

Computer Science (J277)

J277/02, Computational thinking, algorithms and programming, June 2023

Neil Bechley

Please note that you may see slight differences between this paper and the original.

Candidates answer on the Question paper.

OCR supplied materials:

Additional resources may be supplied with this paper.

Other materials required:

- Pencil
- Ruler (cm/mm)

Duration: 90 mins

INSTRUCTIONS TO CANDIDATES

- Write your name, centre number and candidate number in the boxes above. Please write clearly and in capital letters.
- Use black ink. HB pencil may be used for graphs and diagrams only.
- Answer **all** the questions, unless your teacher tells you otherwise.
- Read each question carefully. Make sure you know what you have to do before starting your answer.
- Where space is provided below the question, please write your answer there.
- You may use additional paper, or a specific Answer sheet if one is provided, but you must clearly show your candidate number, centre number and question number(s).

INFORMATION FOR CANDIDATES

- The quality of written communication is assessed in questions marked with either a pencil or an asterisk. In History and Geography a *Quality of extended response* question is marked with an asterisk, while a pencil is used for questions in which *Spelling, punctuation and grammar and the use of specialist terminology* is assessed.
- The number of marks is given in brackets [] at the end of each question or part question.
- The total number of marks for this paper is **80**.
- The total number of marks may take into account some 'either/or' question choices.

1(a) The table contains four statements about programming languages.

Tick (✓) **one** box in each row to identify whether each statement describes a low-level programming language or a high-level programming language.

Statement	Low-level	High-level
The same language can be used on computers that use different hardware		
It allows the user to directly manipulate memory		
It allows the user to write English-like words		
It always needs to be translated into object code or machine code		

[4]

(b) The variables `num1` and `num2` store integers.

Write pseudocode to add the integers stored in `num1` and `num2`. Store the result in a variable with the identifier `total`

[1]

(c) Three incomplete pseudocode algorithms are given with a description of the purpose of each algorithm.

Write the missing arithmetic operator for each algorithm.

i. Outputting 12 to the power of 2.

```
print(12 ..... 2)
```

[1]

ii. Working out if a number is odd or even.

```
number = 53
```

```
if number ..... 2 == 0 then
```

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

```
else
```

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

```
endif
```

[1]

iii. Finding the difference between two measurements.

```
measurement1 = 300
```

```
measurement2 = 100
```

```
difference = measurement1 ..... measurement2
```

[1]

2(a) State what is meant by a syntax error and a logic error.

Syntax error -----

Logic error -----

[2]

(b) This pseudocode algorithm totals all the numbers in the 0-indexed array `scores`

```
01 total = 0  
  
02 for scoreCount = 1 to scores.length - 1  
  
03 scores[scoreCount] = total + total  
  
04 next scoreCount  
  
05 print(total)
```

The function `length` returns the number of elements in the array.

The algorithm contains several errors.

Two types of errors in a program are syntax and logic errors.

Identify **two** logic errors in the pseudocode algorithm.

Write the refined line to correct each error.

Error 1 line number

Corrected line

Error 2 line number

Corrected line

[4]

3(a) An insertion sort is one type of sorting algorithm.

A student has written a pseudocode algorithm to perform an insertion sort on a 1D array `names`.

```
names = ["Kareem", "Sarah", "Zac", "Sundip", "Anika"]

for count = 1 to names.length - 1

    pos = count

    while (pos > 0 and names[pos] < names[pos - 1])

        temp = names[pos]

        names[pos] = names[pos - 1]

        names[pos - 1] = temp

        pos = pos - 1

    endwhile

next count
```

Describe the purpose of the variable `temp` in the insertion sort pseudocode algorithm.

[2]

(b) An insertion sort contains a nested loop; a loop within a loop. In this pseudocode algorithm the outer loop is a count-controlled loop and the inner loop is a condition-controlled loop.

Explain why the inner loop needs to be a condition-controlled loop.

[2]

(c) A bubble sort is a type of sorting algorithm.

i. Describe **one** difference between an insertion sort and a bubble sort.

----- [2]

ii. Describe **two** similarities between an insertion sort and a bubble sort.

1 -----

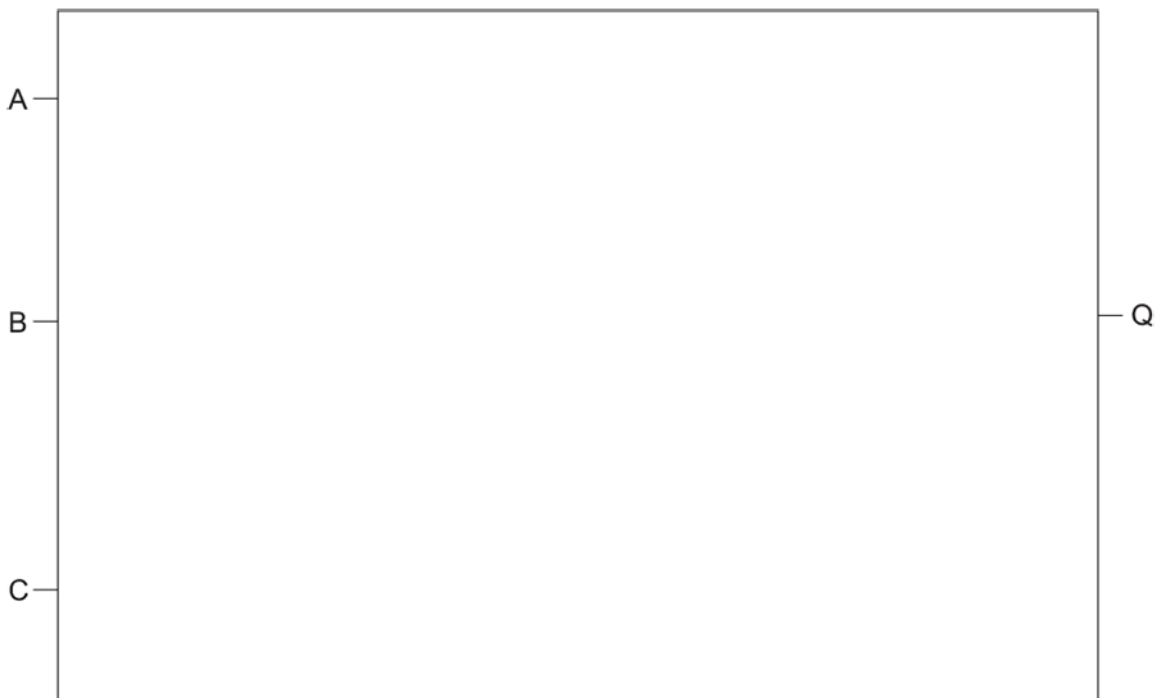
2 -----
----- [2]

4(a) A garden floodlight system uses inputs from sensors and switches to decide whether it should be turned on. The table shows the inputs into the system and the meaning of each input value:

Letter	Input device	Input of 1	Input of 0
A	Motion sensor	Motion is detected	Motion is not detected
B	Light sensor	Light levels indicate it is daytime	Light levels indicate it is nighttime
C	Light switch	The switch is turned on	The switch is turned off

The floodlight (Q) is designed to be on (Q = 1) when the switch is turned on and the motion sensor detects motion at nighttime.

Draw a logic diagram for the floodlight.



[3]

(b) Identify the logic gates for truth table 1 and truth table 2.

Truth table 1:

A	B	Output
0	0	0
0	1	1
1	0	1
1	1	1

Logic gate 1:

Truth table 2:

A	B	Output
0	0	0
0	1	0
1	0	0
1	1	1

Logic gate 2:

[2]

5(a) Charlie is developing an adding game. The rules of the game are:

- the player is asked 3 addition questions
- each question asks the player to add together two random whole numbers between 1 and 10 inclusive
- if the player gets the correct answer, 1 is added to their score
- at the end of the game their score is displayed.

Charlie has been told that the game will need to be tested before giving it to the players.

i. Explain why programs should be tested before use.

[2]

ii. Complete the table by naming and describing **one** type of test that should be used on Charlie's program before releasing it.

Test type	Description

[2]

iii. Complete the table by identifying **and** describing **two** features of an IDE that can be used when testing a program.

Feature	Description

[4]

(b) Validating inputs can reduce errors when a program is being run.

Identify **two** methods of validation **and** explain how they can be used on this game.

Validation method 1

Use

Validation method 2

Use

[6]

(c) Write an algorithm to play a game with the following rules.

- the player is asked 3 addition questions
- each question asks the player to add together two random whole numbers between 1 and 10 inclusive
- if the player gets the correct answer, 1 is added to their score
- at the end of the game their score is displayed.

6(a) OCR Security Services is a company that installs intruder alarm systems in commercial buildings.

The systems use a computer that is connected to the door sensors and window sensors.

The following data is stored in the system:

Data stored	Variable identifier	Example data
The user's name	UserName	Admin123
A telephone number to call when the alarm is activated	EmergencyPhoneNumber	+449999999999
Whether a door sensor is activated	DoorSensorActive	True
Whether a window sensor is activated	WindowSensorActive	True
A timer that counts, to the nearest second, how long a door sensor has been activated	DoorActiveTime	100
A timer that counts, to the nearest second, how long a window sensor has been activated	WindowActiveTime	100
Whether the system is armed	SystemArmed	True
Whether the system is in test mode	TestModeActive	True

Below is a table showing some variables within the program.

Tick (✓) **one** box in each row to identify the most appropriate data type for each variable.

Variable	Boolean	Char	String	Integer	Real
UserName					
EmergencyPhoneNumber					
DoorSensorActive					
DoorActiveTime					

[4]


```
03 sensorNumber = input(" Please input the number of the sensor to reset")
04 sensorID = sensorType + sensorNumber
05 ResetSensor(sensorID)
06 endif
```

i. Give the line number where there is concatenation.

----- [1]

ii. Give the identifier of a variable used in the program.

----- [1]

iii. Identify the data type of the data returned by the function CheckSensorCode ()

----- [1]

iv. Give the line number that contains a function call.

----- [1]

v. Identify **two** programming constructs that have been used in the program.

1 -----

2 ----- [2]

- (f) OCR Security Services need to identify the total number of seconds the sensors have been activated on a specific date.

The data from the database table `events` is imported into the program written in a highlevel programming language.

The program stores the data in a two-dimensional (2D) string array with the identifier `arrayEvents`

The data to be stored is shown in the table.

Date	SensorID	SensorType	Length
05/02/2023	WS2	Window	38
05/02/2023	MS1	Motion	2
06/02/2023	DS3	Door	1
06/02/2023	MS2	Motion	3
06/02/2023	MS1	Motion	2
07/02/2023	WS1	Window	24
07/02/2023	DS1	Door	1

In this table, the value of `events[1, 1]` contains "MS1".

- i. An array can only store data of one data type. Any non-string data must be converted to a string before storing in the array.

Identify the process that converts integer data to string data.

[1]

- ii. Write a program that:

- asks the user to input a date
- totals the number of seconds sensors have been activated on the date input
- outputs the calculated total in an appropriate message including the date, for example:
`Sensors were activated for 40 seconds on 05/02/2023`

You must use **either**:

- OCR Exam Reference Language, or
- A high-level programming language that you have studied.

