

Cambridge Assessment International Education

IGCSE 0478 | O Level 2210

Computer Science

Paper 2

with Solution

for CAIE 2023-25 Syllabus

2015 to 2023

WITH ADDITIONAL PRACTICE QUESTIONS FOR NEW SYLLABUS

Compiled By:

UBAIR KHAN

Bloomfield Hall (JT), LACAS, Cornerstone, Froebel's International

0328-8104135



Airport Road :
Shop 23-24,
Basement Faysal Bank,
Near Yasir Broast,
Airport Road, Lahore.
Mob: 0321-4567519
Tel: 042-35700707

DHA Ph-V:
Plaza No. 52-CCA, Ph-5
DHA Lahore Cantt.
Mob: 0321-4924519
Tel: 042-37180077

Johar Town :
Opp. Beaconhouse JTC
Adjacent Jamia Masjid PIA
Society Shadewal Chowk,
Johar Town Lahore.
Mob: 0313-4567519
Tel: 042-35227007

Bahria Town:
70 - Umer Block
Main Boulevard
Commercial Area
Bahria Town Lahore.
Mob: 0315-4567519
Tel: 042-35342995


Book Title: O Level Computer Science Topical Paper 2

Syllabus Code 2210 | 0478

Compiler: Ubair Khan

Syllabus: Syllabus 2023-

Edition: 2024

Published by:  **STUDENTS RESOURCE** Airport Road 0423-5700707

Price: 2150/-

COPYRIGHT ©STUDENTS RESOURCE® 2023

The rights of Students Resource being Publisher of this book has been asserted by him in accordance with the Copy Right Ordinance 1962 of Pakistan.

No part of this publication may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopy, recording, or any information storage and retrieval system, without permission in writing from the Students Resource or under licence from the Registrar Copyright from Intellectual Property Organization Pakistan.

The syllabus contents and questions from past papers used herein are the property of Cambridge Assessment International Education (CAIE). The use of syllabus and questions from past papers used in this book does not vest in the author or publisher any copyright ownership, nor does the use of CAIE material imply any affiliation with CAIE.

Any individual or institution violating the copyrights will be prosecuted in the court of law under the lex-forei of Pakistan at his/their expense.

No further notes and legal warning would be issued for any kind of legal activity.

Legal Advisor



Content

Topic 1: Locating Errors.....	4
Solution.....	38
Topic 2: Short Algorithms	53
Solution.....	78
Topic 3: Trace Tables.....	92
Solution.....	153
Topic 4: Programming Concepts.....	178
Solution.....	207
Topic 5: Validation & Verification.....	225
Solution.....	237
Topic 6: Test Data.....	245
Solution.....	253
Topic 7: Databases.....	257
Solution.....	310
Topic 8: Check Digit.....	331
Solution.....	335
Topic 9: Boolean Logic.	337
Solution.....	388
Topic 10: Programming.....	417
Solution.....	454

TOPIC

Question
Numbering
in the Topic's
Section

Exam Session
MJ: May/June
ON: October/November

Original
question number
in Past Paper

Year

Variant

Locating Errors & Inefficiencies

Q1

(MJ15/21)

2 Read this section of program code that should input 10 positive numbers and then output the smallest number input.

```
1 Small = 0
2 Counter = 0
3 REPEAT
4   INPUT Num
5   IF Num < Small THEN Num = Small
6   Counter = Counter + 1
7   PRINT Small
8 UNTIL Counter < 10
```

There are four errors in this code.

Locate these errors and suggest a corrected piece of code for each error.

1

2

3

4

.....[4]

PAPER 2

TOPIC 1

LOCATING ERRORS

Compiled By:

UBAIR KHAN

Bloomfield Hall (JT), LACAS, Cornerstone, Froebel's International
0328-8104135

2 Read this section of program code that should input 10 positive numbers and then output the smallest number input.

```
1 Small = 0
2 Counter = 0
3 REPEAT
4     INPUT Num
5     IF Num < Small THEN Num = Small
6     Counter = Counter + 1
7     PRINT Small
8 UNTIL Counter < 10
```

There are **four** errors in this code.

Locate these errors and suggest a corrected piece of code for each error.

1

.....

2

.....

3

.....

4

.....[4]

2 Read this section of program code that should input 30 positive numbers and then output the largest number input.

```
1 Large = 9999
2 Counter = 0
3 WHILE Counter > 30
4 DO
5     INPUT Num
6     IF Num < Large THEN Large = Num
7     Counter = Counter - 1
8 ENDWHILE
9 PRINT Large
```

There are **four** errors in this code.

Locate these errors and suggest a corrected piece of code for each error.

1

.....

2

.....

3

.....

4

.....

.....[4]

2 Read this section of program code that should input 50 numbers and then output the average.

```
1 Total = 0
2 For Counter = 1 TO 50
3     INPUT Num
4     Total = Total + 1
5     Counter = Counter + 1
6     Average = Total/Counter
7 NEXT Counter
8 PRINT Average
```

There are **four** errors in this code.

Locate these errors and suggest code corrections to remove each error.

1

.....

2

.....

3

.....

4

.....

.....[4]

2 Read this section of program code that should input 50 numbers and then output the average of the positive numbers only.

```
1 Total = 0
2 PosCount = 0
3 FOR Counter = 1 TO 50
4   INPUT Num
5   IF Num < 0 THEN Total = Total + Num
6   IF Num > 0 THEN Counter = Counter + 1
7   Average = Total/PosCount
8 NEXT Counter
9 PRINT Num
```

There are **four** errors in this code.

Locate these errors and suggest code corrections to remove each error.

1

.....

2

.....

3

.....

4

.....

.....[4]

- 2 Read this section of program code that inputs positive numbers, discards any negative numbers and then outputs the average. An input of zero ends the process.

```

1 Total = 0
2 Counter = 100
3 REPEAT
4     REPEAT
5         INPUT Num
6     UNTIL Num < 0
7     Total = Total + 1
8     Counter = Counter + Num
9 UNTIL Num = 0
10 Average = Total / (Counter - 1)
11 Print Average
    
```

There are four errors in this code.

Locate these errors and suggest a correction to remove each error.

Error 1

Correction

Error 2

Correction

Error 3

Correction

Error 4

Correction

..... [8]

2 Read this section of program code that:

- inputs 10 numbers
- checks whether each number is within a specified range
- totals the numbers within the range and outside the range

```

1 InRange = 0
2 OutRange = 1000
3 FOR Count = 1 TO 10
4     INPUT Num
5     IF Num > 10 AND Num < 20 THEN InRange = InRange + 1
6     ELSE OutRange = OutRange - 1
7     Count = Count + 1
8 NEXT X
9 PRINT InRange, OutRange
    
```

(a) There are four errors in this code.

Locate these errors and suggest a correction to remove each error.

Error 1

Correction

Error 2

Correction

Error 3

Correction

Error 4

Correction

.....[4]

(b) Decide, with reasons, whether the numbers 10 and 20 are within or outside the range.

Number	Within range (✓)	Outside range (✓)	Reason
10		
20		

[4]

- 2 This section of program code asks for 50 numbers to be entered. The total and average of the numbers are calculated.

```

1 Total = 0
2 Counter = 50
3 PRINT 'When prompted, enter 50 numbers, one at a time'
4 REPEAT
5     PRINT 'Enter a number'
6     INPUT Number
7     Total + Number = Total
8     Number = Number + 1
9 UNTIL Counter = 50
10 Average = Number * Counter
11 PRINT 'The average of the numbers you entered is ', Average
    
```

There are **four** errors in this code.

State the line number for each error and write the correct code for that line.

Error 1 Line number

Correct code

Error 2 Line number

Correct code

Error 3 Line number

Correct code

Error 4 Line number

Correct code

[4]

- 4 An algorithm has been written in pseudocode to input 100 numbers and print out the sum. A REPEAT ... UNTIL loop has been used.

```
Count ← 0
Sum ← 0
REPEAT
  INPUT Number
  Sum ← Sum + Number
  Count ← Count + 1
UNTIL Count > 100
PRINT Sum
```

- (a) Find the error in the pseudocode and suggest a correction.

Error.....

Correction

.....

[2]

- (b) Rewrite the correct algorithm using a more suitable loop structure.

.....

.....

.....

.....

.....

.....

.....

[3]

- 2 This section of program code asks for 80 numbers between 100 and 1000 to be entered. It checks that the numbers are in the correct range, and stores them in an array. It counts how many of the numbers are larger than 500 and then outputs the result when the program is finished.

```

1 Count = 0
2 FOR Index = 1 TO 80
3   INPUT 'Enter a number between 100 and 1000', Number
4   WHILE Number = 99 AND Number = 1001
5     INPUT 'This is incorrect, please try again', Number
6   ENDWHILE
7   Num[80] = Number
8   IF Number > 500 THEN Count = Count + 1
9   UNTIL Index = 80
10  PRINT Index
11  PRINT ' numbers were larger than 500'
```

There are **four** lines of code that contain errors.

State the line number for each error and write the correct code for that line.

Error 1 Line Number

Correct Code

Error 2 Line Number

Correct Code

Error 3 Line Number

Correct Code

Error 4 Line Number

Correct Code

[4]

4 This is a section of program code.

```
1 Total = 100.00
2 PRINT 'Enter the height of each member of your class, one at a
  time, when prompted'
3 FOR Count = 1 TO 30
4   PRINT 'Enter a height in metres'
5   INPUT Height
6   Total = Total + Height
7   PRINT Total / 30
8   Count = Count + 1
9 NEXT Count
```

(a) There are **three** errors in this code.

State the line numbers that contain the errors and describe how to correct each error.

Error 1

.....

.....

Error 2

.....

.....

Error 3

.....

.....

[3]

(b) State the purpose of this program.

.....

.....

.....

[1]

- 2 (a) An algorithm has been written in pseudocode to input 100 numbers, select and print the largest number and smallest number.

```

Count ← 1
INPUT Number
High ← Number
Low ← Count
REPEAT
    INPUT Number
    IF Number > High
        THEN
            High ← Number
    ENDIF
    IF Number > Low
        THEN
            Low ← Number
    ENDIF
    Count ← Count + 1
UNTIL Count = 99
PRINT "Largest Number is ", Number
PRINT "Smallest Number is ", Low
    
```

Find the **four** errors in the pseudocode and suggest a correction for each error.

Error 1

Correction

Error 2

Correction

Error 3

Correction

Error 4

Correction

[4]

- 2 An algorithm has been written in pseudocode to select a random number using the function RandInt (n), which returns a whole number between 1 and the argument n. The algorithm then allows the user to guess the number.

```

Number ← RandInt(100)
TotalTry ← 1
REPEAT
    PRINT "Enter your guess now, it must be a whole number"
    INPUT Guess
    IF TotalTry > Number
        THEN
            PRINT "Too large try again"
        ENDIF
    IF Guess > Number
        THEN
            PRINT "Too small try again"
        ENDIF
    TotalTry ← Guess + 1
UNTIL Guess <> Number
TotalTry ← TotalTry - 1
PRINT "Number of guesses ", TotalTry
    
```

Find the **four** errors in the pseudocode and suggest a correction to remove each error.

Error 1

Correction

Error 2

Correction

Error 3

Correction

Error 4

Correction

[4]

4 The following pseudocode algorithm uses nested IF statements.

```

IF Response = 1
  THEN
    X ← X + Y
  ELSE
    IF Response = 2
      THEN
        X ← X - Y
      ELSE
        IF Response = 3
          THEN
            X ← X * Y
          ELSE
            IF Response = 4
              THEN
                X ← X / Y
              ELSE
                OUTPUT "No response"
            ENDIF
          ENDIF
        ENDIF
      ENDIF
    ENDIF
  ENDIF
ENDIF

```

(a) Name the type of statement demonstrated by the use of IF ... THEN ... ELSE ... ENDIF

.....
 [1]

(b) Re-write the pseudocode algorithm using a CASE statement.

.....

 [4]

- 3 (a) An algorithm has been written in pseudocode to input the names and marks of 35 students. The algorithm stores the names and marks in two arrays Name[] and Mark[]. The highest mark awarded is found and the number of students with that mark is counted. Both of these values are output.

```

01 HighestMark ← 100
02 HighestMarkStudents ← 0
03 FOR Count ← 1 TO 35
04     OUTPUT "Please enter student name"
05     INPUT Name[Count]
06     OUTPUT "Please enter student mark"
07     INPUT Mark[Counter]
08     IF Mark[Count] = HighestMark
09         THEN
10             HighestMarkStudents ← HighestMarkStudents - 1
11     ENDIF
12     IF Mark[Count] > HighestMark
13         THEN
14             Mark[Count] ← HighestMark
15             HighestMarkStudents ← 1
16     ENDIF
17 NEXT Count
18 OUTPUT "There are ", HighestMarkStudents, " with the highest mark of ",
    HighestMark
    
```

Give line numbers where the **four** errors are to be found in the pseudocode. Suggest a correction for each error.

Error 1 line number

Correction

.....

Error 2 line number

Correction

.....

Error 3 line number

Correction

.....

Error 4 line number

Correction

.....

[4]

- 2 An algorithm has been written in pseudocode to check the temperature readings taken from a freezer are within the range -18 degrees to -25 degrees inclusive.

The algorithm counts the number of times that the temperature reading is below -25 degrees and the number of times that the temperature reading is above -18 degrees.

An engineer is called if there are more than 10 temperature readings below -25 degrees.

An alarm sounds if there are more than 5 temperature readings above -18 degrees.

```

01 TooHot ← 0
02 TooCold ← 1000
03 REPEAT
04     OUTPUT "Please enter temperature"
05     INPUT Temperature
06     IF Temperature < -25
07         THEN
08             TooCold ← TooCold - 1
09     ENDIF
10     IF Temperature > -18
11         THEN
12             TooHot ← TooHot + 1
13     ENDIF
14 UNTIL TooHot > 5 OR TooCold > 10
15 IF TooHot < 5
16     THEN
17     INPUT "Alarm!!"
18 ENDIF
19 IF TooCold > 10
20     THEN
21     OUTPUT "Call the Engineer"
22 ENDIF
    
```

- (a) Give the line number(s) from the algorithm of:

an assignment statement

a loop

a counting statement

a selection statement

[4]

Topic 1: Locating Errors & Inefficiencies in Pseudocodes & their Rectification

(b) Give line numbers where the **four** errors are to be found in the pseudocode. Suggest a correction for each error.

Error 1 line number

Correction

.....

Error 2 line number

Correction

.....

Error 3 line number

Correction

.....

Error 4 line number

Correction

.....

[4]

(c) Explain how you could extend the algorithm to count the number of times the temperature readings are within the range -18 degrees to -25 degrees inclusive.

.....

.....

.....

.....

.....

.....

.....

.....

.....

[4]