Computer Science Paper 2

Basic Programming Skills

Model Paper 2025

Time Allowed: 1 hour 45 minutes
Total Marks: 65

You must answer on the question paper.

You must bring a soft pencil (preferably type B or HB), a clean eraser, and a dark blue or black pen. You may use a simple calculator if needed.

Before attempting the paper, write your name, candidate number, centre name, and centre number clearly in the designated spaces.

Instructions for Candidates

- Answer all questions.
- Write your answer to each question in the space provided.
- You must show all necessary working clearly.
- Do not use an erasable pen or correction fluid.
- Avoid writing over any barcodes printed on the paper.

Information for Candidates

- This paper consists of a total of **65 marks**.
- The number of marks assigned for every question or its parts is indicated within brackets [ ].

Please read all questions carefully and follow the instructions exactly to ensure your responses are properly evaluated.

Q1.

(a) A library database must allow quick access to thousands of digital records.
Review the most appropriate file organisation and access method for this situation. [3]

.....
.....
.....
.....
.....
.....
.....
.....

(b) The string AAAAABBBCCD is compressed using Run-Length Encoding (RLE).
Show the compressed form. [2]

.....
.....
.....
.....
.....

(c) Explain the difference between lossy and lossless compression methods. [2]

Lossy compression	**Lossless compression**

Q1 TOTAL: 7

Q2. A school maintains information about its students in a database table with the following structure:

Field	Data type
StudentID (Primary Key)	String
Name	String
Grade	Integer

(a) Compose an SQL query that retrieves the names and grades of students who scored above 80. [3]

……
……
……
……
……
……

(b) Additionally, the database includes a TEACHER table and a CLASS table with the following definitions:

TEACHER

Field	Data type
TeacherID (primary key)	String
Name	String
Department	String
Email	String

CLASS

Field	Data type
ClassID (primary key)	String
TeacherID	String
Subject	String
Schedule	Date/Time

Elaborate on the concept of a foreign key, with reference to the two tables: TEACHER and CLASS, and clarify its significance in maintaining relationships in the database. [2]

……
……
……
……
……
……

(c) Create an SQL statement that increases the grade of all students in Grade 10 by 5%. [2]

……
……
……
……
……
……

(d) Provide a definition for a new table that records the history of student grades over time. Your design should include a composite primary key. [2]

……
……
……
……
……
……

Q2 TOTAL: 9

Q3. A bank system requires a data structure to store the following daily transaction amounts:

2000, 3500, 4200, 1800, 5000

(a) Identify the most appropriate data structure for storing these values and explain why it is suitable. [2]

……
……
……
……
……
……
……
……

(b) Compose pseudocode to calculate the total of all transactions and display a message if the total exceeds 10,000. [5]

(c) Analyze how the second transaction value (3500) can be retrieved from your chosen data structure, and then write pseudocode to access and display it with a suitable message. [2]

Q3 TOTAL: 9

Q4.

The pseudocode below demonstrates a stack implemented using an array:

```
Initialize Stack[5] and Top = 0
Push 10
Push 20
Pop
Push 30
Output Stack contents
```

(a) Fill the trace table to show the content of the stack after each operation. [4]

Operation	Top	Stack[1]	Stack[2]	Stack[3]	Stack[4]	Stack[5]

(b) What error will occur if a Push operation is attempted when Top = 5? [2]

……
……
……
……

c. Suggest how to handle this error in the pseudocode. [2]

……
……
……
……

Q4 TOTAL: 8

Q5.

A program is tasked with the following:

- Accept 30 integer values representing students' scores.
- Identify and display the smallest score.
- Calculate and display the average score, excluding the smallest score.

Draw a program flowchart to represent this algorithm. [6]

Note:

- No need for variable declarations.
- No need to check for duplicate input values.

Q5 TOTAL: 6

Q6.

(a) Identify whether the following are examples of 1D arrays, 2D arrays, or records: [3]

Marks of 5 students in 3 subjects	
A customer profile containing name, age, and address	
Student names in a class	

(b) A text file marks.txt contains a list of students' marks, one per line. The following pseudocode finds the highest mark:

```
Initialize Highest = 0
Open File "marks.txt"
While not End of File
   Read Mark
   If Mark > Highest Then
      Highest = Mark
   End If
End While
Output Highest
Close File
```

Analyze what would happen if the file is empty. Explain the problem and suggest how the pseudocode could be modified to handle this case. [3]

……
……
……
……
……

Q6 TOTAL: 6

Q7.

(a) Recognize two limitations of a file-based data management approach. [2]

……
……
……
……
……

(b) Outline the key features of a relational database that help overcome the limitations of a file-based system. [4]

……
……
……
……
……
……
……

Q7 TOTAL: 6

Q8.

A company maintains an employee database where each employee record must be quickly located using their Employee ID. To make searching faster, the company decides to use a hashing algorithm to store and retrieve records.

(a). Explain why hashing is suitable for locating employee records efficiently in this scenario. [2]

……
……
……
……
……

(b). State one possible issue that may occur when two employee IDs produce the same hash value, and name the term used for this situation. [2]

……
……
……
……
……

Q8 TOTAL: 4

Q9.

The pseudocode provided below determines whether a number entered by the user is even or odd, and then displays the total count of even and odd numbers entered.

```
Start
   EvenCount ← 0
   OddCount ← 0
   REPEAT
      OUTPUT "Enter a number (-1 to stop): "
      INPUT Number
      IF Number MOD 2 = 0 THEN
         EvenCount ← EvenCount + 1
      ELSE
         OddCount ← OddCount + 1
      ENDIF
   UNTIL Number = -1
   OUTPUT "Even numbers: ", EvenCount
   OUTPUT "Odd numbers: ", OddCount
Stop
```

(a) Complete the following trace table for the input sequence: [2]

4, 9, 12, -1

Number	EvenCount	OddCount	Output

(b) i. Identify the logic error that occurs in the program when -1 is entered. [1]

……

……

……

ii. Suggest one correction to prevent this error. [2]

……

……

……

……

……

Q9 TOTAL: 5

Q10.

The pseudocode below declares several variables and arrays used in a student record system.

```
DECLARE Name : STRING
DECLARE Age : INTEGER
DECLARE Marks : REAL
DECLARE Passed : BOOLEAN
DECLARE Students[1:5] : STRING
```

(a) Identify the data type used for each of the following variables: [3]

i. Name

ii. Age

iii. Passed

……

……

……

……

……

……

……

……

……

(b) Explain why the array Students[1:5] is used in this program instead of creating five separate variables. [2]

……

……

……

……

……

Q10 TOTAL: 5

Computer Science Paper 2

Basic Programming Skills

Answering key & Marking Scheme

Model Paper 2025

Time Allowed: 1 hour 45 minutes
Total Marks: 65

You must answer on the question paper.

You must bring a soft pencil (preferably type B or HB), a clean eraser, and a dark blue or black pen. You may use a simple calculator if needed.

Before attempting the paper, write your name, candidate number, centre name, and centre number clearly in the designated spaces.

Instructions for Candidates

- Answer all questions.
- Write your answer to each question in the space provided.
- You must show all necessary working clearly.
- Do not use an erasable pen or correction fluid.
- Avoid writing over any barcodes printed on the paper.

Information for Candidates

- This paper consists of a total of **65 marks**.
- The number of marks assigned for every question or its parts is indicated within brackets [ ].

Please read all questions carefully and follow the instructions exactly to ensure your responses are properly evaluated.

Q1.

(a) A library database must allow quick access to thousands of digital records. Review the most appropriate file organization and access method for this situation. [3]

Answer:

- File Organisation: Indexed file organisation **or** Hash file organisation
  Indexed file organisation: Records are stored sequentially and an index table is used to quickly locate a specific record.
  Hash file organisation: A hash function directly calculates the location of a record, allowing immediate access.
- Access Method: Direct (Random) access
  Records can be retrieved immediately using a key (e.g., book ID or ISBN) without scanning the entire database.
- Reasoning: Using indexed or hashed organisation with direct access ensures fast searching, insertion, and deletion, which is essential for managing thousands of digital records efficiently.

Marking Scheme:

- 1 mark for correctly identifying a suitable file organisation (Indexed or Hash).
- 1 mark for correctly identifying the access method **(**Direct/Random access).
- 1 mark for explaining why this combination is suitable for fast access in a large database.

(b) The string AAAAABBBCCD is compressed using Run-Length Encoding (RLE). Show the compressed form. [2]

Answer:

- Original string: AAAAABBBCCD
- Compressed using RLE: 5A3B2C1D

Explanation:

- Count consecutive occurrences of each character and write the count followed by the character**.**

A appears 5 times → 5A
B appears 3 times → 3B
C appears 2 times → 2C
D appears 1 time → 1D

Marking Scheme:

- 1 mark for correctly counting and encoding each group of characters.
- 1 mark for providing the correct final compressed form: 5A3B2C1D.

(c) Explain the difference between lossy and lossless compression methods. [2]

Basis	Lossy Compression	Lossless Compression
Definition	Reduces file size by removing some data, which may reduce quality.	Reduces file size without losing any data; original file can be fully restored.
Quality	Some quality is lost (e.g., images, audio).	Original quality is preserved exactly.
Example	JPEG image, MP3 audio	PNG image, ZIP file

Marking Scheme:

- 1 mark for correctly explaining lossy and lossless compression (difference in data retention/quality).
- 1 mark for giving a valid example of each compression type.

Q1 TOTAL: 7

Q2. A school maintains information about its students in a database table with the following structure:

Field	Data type
StudentID (Primary Key)	**String**
Name	**String**
Grade	**Integer**

(a) Compose an SQL query that retrieves the names and grades of students who scored above 80. [3]

Answer:

```
SELECT Name, Grade
FROM Students
WHERE Grade > 80;
```

Marking Scheme:

- 1 mark for correctly using the SELECT statement with the required columns.
- 1 mark for specifying the correct table with FROM.
- 1 mark for using the correct condition (WHERE Grade > 80).

(b) Additionally, the database includes a TEACHER table and a CLASS table with the following definitions:

TEACHER

Field	Data type
TeacherID (primary key)	**String**
Name	**String**
Department	**String**
Email	**String**

CLASS

Field	Data type
ClassID (primary key)	**String**
TeacherID	**String**
Subject	**String**
Schedule	**Date/Time**

Elaborate on the concept of a foreign key, with reference to the two tables: TEACHER and CLASS, and clarify its significance in maintaining relationships in the database. [2]

Answer:

- Foreign Key Concept: A foreign key is a field in one table that refers to the primary key in another table to establish a relationship between the two tables.
- Example in TEACHER and CLASS tables:

TeacherID in the CLASS table is a foreign key referencing TeacherID in the TEACHER table.
This ensures that each class is linked to a valid teacher from the TEACHER table.

- Significance:

Maintains referential integrity, preventing invalid or orphaned records.
Ensures consistent and meaningful relationships between tables in the database.

Marking Scheme:

- 1 mark for correctly defining a foreign key and linking it to TEACHER and CLASS tables.
- 1 mark for explaining its significance in maintaining relationships and referential integrity.

(c) Create an SQL statement that increases the grade of all students in Grade 10 by 5%. [2]

Answer:

```
UPDATE Students
SET Grade = Grade * 1.05
WHERE Grade = 10;
```

Explanation:

- UPDATE Students → specifies the table to update.
- SET Grade = Grade * 1.05 → increases the grade by 5%.
- WHERE Grade = 10 → applies the change only to students in Grade 10.

Marking Scheme:

- 1 mark for correctly using the UPDATE statement and specifying the table.
- 1 mark for using the correct arithmetic operation and WHERE condition to target Grade 10 students.

(d) Provide a definition for a new table that records the history of student grades over time. Your design should include a composite primary key. [2]

Answer:

```
CREATE TABLE StudentGradeHistory (
   StudentID VARCHAR(10),
   Subject VARCHAR(50),
   Grade INT,
   DateRecorded DATE,
   PRIMARY KEY (StudentID, Subject, DateRecorded)
);
```

Explanation:

- StudentID, Subject, and DateRecorded together form a composite primary key, ensuring that each record for a student, subject, and date is unique.
- This table records the history of grades over time, allowing tracking of student performance.

Marking Scheme:

- 1 mark for correctly defining the table with relevant fields.
- 1 mark for correctly implementing a composite primary key.

Q2 TOTAL: 9

Q3. A bank system requires a data structure to store the following daily transaction amounts:

2000, 3500, 4200, 1800, 5000

(a) Identify the most appropriate data structure for storing these values and explain why it is suitable. [2]

Answer:

Most appropriate data structure: **Array**

Explanation:

An array can store multiple values of the same type (e.g., integers) in a contiguous block of memory.

It allows easy access to each transaction using an index and is suitable for a fixed number of daily transactions like 2000, 3500, 4200, 1800, 5000.

Marking Scheme:

- 1 mark for correctly identifying the data structure as an array.
- 1 mark for explaining why it is suitable for storing daily transaction amounts.

(b) Compose pseudocode to calculate the total of all transactions and display a message if the total exceeds 10,000. [5]

Answer:

```
BEGIN
   // Initialize array with transaction amounts
   transactions ← [2000, 3500, 4200, 1800, 5000]
   total ← 0

   // Calculate total of all transactions
   FOR i ← 0 TO length(transactions) - 1 DO
      total ← total + transactions[i]
   END FOR

   // Display total
   PRINT "Total of transactions: ", total
   // Check if total exceeds 10,000
   IF total > 10000 THEN
      PRINT "Warning: Total exceeds 10,000!"
   END IF
END
```

Marking Scheme:

- 1 mark for correctly initializing the array of transactions.
- 1 mark for initializing a variable to store the total.
- 1 mark for correctly using a loop to calculate the total.
- 1 mark for printing the total.
- 1 mark for using the correct condition to display a warning if the total exceeds 10,000.

(c) Analyze how the second transaction value (3500) can be retrieved from your chosen data structure, and then write pseudocode to access and display it with a suitable message. [2]

Answer:

Analysis:

In an array, each element can be accessed using its index**.**

Since array indexing typically starts at 0, the second transaction (3500) is at index 1**.**

Pseudocode:

```
BEGIN
transactions ← [2000, 3500, 4200, 1800, 5000]
secondTransaction ← transactions[1]
PRINT "The second transaction amount is: ", secondTransaction
END
```

Marking Scheme:

- 1 mark for correctly explaining how to retrieve the second transaction using array indexing.
- 1 mark for writing pseudocode that correctly accesses and displays the value with a message.

Q3 TOTAL: 9

Q4.

The pseudocode below demonstrates a stack implemented using an array:

```
Initialize Stack[5] and Top = 0
Push 10
Push 20
Pop
Push 30
Output Stack contents
```

(a) Fill the trace table to show the content of the stack after each operation. [4]

Answer:

Operation	Top	Stack[1]	Stack[2]	Stack[3]	Stack[4]	Stack[5]
Initialize	0					
Push 10	1	10				
Push 20	2	10	20			
Pop	1	10	-			
Push 30	2	10	30			
Output Stack contents	2	10	30			

Marking Scheme:

- 1 mark for correctly updating Top after each operation.
- 3 marks for correctly showing the stack contents after each operation.

(b) What error will occur if a Push operation is attempted when Top = 5? [2]

Answer:

Error on Push when Top = 5:

Stack Overflow Error**:**

Occurs because the stack has reached its maximum capacity, and no more elements can be pushed.

Marking Scheme:

- 2 marks for correctly identifying the error as Stack Overflow and explaining why it occurs.

(c) Suggest how to handle this error in the pseudocode. [2]

Answer:

To handle a stack overflow error, check if the stack is full before performing a Push**.** Only push the element if there is space.

Example pseudocode:

```
IF Top < 5 THEN
   Top ← Top + 1
```

```
    Stack[Top] ← value
ELSE
    PRINT "Error: Stack Overflow! Cannot push more elements."
END IF
```

Marking Scheme:

- 1 mark for suggesting a check to prevent pushing when the stack is full**.**
- 1 mark for providing correct pseudocode or explanation to handle the overflow.

Q4 TOTAL: 8

Q5.

A program is tasked with the following:

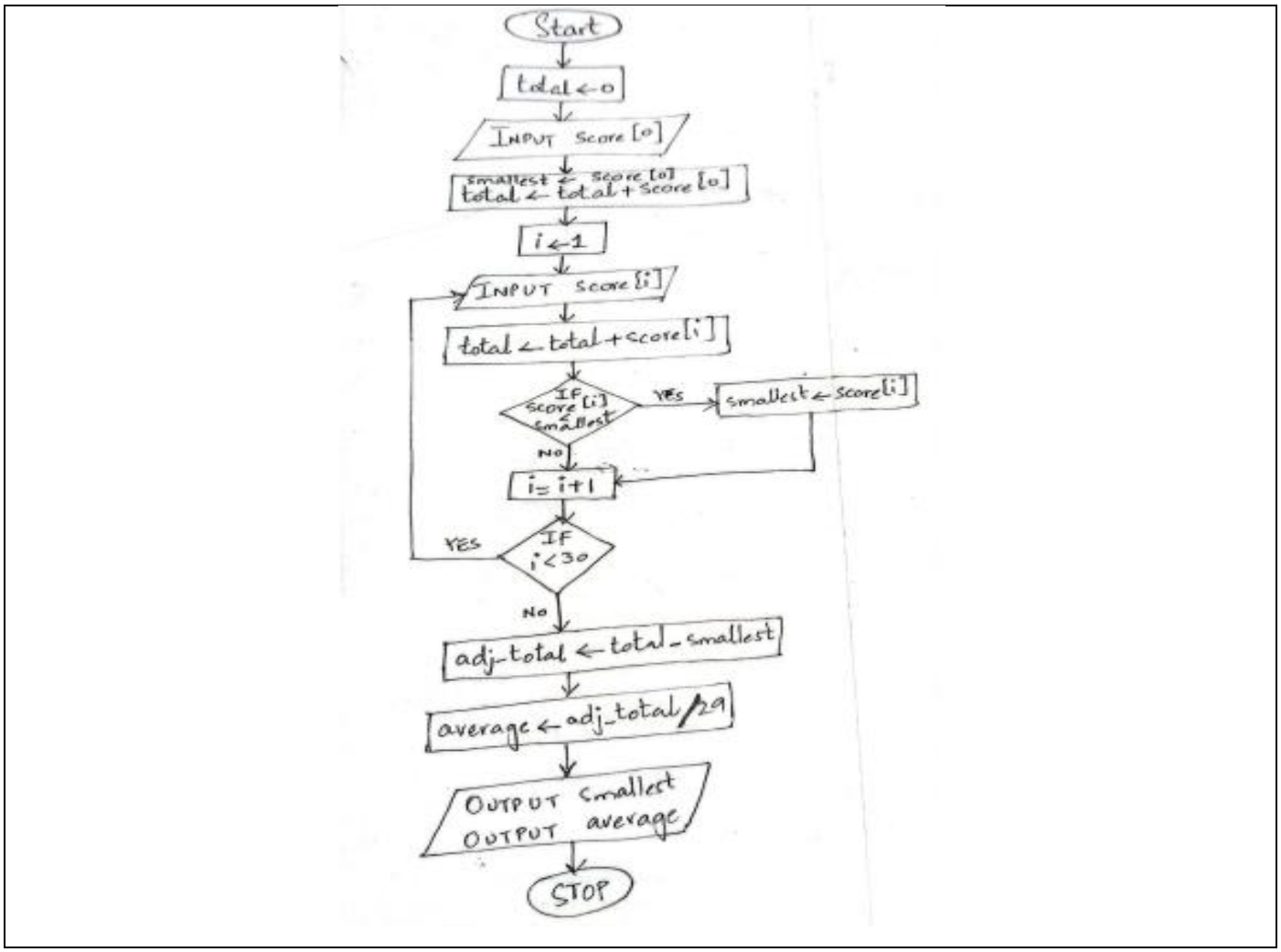
- **Accept 30 integer values representing students' scores.**
- **Identify and display the smallest score**.
- **Calculate and display the average score, excluding the smallest score.**

Draw a program flowchart to represent this algorithm. **[6]**

Note:

- **No need for variable declarations.**
- **No need to check for duplicate input values.**

Answer:



Marking Scheme:

1 mark: For accepting 30 integer input values (loop or repeated input shown).

1 mark: For initializing `total` and `smallest` correctly before the loop.

1 mark: For correctly identifying the smallest score within the loop using comparison.
1 mark: For correctly calculating the total of all scores.
1 mark: For correctly computing the average excluding the smallest score (divide by 29).
1 mark: For displaying both the smallest score and the average.

Q5 TOTAL: 6

Q6.

(a) Identify whether the following are examples of 1D arrays, 2D arrays, or records: **[3]**

Answer:

Marks of 5 students in 3 subjects	2D Array
A customer profile containing name, age, and address	Record
Student names in a class	1D Array

Marking Scheme:

- 1 mark for correctly identifying each type. (THREE required)

(b) A text file marks.txt contains a list of students' marks, one per line. The following pseudocode finds the highest mark:

```
Initialize Highest = 0
Open File "marks.txt"
While not End of File
   Read Mark
   If Mark > Highest Then
      Highest = Mark
   End If
End While
Output Highest
Close File
```

Analyze what would happen if the file is empty. Explain the problem and suggest how the pseudocode could be modified to handle this case. **[3]**

Answer:

Problem if the file is empty:

If marks.txt is empty, the variable Highest remains 0.

This may incorrectly suggest that the highest mark is 0, which is misleading because no marks exist in the file.

Suggested modification to handle this case:

```
Initialize Highest = -1  // or a sentinel value indicating no marks
Open File "marks.txt"
IF End of File THEN
   PRINT "No marks available."
ELSE
   While not End of File
```

```
        Read Mark
        If Mark > Highest Then
            Highest = Mark
        End If
    End While
    Output Highest
End If
Close File
```

Using a sentinel value or checking if the file is empty ensures the program can inform the user that no data exists, rather than giving an incorrect highest mark.

Marking Scheme:

- 1 mark for identifying the problem if the file is empty.
- 1 mark for suggesting a sentinel value or file-empty check.
- 1 mark for modifying the pseudocode to handle the empty file correctly.

Q6 TOTAL: 6

Q7.

(a) Recognize two limitations of a file-based data management approach. [2]

Answer:

Two limitations of a file-based data management approach:

1. **Data Redundancy and Inconsistency:** The same data may be stored in multiple files, leading to duplication and potential inconsistency.
2. **Difficulty in Data Retrieval:** Searching, updating, or generating reports from multiple files is inefficient and time-consuming compared to using a database system.

Marking Scheme:

- 1 mark for each correctly identified limitation (TWO required).

(b) Outline the key features of a relational database that help overcome the limitations of a file-based system. [4]

Answer:

Key features of a relational database that help overcome file-based system limitations:

1. **Data Integrity:** Enforces rules (like primary and foreign keys) to maintain accurate and consistent data.
2. **Elimination of Redundancy:** Stores data centrally in tables, reducing duplication across multiple files.
3. **Efficient Data Retrieval:** Supports powerful querying using SQL to access, filter, and sort data quickly.
4. **Data Relationships:** Allows linking of tables through keys, enabling complex data analysis and reporting.

Marking Scheme:

- 1 mark for each correctly identified feature (FOUR required).

Q7 TOTAL: 6

Q8.

A company maintains an employee database where each employee record must be quickly located using their Employee ID**. To make searching faster, the company decides to use a** hashing algorithm **to store and retrieve records.**

(a) Explain why hashing is suitable for locating employee records efficiently in this scenario. [2]

Answer:

- Hashing is suitable because it computes a unique index (hash value) from the Employee ID, allowing direct access to the record without searching through all records.
- This provides constant-time (O(1)) retrieval**,** making locating employee records fast and efficient even in a large database.

Marking Scheme:

- 1 mark for explaining that hashing generates an index for direct access.
- 1 mark for mentioning efficiency / constant-time retrieval.

(b) State one possible issue that may occur when two employee IDs produce the same hash value, and name the term used for this situation. [2]

Answer:

- **Issue:** Two employee IDs producing the same hash value can cause conflicts in storing or retrieving records**,** as both records may try to occupy the same location.
- **Term:** This situation is called a collision**.**

Marking Scheme:

- 1 mark for identifying the issue (conflict in storage/retrieval).
- 1 mark for naming the term collision**.**

Q8 TOTAL: 4

Q9.

The pseudocode provided below determines whether a number entered by the user is even or odd, and then displays the total count of even and odd numbers entered.

```
Start
   EvenCount ← 0
   OddCount ← 0
   REPEAT
      OUTPUT "Enter a number (-1 to stop): "
      INPUT Number
      IF Number MOD 2 = 0 THEN
         EvenCount ← EvenCount + 1
      ELSE
         OddCount ← OddCount + 1
      ENDIF
   UNTIL Number = -1
   OUTPUT "Even numbers: ", EvenCount
   OUTPUT "Odd numbers: ", OddCount
Stop
```

(a) Complete the following trace table for the input sequence: 4, 9, 12, -1 [2]

Answer:

Number	EvenCount	OddCount	Output
4	1	0	
9	1	1	

12	2	1	
-1	2	1	
			Even numbers: 2 Odd numbers: 1

Marking Scheme:

- 1 mark for correctly updating EvenCount and OddCount for all inputs except the sentinel (-1).
- 1 mark for correctly showing the final output after the loop.

(b) (i) Identify the logic error that occurs in the program when -1 is entered. [1]

Answer:

The program counts **-1** as an odd number before checking the stop condition.

This happens because the condition UNTIL Number = -1 is evaluated **after** incrementing the counters, so -1 is still processed inside the loop.

Marking Scheme:

1 mark for identifying that -1 is incorrectly counted as odd.

ii. Suggest one correction to prevent this error. [2]

Answer:

Add a check before updating EvenCount and OddCount to ensure -1 is not processed.

```
REPEAT
   OUTPUT "Enter a number (-1 to stop): "
   INPUT Number
   IF Number <> -1 THEN
      IF Number MOD 2 = 0 THEN
         EvenCount ← EvenCount + 1
      ELSE
         OddCount ← OddCount + 1
      ENDIF
   ENDIF
UNTIL Number = -1
```

Marking Scheme:

- 1 mark for suggesting to check Number <> -1 before counting.
- 1 mark for correct placement of the condition or showing corrected pseudocode.

Q9 TOTAL: 5

Q10.

The pseudocode below declares several variables and arrays used in a student record system.

```
DECLARE Name : STRING
DECLARE Age : INTEGER
DECLARE Marks : REAL
DECLARE Passed : BOOLEAN
DECLARE Students[1:5] : STRING
```

(a) Identify the data type used for each of the following variables: [3]

Answer:

i. Name – String

ii. Age – Integer

iii. Passed – Boolean

Marking Scheme:

• 1 mark for correctly identifying each data type (All THREE required).

(b) Explain why the array Students[1:5] is used in this program instead of creating five separate variables. [2]

Answer:

The array Students[1:5] is used to store multiple student names under a single variable name. This makes the program more efficient and easier to manage, as it allows the use of loops to process all names instead of writing separate code for each variable.

Marking Scheme:

• 1 mark for stating that the array stores multiple values under one variable name.

• 1 mark for explaining that it simplifies processing and reduces repetitive code.

Q10 TOTAL: 5