer components, especially on **fans and heat sinks**. This dust blocks the **ventilation holes** and prevents fans from spinning properly, reducing airflow inside the system unit. As a result, heat generated by the CPU and other components cannot escape, causing the computer to **overheat**. When the temperature reaches a dangerous level, the computer's **thermal protection system** automatically restarts or shuts down to prevent permanent damage to the hardware.

b) Three Safety Precautions Before Opening a computer (3 Marks)

Criteria	Maximum Marks	Student's Response
First precaution	1	Turn off and unplug the computer from the power source
Second precaution	1	Wear an anti-static wrist strap or discharge static electricity by touching a metal object
Third precaution	1	Remove all jewelers (rings, watches, bracelets) / Work on a clean, dry surface / Use proper tools

Sample Answer:

1. **Power off and unplug** the computer from the electrical socket to prevent electric shock.
2. **Wear an anti-static wrist strap** or touch a grounded metal object to discharge static electricity, which can damage sensitive internal components.
3. **Remove all jewellery** such as rings, watches, and bracelets, and work on a **clean, dry surface** to prevent scratches, shorts, or electrical hazards.

c) Three Reasons for Regular Cleaning of Computer Hardware (3 Marks)

Criteria	Maximum Marks	Student's Response
First reason	1	Prevents overheating by ensuring proper airflow
Second reason	1	Extends the lifespan of computer components
Third reason	1	Prevents hardware failure and reduces repair costs / Maintains performance

Sample Answer:

1. **Prevents overheating:** Regular removal of dust ensures that fans and ventilation holes remain clear, allowing proper airflow and preventing components from overheating.
2. **Extends component lifespan:** Clean computers experience less stress from heat and dust, meaning parts like the power supply, CPU, and hard drive last longer.
3. **Reduces hardware failure and repair costs:** Regular maintenance catches problems early and prevents costly breakdowns, saving the school money on repairs and replacements.

d) Why a Vacuum Cleaner Should NOT Be Used Inside a Computer (2 Marks)

Criteria	Maximum Marks	Student's Response
Valid reason explained	2	Vacuum cleaners generate static electricity that can damage sensitive electronic components / They create strong suction that can dislodge or damage small components / They cannot reach tight spaces effectively

Sample Answer:

A vacuum cleaner should not be used inside a computer case because it generates **static electricity** as air and dust move through the plastic hose. This static electricity can discharge onto sensitive electronic components like the motherboard, RAM, and CPU, causing **permanent damage**. Additionally, the **strong suction** can dislodge small components or bend delicate pins, and it cannot effectively remove dust from tight spaces inside the case.

QUESTION 5: File Size Calculations (10 Marks)**a) Calculate Total Storage in Megabytes (MB) - (5 Marks)**

Criteria	Maximum Marks	Student's Response
Correct calculation of images	1	$15 \times 2 \text{ MB} = 30 \text{ MB}$
Correct calculation of videos	1	$3 \times 700 \text{ MB} = 2,100 \text{ MB}$
Correct calculation of software installer	1	$1.5 \text{ GB} = 1.5 \times 1,024 = 1,536 \text{ MB}$
Correct calculation of text documents	1	$50 \times 50 \text{ KB} = 2,500 \text{ KB} = 2,500 \div 1,024 = 2.44 \text{ MB}$ (or $50 \times 50 / 1024$)
Correct total addition	1	$30 + 2,100 + 1,536 + 2.44 = \mathbf{3,668.44 \text{ MB}}$

Sample Answer:

- Images: $15 \times 2 \text{ MB} = \mathbf{30 \text{ MB}}$
- Videos: $3 \times 700 \text{ MB} = \mathbf{2,100 \text{ MB}}$
- Software installer: $1.5 \text{ GB} \times 1,024 = \mathbf{1,536 \text{ MB}}$
- Text documents: $50 \times 50 \text{ KB} = 2,500 \text{ KB} \div 1,024 = \mathbf{2.44 \text{ MB}}$

Total = $30 + 2,100 + 1,536 + 2.44 = 3,668.44 \text{ MB}$

b) Convert Total to Gigabytes (GB) - (3 Marks)

Criteria	Maximum Marks	Student's Response
Correct conversion formula	1	Divide MB by 1,024
Correct calculation	1	$3,668.44 \div 1,024$
Correct answer	1	3.58 GB

Sample Answer:

Total in GB = $3,668.44 \text{ MB} \div 1,024 = \mathbf{3.58 \text{ GB}}$ (approximately)

c) Is 4 GB USB Flash Drive Enough? - (2 Marks)

Criteria	Maximum Marks	Student's Response
Correct decision	1	Yes, it is enough
Valid reason	1	4 GB > 3.58 GB, so there is enough space with about 0.42 GB remaining

Sample Answer:

Yes, the 4 GB USB flash drive is enough to store all the files. The total file size is **3.58 GB**, which is less than the flash drive's capacity of **4 GB**. There will be approximately **0.42 GB (about 430 MB)** of free space remaining.

QUESTION 6: Algorithm and Flowchart (10 Marks)

a) Algorithm to Calculate Total Until Zero (4 Marks)

Criteria	Maximum Marks	Student's Response
Step 1: Initialize total to zero	1	Set total = 0
Step 2: Input first number	1	Prompt user to enter a number
Step 3: Loop while number is not zero	1	While number $\neq$ 0, add to total and input next number
Step 4: Display total when loop ends	1	Print the final total

Sample Answer:

Step 1: Start

Step 2: Set total = 0

Step 3: Display "Enter a number (enter 0 to stop):"

Step 4: Read number

Step 5: WHILE number $\neq$ 0 DO

Step 6: total = total + number

Step 7: Display "Enter a number (enter 0 to stop):"

Step 8: Read number

Step 9: END WHILE

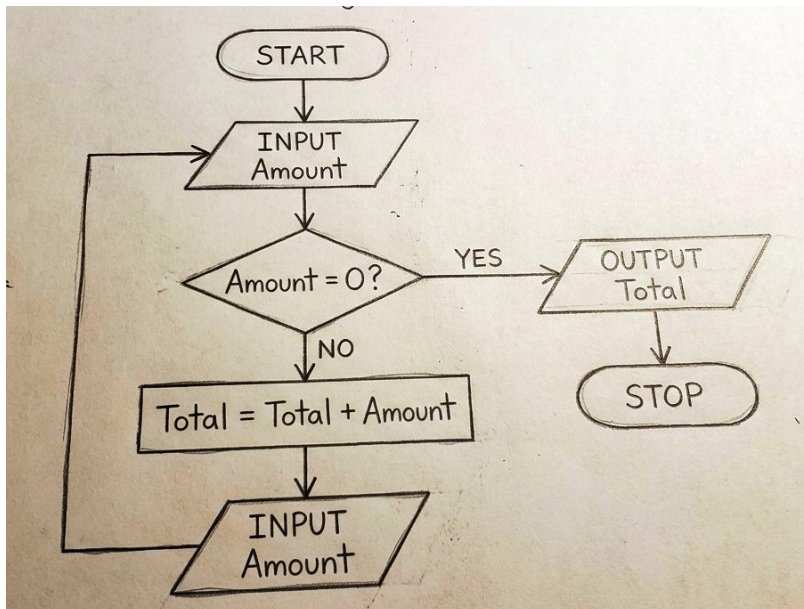
Step 10: Display "Total = " + total

Step 11: Stop

b) Flowchart (4 Marks)

Criteria	Maximum Marks	Student's Response
Correct use of Start/Stop symbol (oval)	1	Oval shape with "Start" and "Stop"
Correct use of Input/Output symbol (parallelogram)	1	Parallelogram for reading and displaying
Correct use of Process symbol (rectangle)	1	Rectangle for total = total + number
Correct use of Decision symbol (diamond)	1	Diamond for "Is number = 0?" with two arrows

Sample Flowchart:



c) Control Structures Used (2 Marks)

Criteria	Maximum Marks	Student's Response
First control structure	1	Iteration/Looping (WHILE loop)
Second control structure	1	Sequence (steps executed in order)

Sample Answer:

The algorithm uses the following control structures:

1. **Iteration (Looping):** The WHILE loop repeats the process of accepting numbers and adding them until the user enters 0.
2. **Sequence:** Instructions are executed in order from top to bottom.

QUESTION 7: Cyber Security (10 Marks)

a) Five Actions After Account Hacking (5 Marks)

Criteria	Maximum Marks	Student's Response
First action	1	Change password immediately
Second action	1	Inform friends/contacts not to send money
Third action	1	Report to the email service provider
Fourth action	1	Enable two-factor authentication
Fifth action	1	Check for suspicious activities / Report to authorities

Sample Answer:

1. **Change the password immediately** to a strong, unique password to lock the hacker out.
2. **Inform all friends and contacts** by phone or in person not to respond to any money requests from the hacked account.
3. **Contact the email service provider** (e.g., Gmail support) to report the hack and get help recovering the account.
4. **Enable two-factor authentication (2FA)** to add an extra layer of security.
5. **Check account settings** for any changes (recovery email, phone number, forwarding rules) and revert them, then **report the incident to the authorities** if money was stolen.

b) Government Authority for Cybercrime in Tanzania (1 Mark)

Criteria	Maximum Marks	Student's Response
Correct authority name	1	Tanzania Police Force - Cybercrime Unit / TCRA (Tanzania Communications Regulatory Authority) / PSE (Prevention of Cybercrime Unit)

Sample Answer:

Serious cybercrime cases in Tanzania can be reported to the **Tanzania Police Force - Cybercrime Unit** or the **Tanzania Communications Regulatory Authority (TCRA)**.

c) Two Measures to Prevent Account Hacking (2 Marks)

Criteria	Maximum Marks	Student's Response
First measure	1	Use strong passwords (combination of letters, numbers, symbols)
Second measure	1	Enable two-factor authentication / Avoid clicking suspicious links

Sample Answer:

- i. **Use a strong password** that includes a combination of uppercase and lowercase letters, numbers, and special characters (e.g., "S3cur3@2026"). Avoid using easy-to-guess information like birthdates or names.
- ii. **Enable two-factor authentication (2FA)** so that even if the password is stolen, the hacker cannot access the account without the second verification code sent to the user's phone.

d) Two-Factor Authentication (2FA) and How It Improves Security (2 Marks)

Criteria	Maximum Marks	Student's Response
Clear definition of 2FA	1	A security method that requires two different forms of verification before granting access
Explanation of how it improves security	1	Even if a password is stolen, the hacker cannot access the account without the second factor (e.g., phone code, fingerprint)

Sample Answer:

Two-factor authentication (2FA) is a security method that requires the user to provide **two different forms of identification** before gaining access to an account. Typically, this involves:

1. **Something you know** (password or PIN)
2. **Something you have** (a verification code sent to your phone, or a physical security key)

How it improves security: Even if a hacker steals your password, they cannot access your account because they do not have the second factor (e.g., the SMS code sent to your phone). This makes it much more difficult for hackers to break into accounts.

QUESTION 8: C Programming (10 Marks)

a) Declare Variables in C (3 Marks)

Criteria	Maximum Marks	Student's Response
Correct declaration for customer name	1	char name[50];
Correct declaration for deposit amount	1	float deposit;
Correct declaration for number of transactions	1	int transactions;

Sample Answer:

```
#include <stdio.h>
```

```
int main() {
    char customerName[50]; // Variable for customer name
    float depositAmount;   // Variable for deposit amount (with decimals)
    int numTransactions;    // Variable for number of transactions (whole number)

    return 0;
}
```

b) Complete C Program (7 Marks)

Criteria	Maximum Marks	Student's Response
Correct include statement (stdio.h)	1	#include <stdio.h>
Correct main function structure	1	int main() { ... return 0; }
Correct variable declarations	1	float deposit, total = 0; int i, transactions;
Correct input for transactions	1	printf("Enter number of transactions: "); scanf("%d", &transactions);
Correct loop for deposits	1	for(i=0; i<transactions; i++) { scanf("%f", &deposit); total += deposit; }
Correct output for total	1	printf("Total amount deposited: %.2f\n", total);
Correct use of data types (float for money)	1	Using float/double for monetary values

Sample Answer:

```
#include <stdio.h>
```

```
int main() {
    char customerName[50];
    float depositAmount, totalDeposit = 0;
    int numTransactions, i;

    // Get customer name
    printf("Enter customer name: ");
    scanf("%s", customerName);

    // Get number of transactions
    printf("Enter number of transactions for %s: ", customerName);
    scanf("%d", &numTransactions);

    // Loop to get each deposit
    for (i = 0; i < numTransactions; i++) {
        printf("Enter deposit amount for transaction %d: ", i + 1);
        scanf("%f", &depositAmount);
        totalDeposit = totalDeposit + depositAmount;
    }
}
```

```
// Display total
printf("\nCustomer: %s\n", customerName);
printf("Total amount deposited: %.2f Tsh\n", totalDeposit);

return 0;
}
```

QUESTION 9: Python Programming (10 Marks)

a) What is Python and Why is it Easy? (2 Marks)

Criteria	Maximum Marks	Student's Response
Definition of Python	1	A high-level, interpreted programming language
Reason for ease of learning	1	Uses simple English-like syntax / No need for complex punctuation like semicolons

Sample Answer:

Python is a high-level, interpreted programming language that is widely used for web development, data analysis, artificial intelligence, and education. It is considered easy for beginners because it uses **simple, English-like commands** such as print() and input(), and it does not require complex punctuation like semicolons or curly braces. The code is easy to read and write, making it ideal for new programmers.

b) Python Program to Display "Hello, World!" (2 Marks)

Criteria	Maximum Marks	Student's Response
Correct use of print() function	1	print("Hello, World!")
Correct string syntax (quotes)	1	Using single or double quotes correctly

Sample Answer:

```
print("Hello, World!")
```

c) Explain input() Function with Example (2 Marks)

Criteria	Maximum Marks	Student's Response
Explanation of input()	1	It reads data entered by the user from the keyboard and returns it as a string
Example of use	1	name = input("Enter your name: ")

Sample Answer:

The input() function in Python **reads data entered by the user** from the keyboard and returns it as a **string**. It can display a prompt message to tell the user what to enter.

Example:

```
name = input("Enter your name: ")
```

```
print("Hello, " + name + "!")
```

This program asks the user for their name and then greets them.

d) Python Program to Check Even or Odd (4 Marks)

Criteria	Maximum Marks	Student's Response
Correct input using input()	1	num = int(input("Enter a number: "))

Criteria	Maximum Marks	Student's Response
Correct conversion to integer	1	Using int() to convert input to a number
Correct if-else statement	1	if num % 2 == 0: ... else: ...
Correct output messages	1	print("Even") and print("Odd")

Sample Answer:

Program to check if a number is even or odd

Accept a number from the user

```
number = int(input("Enter a number: "))
```

Check if the number is even or odd

```
if number % 2 == 0:
```

```
    print("Even")
```

```
else:
```

```
    print("Odd")
```

Explanation:

- number % 2 == 0 checks if the number divided by 2 leaves a remainder of 0 (even)
- If true, it prints "Even"
- Otherwise, it prints "Odd"

SECTION C (15 Marks)

QUESTION 10: Essay on Programming Concepts (15 Marks)

Marking Rubric for Essay

Criteria	Maximum Marks	Student's Response
Content Knowledge (10 Marks)		
Definition of programming	1.5	Clear explanation of programming as writing instructions for computers
Definition of a program	1.5	Clear explanation of a program as a set of instructions
Difference between source code and machine code	2	Source code is human-readable; machine code is binary (0s and 1s)
Purpose of variables and data types in Python	2	Variables store data; data types define the kind of data (int, float, string, bool)
Use of if-else statements and loops	3	If-else for decisions; loops for repetition/processing many students (with examples)
Structure and Organization (3 Marks)		
Introduction	1	Clear introduction stating the purpose of the essay
Body paragraphs	1	Well-organized body covering all required concepts
Conclusion	1	Clear conclusion summarizing the main points
Language and Clarity (2 Marks)		
Correct grammar and spelling	1	No major errors

Criteria	Maximum Marks	Student's Response
Clarity of explanations	1	Easy to understand, student-friendly language

Model Essay Answer

INTRODUCTION

Programming is the backbone of modern computer systems, enabling machines to perform useful tasks such as storing student information, calculating averages, and displaying results. In today's digital world, understanding how programming works is essential for anyone using computers. This essay explains what programming and a program are, the difference between source code and machine code, and how variables, data types, and if-else statements are used in Python to process student data and make decisions.

MAIN BODY

Programming and a Program

Programming is the process of writing step-by-step instructions that tell a computer how to perform specific tasks. It is similar to writing a recipe that guides a cook through preparing a meal. A program, on the other hand, is the final collection of these instructions saved as a file that the computer can execute. For example, in a school computer system, programmers write instructions that tell the computer to accept student names and marks, calculate averages, and display the results on the screen.

Source Code vs. Machine Code

Programmers write instructions using human-readable languages like Python or C. This written code is called **source code**. However, computers cannot understand source code directly because they only work with electrical signals that represent binary numbers (0s and 1s). Therefore, source code must be translated into **machine code**—a set of binary instructions that the computer's processor can execute immediately. This translation is done by special programs called compilers or interpreters. For example, when a programmer writes `print("Hello")` in Python, the computer translates this into machine code that the processor understands and executes.

Variables and Data Types in Python

In Python programming, **variables** act like labeled storage containers that hold data temporarily while the program runs. For instance, a programmer can create a variable called `student_name` to store a student's name. **Data types** define the kind of data a variable can hold and what operations can be performed on it. The most common data types in Python include:

- **String (str):** For text, such as `student_name = "John"`
- **Integer (int):** For whole numbers, such as `age = 15`
- **Float:** For decimal numbers, such as `average_mark = 75.5`
- **Boolean (bool):** For true/false values, such as `passed = True`

In the school computer system, variables are used to store each student's name (as a string), marks (as integers or floats), and calculated average (as a float), making it possible to process different types of data accurately.

If-Else Statements and Loops

To make decisions, Python uses **if-else statements**. These allow the program to check conditions and execute different actions based on the result. For example, the school system can use an if-else statement to check if a student's average mark is above 50. If it is, the program displays "Pass"; otherwise, it displays "Fail". This decision-making ability is what makes programs intelligent and responsive.

To process many students efficiently, Python uses **loops** (such as for loops and while loops). Loops repeat a block of code multiple times without the programmer having to write the same instructions

over and over. For instance, a for loop can go through a list of all students, calculate each student's average, and display their results one by one. This is essential for processing large numbers of records quickly and without errors.

CONCLUSION

In conclusion, programming is the process of creating instructions that computers follow to perform tasks, and a program is the final set of these instructions. The difference between source code and machine code is that source code is human-readable while machine code consists of binary numbers that computers understand directly. Python uses variables and data types to store and manage different kinds of information, such as student names and marks. Additionally, if-else statements allow the system to make decisions, while loops enable it to process many students efficiently. Together, these programming concepts make the school computer system powerful, accurate, and capable of handling large amounts of student data with ease.

DAILY ACTIVITIES AND VIVID EXAMPLES

Activity 1: Identifying Computer Hardware (From Question 1)

Daily Life Example: Every morning, a school secretary turns on the computer, types on the **keyboard**, views information on the **monitor**, and saves files to the **hard disk**. These are all examples of computer hardware.

Real-world Application: When students visit a computer shop, they can identify:

- **Hardware:** The physical items they can touch (keyboard, mouse, monitor, system unit)
- **Motherboard:** The large green circuit board inside the system unit that connects everything

Activity 2: Understanding Computer Generations (From Question 3)

Daily Life Example: Baraka visiting the National Museum shows the evolution of computers. Today, students use **smartphones** that are more powerful than the ENIAC computer that filled an entire room.

Real-world Application:

- **First Generation (ENIAC):** Used vacuum tubes, huge, consumed much electricity
- **Fifth Generation (Today):** Uses microprocessors, small, AI-powered, energy efficient

Vivid Example: A student's **calculator** is a special-purpose computer (only for math), while their **laptop** is general-purpose (for games, homework, internet, music).

Activity 3: Computer Maintenance (From Question 4)

Daily Life Example: In a dusty school computer lab, fans become noisy and computers restart automatically—just like Question 4 describes.

Practical Action:

1. **Before cleaning:** Turn off and unplug, wear anti-static wrist strap, remove jewellery
2. **During cleaning:** Use compressed air (NOT vacuum cleaner) to blow dust out
3. **Why clean:** Prevents overheating, extends lifespan, saves repair costs

Vivid Example: Just as dust in a house fan makes it slow and noisy, dust in a computer fan blocks airflow and causes overheating—like a car engine overheating without proper coolant.

Activity 4: File Size Calculations (From Question 5)

Daily Life Example: A student saving research project files needs to know if a 4 GB USB drive is enough.

Real-world Application:

- **Images (2 MB each):** Like 15 photos from a school event
- **Videos (700 MB each):** Like 3 recorded lessons
- **Software installer (1.5 GB):** Like installing Microsoft Office
- **Text documents (50 KB each):** Like 50 assignment reports

Practical Calculation:

Total = 3.58 GB → Fits on 4 GB drive (with 0.42 GB to spare)

Activity 5: Writing Algorithms and Flowcharts (From Question 6)

Daily Life Example: A school accountant calculating total fees paid by students until the last payment is entered.

Practical Exercise: Students can practice by:

1. Writing steps to calculate daily pocket money spent
2. Drawing a flowchart for "enter marks until -1 is entered"
3. Identifying sequence, selection, and iteration

Vivid Example: Like counting money in a piggy bank—you add each coin until the bank is empty, then display the total.

Activity 6: Cyber Security Awareness (From Question 7)

Daily Life Example: A student's social media account is hacked and friends receive fake money requests.

Practical Actions:

1. **Change password** immediately (like changing house locks after losing keys)
2. **Tell friends** not to respond (like warning neighbours about a thief)
3. **Enable 2FA** (like having both a key and a security code to enter a building)

Vivid Example: Two-factor authentication is like an ATM—you need both your **card** (something you have) and your **PIN** (something you know) to withdraw money.

Activity 7: C Programming Practice (From Question 8)

Daily Life Example: A mobile money agent wants to calculate total daily deposits.

Practical Exercise: Write a C program that:

- Asks for customer name
- Loops to accept multiple deposit amounts
- Displays total deposited

Vivid Example: Like a shop cashier adding prices of items in a customer's basket—the program keeps a running total.

Activity 8: Python Programming Practice (From Question 9)

Daily Life Example: A student writing a Python program to check if test marks are passing or failing.

Practical Exercise:

1. Write a program that asks for a mark and displays "Pass" or "Fail"
2. Write a program that asks for a number and displays "Even" or "Odd"

Vivid Example: Python's input() is like a teacher asking a question and waiting for the student's answer—the program pauses until the user types something.

Activity 9: Essay Writing on Programming Concepts (From Question 10)

Daily Life Example: A school computer system storing student records.

Practical Application:

- **Introduction:** State what the essay will cover
- **Body:** Explain programming, program, source code vs machine code, variables, data types, if-else, and loops with examples
- **Conclusion:** Summarize and connect back to the school scenario

Vivid Example:

- **Programming:** Like writing a recipe for cooking
- **Program:** The final recipe card

- **Source code:** Recipe written in English (human-readable)
- **Machine code:** Recipe translated into cooking actions (machine-executable)
- **Variables:** Like labelled containers (e.g., "student_name", "average_mark")
- **Data types:** What fits in each container (text, numbers, true/false)
- **If-else:** Like checking if a student passed or failed
- **Loops:** Like marking attendance for all students in a class

SUMMARY MARKING GRID

Question	Section	Marks	Topic
1	A	10	Multiple Choice - Hardware, Memory, Security
2	A	5	Matching - Computer Components
3	B	10	Computer Generations and Types
4	B	10	Computer Maintenance
5	B	10	File Size Calculations
6	B	10	Algorithms and Flowcharts
7	B	10	Cyber Security
8	B	10	C Programming
9	B	10	Python Programming
10	C	15	Essay on Programming Concepts
TOTAL		100	