JetLearn's GCSE (9–1) Computer Science - 2 Mock Exam
J277/01 Computational thinking, algorithms and programming - Algorithm
Time allowed: 1 hour 30 minutes

INSTRUCTIONS

- Use black ink. You can use an HB pencil, but only for graphs and diagrams.
- Write your answer to each question in the space provided. If you need extra space use the lined pages at the end of this booklet. The question numbers must be clearly shown.
- Answer all the questions.

INFORMATION

- The total marks for this paper is 80.
- The marks for each question are shown in brackets [ ].
- Quality of extended response will be assessed in questions marked with an asterisk (*).
- This document has 16 pages.

ADVICE

- Read each question carefully before you start your answer.

Attempt all questions

1. A fast food restaurant offers half-price meals if the customer is a student or has a discount card.
   The offer is not valid on Saturdays.

   (a) The restaurant needs an algorithm designed to help employees work out if a customer can have a half price meal or not. It should:

   • input required data
   • decide if the customer is entitled to a discount
   • output the result of the calculation.

   Design the algorithm using a flowchart.

[5]

(b) The restaurant adds a service charge to the cost of a meal depending on the number of people at a table. If there are more than five people 5% is added to the total cost of each meal.

Customers can also choose to leave a tip, this is optional and the customer can choose between a percentage of the cost, or a set amount.

Identify all the additional inputs that will be required for this change to the algorithm.

.....

.....

.....

.....[2]

(c) Each member of staff that works in the restaurant is given a Staff ID. This is calculated using the following algorithm.

```
01 surname = input("Enter surname")
02 year = input("Enter starting year")
03 staffID = surname + str(year)
04 while staffID.length < 10
05 staffID = staffID + "x"
06 endwhile
07 print("ID " + staffID)
```

Complete the following trace table for the given algorithm when the surname “Kofi” and the year 2021 are entered.

You may not need to use all rows in the table.

Line number	surname	year	staffID	Output
01	Kofi			
02		2021		

[4]

2. A program stores the following list of positive and negative numbers. The numbers need sorting in ascending order using a merge sort.

45	12	-99	100	-13	0	17	-27

(a) The first step is to divide the list into individual lists of one number each. This has been done for you.
Complete the merge sort of the data by showing each step of the process.

45	12	-99	100	-13	0	17	-27

[3]

(b) Once the numbers are in order, a binary search can be run on the data.

Describe the steps a binary search will follow to look for a number in a sorted list.

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....[4]

(c) A linear search could be used instead of a binary search.

Describe the steps a linear search would follow when searching for a number that is not in the given list.

.....

.....

.....

.....[2]

3. Jack is writing a program to add up some numbers. His first attempt at the program is shown.

```
a = input("Enter a number")
b = input("Enter a number")
c = input("Enter a number")
d = input("Enter a number")
e = input("Enter a number")
f = (a + b + c + d + e)
print(f)
```

(a) Jack decides to improve his program. He wants to be able to input how many numbers to add together each time the algorithm runs, and also wants it to calculate and display the average of these numbers.

Write an algorithm to:

- ask the user to input the quantity of numbers they want to enter and read this value as input
- repeatedly take a number as input, until the quantity of numbers the user input has been entered
- calculate and output the total of these numbers
- calculate and output the average of these numbers.

.....

.....

.....

.....

.....

.....[6]

(b) State the name of each of the following computational thinking techniques.

Description	**Technique**
Breaking a complex problem down into smaller problems.	
Hiding or removing irrelevant details from a problem to reduce the complexity.	

[2]

4. An insertion sort is used to put the following words into ascending alphabetical order.

pumpkin	flour	wall	house	wall

(a) Tick **(√) one** box in each row to identify whether each statement about the insertion sort is true or false.

Statement	**True (√)**	**False (√)**
The list of words is initially split into a sorted set and an unsorted set.		
The insertion sort uses a divide stage and then a conquer stage.		

The list of words must be in order before the insertion sort can start.		
Each word is inserted into the correct place in the array, one by one.		
The insertion sort will not work because the word “wall” appears twice.		

[5]

(b) The sorted list of words is shown below.

flour	house	pumpkin	wall	wall

Explain how a binary search would be used to try to find whether the word “house” appears in this list.

.....

.....

.....

.....

.....

.....

.....

..... [4]

5. Taylor is writing an algorithm to record the results of an experiment.

Taylor needs to be able to enter a numeric value which is added to a total which initially starts at 0.

Every time she enters a value, the total is output.

The algorithm repeats until the total is over 100.

(a) Write an algorithm to implement Taylor's requirements.

.....

.....

.....

.....

.....

.....

.....

.....

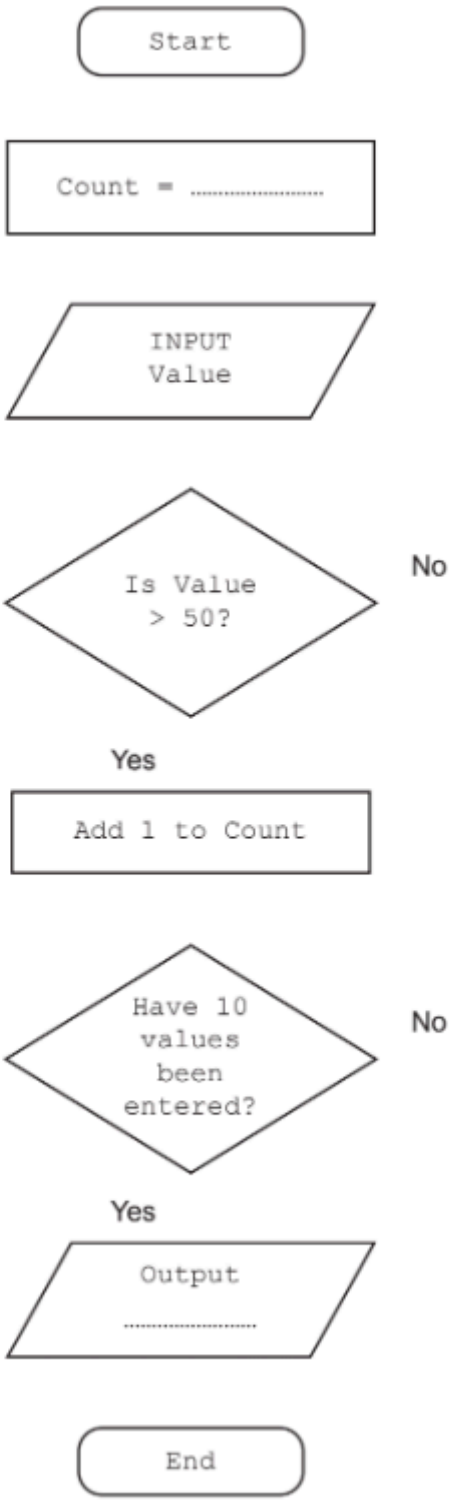
.....

.....

..... [6]

(b) For the next part of the experiment, Taylor needs to be able to enter 10 values and count how many of the values are over 50, outputting this value once all values have been entered.

(i) Complete the following flowchart to implement this algorithm.



[5]

(ii) Write a pseudocode algorithm that uses iteration to allow Taylor to:

- enter 10 values

- count how many values are over 50
- output the count of values over 50 after all 10 values are entered.

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

..... [5]

(c) Taylor used computational thinking techniques to develop the algorithms.

Give two computational thinking techniques that Taylor has used, describing how they have been used.

1 .....

.....

.....

.....

2 .....

.....

.....

.....[4]

6. The following names of students are stored in an array with the identifier studentnames.
studentnames = ["Rob", "Anna", "Huw", "Emma", "Patrice", "Iqbal"]

(a) Describe the steps that a linear search would take to find Anna in studentnames

.....

.....

.....

.....

.....

.....

.....

.....[4]

(b) The names of students are sorted into ascending alphabetical order using an insertion sort.

Complete the following diagram to show the stages an insertion sort would take to complete this task.

Each row represents one pass of the insertion sort algorithm. You may not need to use all empty rows.

Rob	Anna	Huw	Emma	Patrice	Iqbal

[5]

(c) A school uses the array to call an attendance register every morning.

Write an algorithm using iteration to:

- display the name of each student one at a time from studentnames
- take as input whether that student is present or absent
- display the total number of present students and number of absent students in a suitable message, after all student names have been displayed.

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

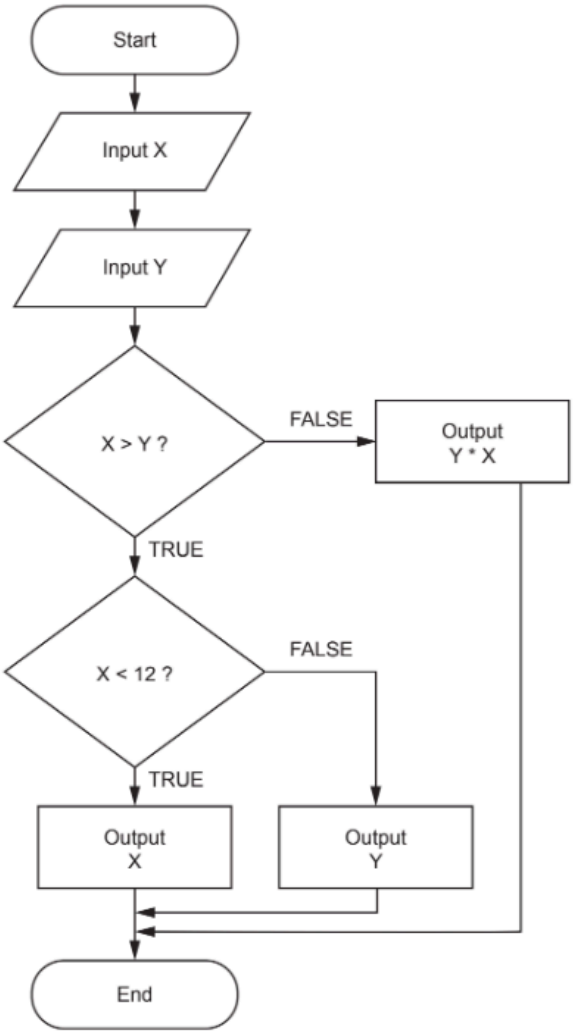
.....

.....

.....

.....[6]

7. A programmer creates an algorithm using a flow chart.



(a) Complete the table to give the output when each of the following set of values are input into the algorithm as X and Y.

Input value of X	Input value of Y	Output
15	10	
6	5	
2	3	
12	2	

[2]

(b) Write this algorithm using pseudocode.

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....[6]

ADDITIONAL ANSWER SPACE

If additional space is required, you should use the following lined page(s). The question number(s) must be clearly shown in the margin(s).

.....

.....

.....

.....