OCR GCSE (J277)

Module M: Programming

Workbooklet · 22 parts · 469 marks

Contents

Part	Topic	Marks
M1	Variables, constants, input and output	22
M3	Data types and casting	21
M5	Arithmetic operators, including MOD, DIV and exponentiation	20
M6	Comparison operators	22
M7	Boolean operators AND, OR and NOT in code	22
M8	Selection: IF, ELSE IF and ELSE	20
M9	Selection: switch/case statements	22
M10	Count-controlled loops (FOR)	22
M11	Condition-controlled loops (WHILE)	21
M12	Post-condition loops, and choosing the right loop	22
M14	Nested selection and nested loops	22
M16	String handling: length, substring, case, concatenation and slicing	22
M18	One-dimensional arrays	22
M19	Two-dimensional arrays and looping through them	21
M21	File handling: open, read, write and close	22
M22	Procedures	22
M23	Functions and return values	22
M24	Parameters and arguments	20
M25	Local and global variables	22
M26	Random numbers	20
M27	Records	20
M28	Writing maintainable code: naming, comments, indentation and sub programs	20
	Total	**469**

MODULE M PROGRAMMING · M1

Variables, constants, input and output

Original practice questions · 22 marks · ~25 minutes · OCR J277 spec 2.2.1

Name _____ Date _____

Instructions. Answer all questions in the spaces provided. The marks for each part are shown in brackets [ ]. The total for this worksheet is **22 marks**. Code on this sheet is written in OCR Exam Reference Language.

1

Total: 3 marks

Tick (✓) **one** box in each row to show whether the statement describes a variable, a constant, or both. **[3]**

Statement	Variable	Constant	Both
Its value can be changed while the program is running.			
Its value stays the same while the program is running.			
It has an identifier (a name) that the program uses to refer to its value.			

2 **Total: 5 marks**

A bike hire kiosk uses this program to work out the cost of hiring a bike.

```
01 const RATE = 4
02 print("Bike hire: price per hour")
03 print(RATE)
04 name = input("Enter your name")
05 hours = int(input("Enter number of hours"))
06 cost = hours * RATE
07 print("Thank you " + name)
08 print(cost)
```

(a) Identify the constant used in the program. **[1]**

(b) Identify **one** variable used in the program. **[1]**

(c) A customer enters `3` when asked for the number of hours. State the value that line 08 outputs. **[1]**

(d) Give **two** reasons why the programmer used the constant `RATE` instead of writing the number 4 on lines 03 and 06. **[2]**

1

2

3 **Total: 5 marks**

Look at this program.

```
01 x = 6
02 y = 4
03 x = x + y
04 y = x * 2
05 x = 3
06 print(x)
07 print(y)
```

(a) Complete the trace table for the program. You may not need to use all the rows. **[3]**

Line	x	y	Output

(b) After line 03 runs, x holds the value 10. Explain why line 06 outputs 3 and not 10. **[2]**

4 **Total: 4 marks**

A campsite has 25 pitches. This program should:

- ask the manager to enter the number of pitches booked
- store the number of pitches still free in the variable `pitchesLeft`
- output the heading `Pitches left`, then the number of pitches still free.

```
01 const PITCHES = 25
02 booked = int(input("Enter number of pitches booked"))
03 PITCHES = PITCHES - booked
04 print(Pitches left)
05 print(pitchesLeft)
```

The program contains **two** errors. Identify the line number of each error and write the corrected line of code. **[4]**

Line number	Corrected line of code

5 **Total: 5 marks**

A print shop charges £0.15 for each page it prints.

Write a program that: **[5]**

- stores the price of one page, 0.15, as a constant
- asks the user to enter the number of pages and stores it in a variable
- calculates the total cost of printing the pages
- outputs the total cost with a suitable message.

You must use either:

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

Total for this worksheet: 22 marks

MODULE M PROGRAMMING · M3

Data types and casting

Original practice questions · 21 marks · ~25 minutes · OCR J277 spec 2.2.2

Name _____ Date _____

Instructions. Answer all questions in the spaces provided. The marks for each part are shown in brackets [ ]. The total for this worksheet is **21 marks**. Code on this sheet is written in OCR Exam Reference Language.

1

Total: 6 marks

A bicycle hire kiosk uses a program to record each hire.

(a) Complete the table by giving the most suitable data type for each item of data. Each data type should be different. **[4]**

Data	Example	Data type
The number of whole hours the bicycle is hired for	`3`	
The price per hour, in pounds	`4.50`	
Whether a helmet is included in the hire	`True`	
A one-letter code for the frame size (S, M or L)	`M`	

(b) The kiosk also stores the customer's phone number, for example `07700 900461`. Explain why the phone number should be stored as a string and not as an integer. **[2]**

2

Total: 4 marks

A self-service laundry's drying machine uses this program to work out the cost of a drying session. It costs 0.2 pounds per minute.

```
01 minutes = input("Enter drying time in minutes")
02 cost = minutes * 0.2
03 print("Cost in pounds: " + cost)
```

(a) State the data type of the value stored in `minutes` by line 01. **[1]**

(b) State what is meant by **casting**. **[1]**

(c) Line 02 causes an error. Rewrite line 02 using casting so that the calculation works. **[1]**

(d) Line 03 also causes an error. Rewrite line 03 using casting so that the message and the cost are output together. **[1]**

3

Total: 4 marks

Each statement in the table is run on its own. Give the value stored in `result` after each statement. Show a string by putting it in speech marks.

[4]

Statement	Value stored in `result`
`result = int("15") + 5`	
`result = "15" + "5"`	
`result = str(9) + "0"`	
`result = float("1.5") * 2`	

4

Total: 3 marks

A program for a car park contains the statement `total = 10`, so `total` stores an integer.

Give the data type of the value produced by each expression. **[3]**

Expression	Data type
`total / 4`	
`total MOD 4`	
`total >= 10`	

5 **Total: 4 marks**

A post office counter uses a program to work out the cost of posting a parcel.

Write a program that: **[4]**

- asks the user to enter the weight of the parcel in kilograms, for example 3.5
- converts the weight that is entered into a real number
- calculates the cost by multiplying the weight by 2.40
- outputs the cost with a suitable message.

You must use either:

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

Total for this worksheet: 21 marks

MODULE M PROGRAMMING · M5

Arithmetic operators, including MOD, DIV and exponentiation

Original practice questions · 20 marks · ~25 minutes · OCR J277 spec 2.2.1

Name _____ Date _____

Instructions. Answer all questions in the spaces provided. The marks for each part are shown in brackets [ ]. The total for this worksheet is **20 marks**. Code is written in OCR Exam Reference Language. No calculator.

1

Total: 5 marks

(a) Complete the table to show the value that each expression gives. **[5]**

Expression	Value
`17 MOD 5`	
`17 DIV 5`	
`3 ^ 4`	
`9 / 2`	
`2 + 3 * 4`	

2

Total: 3 marks

A fitness watch and a garden-planning program use the statements below. Each statement is missing one arithmetic operator.

(a) Complete each statement by writing the missing arithmetic operator in the gap. **[3]**

What the statement must do	Statement
Store the number of **whole** minutes in `totalSeconds`	`minutes = totalSeconds ..... 60`
Store the number of seconds left over after the whole minutes	`seconds = totalSeconds ..... 60`
Store the area of a square flower bed with sides of length `side`	`area = side ..... 2`

3

Total: 3 marks

A machine in a factory makes 100 items in a batch, numbered 1 to 100. This program tells a quality checker which items to inspect.

```
01 for item = 1 to 100
02     if item MOD 25 == 0 then
03         print("Inspect item " + str(item))
04     endif
05 next item
```

(a) State how many times the program outputs a message. **[1]**

(b) State the **first** message the program outputs. **[1]**

(c) Line 02 is changed by mistake to `if item DIV 25 == 0 then` **[1]**
State how many times the program now outputs a message.

4

Total: 4 marks

A small farm packs its eggs into boxes. Each box holds 6 eggs. This program runs for three days.

```
01 for day = 1 to 3
02     eggs = int(input("Eggs collected: "))
03     boxes = eggs DIV 6
04     spare = eggs MOD 6
05     print(str(boxes) + " boxes, " + str(spare) + " spare")
06 next day
```

(a) The farmer enters **20**, then **36**, then **5**. Complete the trace table. Use one row each time round the loop. **[3]**

day	eggs	boxes	spare	Output

(b) On another day the farmer enters **47**. State what the program outputs. **[1]**

5 **Total: 5 marks**

A company that hires out a function room seats guests at round tables. Each table seats 8 guests. A table that is only partly full still counts as one table.

(a) Write a program that: **[5]**

- asks the user to enter the number of guests
- calculates the number of tables needed so that every guest has a seat
- outputs the number of tables needed
- outputs the number of empty seats there will be at those tables.

For example, 41 guests need 6 tables, with 7 empty seats.

You must use either OCR Exam Reference Language or a high-level programming language that you have studied.

Total for this worksheet: 20 marks

MODULE M PROGRAMMING · M6

Comparison operators

Original practice questions · 22 marks · ~25 minutes · OCR J277 specification 2.2.1

Name _____ Date _____

Instructions. Answer all questions in the spaces provided. All code on this sheet is written in OCR Exam Reference Language. The marks for each part are shown in brackets [ ]. The total for this worksheet is **22 marks**.

1

Total: 4 marks

(a) Complete the table. Write the meaning of each comparison operator given, or the comparison operator for the meaning given. **[3]**

Comparison operator	Meaning
`!=`	
	Greater than or equal to
`==`	

(b) State the data type of the result of a comparison, such as `price < 10`. **[1]**

2

Total: 4 marks

A stock-control program for a clothing warehouse has these variables.

```
stock = 12
reorder = 12
size = "M"
boxes = 3
```

(a) Complete the table to show whether each comparison is **True** or **False**. **[4]**

Comparison	True or False
`stock > reorder`	
`stock >= reorder`	
`size != "M"`	
`boxes * 4 <= stock`	

3 **Total: 3 marks**

Write a condition in OCR Exam Reference Language for each rule. Use the variable name given in bold.

(a) A lift sounds an alarm when the load, **load**, is more than 630 kg. **[1]**

(b) A parcel goes in the small-parcel bin when its weight in kg, **weight**, is 2 or less. **[1]**

(c) A smart lock opens when the code entered, stored as a string in **code**, is B4X9. **[1]**

4 **Total: 4 marks**

A program controls the fan and water pump in a garden centre's greenhouse. It should follow these rules:

- the fan is switched on when the temperature is 30 °C **or higher**, otherwise it is switched off
- the pump runs **while** the soil moisture reading is **below** 40. Each time the pump runs, the program adds 5 to the moisture reading.

```
01 temp = int(input("Enter temperature"))
02 if temp > 30 then
03     print("Fan on")
04 else
05     print("Fan off")
06 endif
07 moisture = int(input("Enter moisture reading"))
08 while moisture > 40
09     print("Pump on")
10     moisture = moisture + 5
11 endwhile
12 print("Pump off")
```

(a) The program contains **two** logic errors. Identify the line number of each error and write the corrected line of code. **[4]**

Line number	Corrected line of code

5

Total: 3 marks

A factory checks the speed of a conveyor belt, in cm per second.

```
01 speed = int(input("Enter belt speed"))
02 if speed > 70 then
03     print("Too fast")
04 elseif speed < 30 then
05     print("Too slow")
06 else
07     print("OK")
08 endif
```

(a) Complete the table to show the output of the program for each input. **[3]**

Input	Output
70	
30	
29	

6 **Total: 4 marks**

A farm uses a program to check the water level, in cm, in a cattle trough.

(a) Write an algorithm that: **[4]**

- asks the user to enter the water level
- outputs **Overflow warning** if the level is more than 180
- outputs **Refill needed** if the level is 20 or less
- outputs **Level normal** for any other level.

Only one message should be output.
You must use either: OCR Exam Reference Language, or a high-level programming language that you have studied.

Total for this worksheet: 22 marks

MODULE M PROGRAMMING · M7

Boolean operators AND, OR and NOT in code

Original practice questions · 22 marks · ~25 minutes · OCR J277 spec 2.2.1

Name _____ Date _____

Instructions. Answer all questions in the spaces provided. The marks for each part are shown in brackets [ ]. The total for this worksheet is **22 marks**. Code is written in OCR Exam Reference Language. All data on this sheet is made up.

1

Total: 3 marks

(a) Tick (✓) **one** box in each row to show which Boolean operator matches each description. **[3]**

Description	AND	OR	NOT
Gives True only when both conditions are True.			
Reverses a Boolean value, so True becomes False.			
Gives True when at least one of the conditions is True.			

2

Total: 3 marks

A security unit in a storeroom uses three variables. At one moment they hold these values:

```
temperature = 23
doorOpen = False
alarmSet = True
```

(a) Complete the table to show whether each expression evaluates to True or False for these values. **[3]**

Expression	True or False
`temperature > 20 AND doorOpen`	
`temperature > 30 OR NOT doorOpen`	
`NOT(temperature < 25 AND alarmSet)`	

3 **Total: 4 marks**

A weather station stores each wind speed reading as a whole number in the variable `windSpeed`. A reading is accepted only if it is between 0 and 250 inclusive.

(a) Write a condition that is True when the reading is accepted. **[2]**

(b) Write a condition, using the **OR** operator, that is True when the reading is **not** accepted. **[2]**

4 **Total: 6 marks**

The door of a shared workshop should unlock only when **both** of these are true:

- the hour is from 7 to 19 inclusive
- the person has a keycard, entered as `"Y"` or `"y"`.

The program below has **two** errors.

```
01 hour = int(input("Enter the hour (0 to 23)"))
02 keycard = input("Do you have a keycard? (Y/N)")
03 if hour >= 7 OR hour <= 19 then
04     if keycard == "Y" OR "y" then
05         print("Door unlocked")
06     else
07         print("Keycard needed")
08     endif
09 else
10     print("Workshop closed")
11 endif
```

(a) Identify the line number of each error and write the corrected line of code. **[4]**

Error 1 Line number ..... Corrected line

Error 2 Line number ..... Corrected line

(b) The program is run **before** it is corrected. A person enters an hour of `2`. Explain why the program does **not** output `Workshop closed`. **[2]**

5 **Total: 6 marks**

A city e-bike hire dock decides whether a bike can be hired.

(a) Write a program that: **[6]**

- asks the user to enter the battery level of the bike, as a whole number from 0 to 100, and stores it
- asks the user whether the bike is already booked, entered as `"Y"` or `"N"`, and stores it
- outputs `Available` if the battery level is 20 or more **and** the bike is **not** booked
- otherwise outputs `Not available`.

You must use either:

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

Total for this worksheet: 22 marks

MODULE M PROGRAMMING · M8

Selection: IF, ELSE IF and ELSE

Original practice questions · 20 marks · ~25 minutes · OCR J277 spec 2.2.1, 2.1.2

Name _____ Date _____

Instructions. Answer all questions in the spaces provided. The marks for each part are shown in brackets [ ]. The total for this worksheet is **20 marks**. All code on this sheet is written in OCR Exam Reference Language.

1

Total: 2 marks

Sequence, selection and iteration are the three basic programming constructs.

(a) Describe what is meant by **selection** in a program. **[2]**

2 **Total: 5 marks**

A greenhouse controller sets its cooling fan from the air temperature, in °C.

```
01 temp = int(input("Enter temperature"))
02 if temp > 40 then
03     print("Alarm")
04 elseif temp >= 30 then
05     print("Fan high")
06 elseif temp >= 22 then
07     print("Fan low")
08 else
09     print("Fan off")
10 endif
```

(a) Complete the table to show what the program outputs for each input. **[4]**

Input `temp`	Output
22	
30	
40	
15	

(b) When the input is `45`, the conditions on line 02 **and** line 04 are both true, but only `Alarm` is output. Explain why. **[1]**

3 **Total: 4 marks**

A fitness centre gives each member a badge for the number of classes they attended in a month:

- 12 or more classes: `Gold`
- 8 to 11 classes: `Silver`
- 4 to 7 classes: `Bronze`
- fewer than 4 classes: `None`

A programmer writes this program.

```
01 classes = int(input("Classes attended"))
02 if classes >= 4 then
03     badge = "Bronze"
04 elseif classes >= 8 then
05     badge = "Silver"
06 elseif classes >= 12 then
07     badge = "Gold"
08 else
09     badge = "None"
10 endif
11 print(badge)
```

(a) State what the program outputs when `classes` is 10. **[1]**

(b) The program never outputs `Silver` or `Gold`. Explain why. **[1]**

(c) Refine lines 02 to 07 so that every member gets the correct badge. **[2]**

4 **Total: 3 marks**

A goods lift in a warehouse uses a nested `if` statement to decide whether it can move. The maximum load is 600 kg.

```
01 load = int(input("Load in kg"))
02 doorClosed = input("Door closed? Y/N")
03 if doorClosed == "Y" then
04     if load <= 600 then
05         print("Lift moving")
06     else
07         print("Overloaded")
08     endif
09 else
10     print("Close the door")
11 endif
```

(a) Complete the table to show what the program outputs for each set of inputs. **[3]**

Input `load`	Input `doorClosed`	Output
600	"Y"	
720	"N"	
650	"Y"	

5 **Total: 6 marks**

A parcel shop has storage lockers in three sizes. A program tells staff which locker to use from the length of a parcel's longest side, in cm.

(a) Write a program that: **[6]**

- asks the user to enter the length of the parcel's longest side
- outputs `Invalid` if the length is 0 or less
- outputs `Small` if the length is more than 0 and up to and including 30
- outputs `Medium` if the length is more than 30 and up to and including 50
- outputs `Large` if the length is more than 50 and up to and including 80
- outputs `Too big` if the length is more than 80.

Only **one** message must be output for each parcel.

You must use either: OCR Exam Reference Language, or a high-level programming language that you have studied.

Total for this worksheet: 20 marks

MODULE M PROGRAMMING · M9

Selection: switch/case statements

Original practice questions · 22 marks · ~25 minutes · OCR J277 spec 2.2.1 (with 2.1.2)

Name _____ Date _____

Instructions. Answer all questions in the spaces provided. The marks for each part are shown in brackets [ ]. The total for this worksheet is **22 marks**. Code is written in OCR Exam Reference Language. All data on this sheet is made up.

1

Total: 3 marks

In OCR Exam Reference Language, each `case` in a `switch` statement compares one variable with **one fixed value**.

(a) Tick (✓) **one** box in each row to show whether each task is better written with a `switch` statement or with an `if` statement. **[3]**

Task	switch	if
Choose a paper tray from a paper size code entered as `"A3"`, `"A4"` or `"A5"`.		
Decide whether a parcel with a weight such as 4.75 kg is over 20 kg.		
Choose a fan speed from a temperature such as 18.5: below 20.0 is low, 20.0 to 25.0 is medium, above 25.0 is high.		

2 **Total: 5 marks**

A recycling centre has a kiosk that tells visitors which bin to use.

```
01 material = input("Enter material code")
02 switch material:
03     case "G":
04         print("Green bin")
05     case "P":
06         print("Blue bin")
07     case "M":
08         print("Grey bin")
09     default:
10         print("Ask staff")
11 endswitch
```

(a) Complete the table to show the output for each input. **[3]**

Input	Output
P	
M	
g	

(b) Explain why the input g gives the output you wrote in part (a). **[2]**

3 **Total: 4 marks**

A museum audio guide uses this code. The variable `exhibit` holds a whole number.

```
if exhibit == 1 then
    print("Fossils")
elseif exhibit == 2 then
    print("Ancient coins")
elseif exhibit == 3 then
    print("Clocks")
else
    print("No guide for this number")
endif
```

(a) Rewrite this code using a `switch` statement. Your code must give the same output as the code above for every value of `exhibit`. **[4]**

4 **Total: 4 marks**

A tram ticket machine sets the fare for zones 1, 2 and 3. The program below has **one** logic error.

```
01 zone = int(input("Enter zone (1, 2 or 3)"))
02 switch zone:
03     case 1:
04         fare = 2.00
05     case 2:
06         fare = 3.50
07     case 2:
08         fare = 5.00
09     default:
10         fare = 0
11         print("Invalid zone")
12 endswitch
13 print("Fare: " + str(fare))
```

(a) Identify the line number of the logic error and write the corrected line of code. **[2]**

Line number ..... Corrected line

(b) The program is run **before** it is corrected. The user enters 3. **[2]**

State the output of the program.

5 **Total: 6 marks**

A smart thermostat has three heating modes, chosen by a single letter.

(a) Write a program that uses a `switch` statement to: **[6]**

- ask the user to enter a mode letter and store it
- output `Eco: 18 degrees` if the letter is `"E"`
- output `Comfort: 21 degrees` if the letter is `"C"`
- output `Boost: 24 degrees` if the letter is `"B"`
- output `Invalid mode` for any other input.

You must use either:

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

Total for this worksheet: 22 marks

MODULE M PROGRAMMING · M10

Count-controlled loops (FOR)

Original practice questions · 22 marks · ~25 minutes · OCR J277 spec 2.2.1

Name _____ Date _____

Instructions. Answer all questions in the spaces provided. The marks for each part are shown in brackets [ ]. The total for this worksheet is **22 marks**. Code on this sheet is written in OCR Exam Reference Language. In a `for` loop, the loop variable takes every value from the start value to the end value **inclusive**.

1

Total: 3 marks

(a) State what is meant by a **count-controlled** loop. **[1]**

(b) A program for a dental surgery's reception desk needs two loops. Tick (✓) **one** box in each row to show the type of loop that is most suitable for each task. **[2]**

Task	Count-controlled	Condition-controlled
Print exactly 5 copies of the day's appointment list		
Keep asking for the receptionist's password until the correct password is entered		

2

Total: 3 marks

Each loop below contains one `print` statement inside it.

Complete the table to show how many times the statement inside each loop is run. **[3]**

First line of the loop	Number of times the loop body runs
`for k = 0 to 6`	
`for k = 5 to 20 step 5`	
`for k = 9 to 1 step -2`	

3 **Total: 4 marks**

A farm shop plans how many sacks of potatoes it will have left over four weeks. The algorithm below is used.

```
01 stock = 20
02 for week = 1 to 4
03     sold = week * 3
04     if sold < stock then
05         stock = stock - sold
06     endif
07 next week
08 print("Left: " + str(stock))
```

Complete the trace table for the algorithm. You may not need to use all the rows in the table. **[4]**

Line number	week	sold	stock	Output

4 **Total: 3 marks**

Before a sports hall's lights switch off at night, a display counts down.

Write an algorithm that uses a `for` loop to output the numbers 60, 50, 40, 30, 20, 10 and 0, in that order, and then outputs `Lights off` once. **[3]**

5

Total: 4 marks

A garden centre sells plant pots. This program should ask for the price of one pot, then output the cost of 1, 2, 3, 4, 5 and 6 pots, each on its own line. For example, if one pot costs 2.5, the third line output should be `3 pots cost 7.5`

```
01 price = float(input("Enter the price of one pot"))
02 for pots = 1 to 5
03     cost = price * 6
04     print(str(pots) + " pots cost " + str(cost))
05 next pots
```

The program contains two logic errors. **[4]**

Identify the line number of each logic error and write the corrected line of code.

Line number	Corrected line of code

6

Total: 5 marks

A company's post room weighs the parcels that arrive each morning.

Write a program that: **[5]**

- asks the user to enter how many parcels have arrived
- uses a `for` loop to ask the user to enter the weight of each parcel, in kilograms
- adds up the weights of all the parcels
- outputs the total weight once all the weights have been entered.

You must use either:

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

Total for this worksheet: 22 marks

MODULE M PROGRAMMING · M11

Condition-controlled loops (WHILE)

Original practice questions · 21 marks · ~25 minutes · OCR J277 spec 2.2.1 (with 2.1.2)

Name _____ Date _____

Instructions. Answer all questions in the spaces provided. The marks for each part are shown in brackets [ ]. The total for this worksheet is **21 marks**. Code on this sheet is written in OCR Exam Reference Language.

1

Total: 3 marks

Tick (✓) **one** box in each row to show whether each statement about a `while` loop is true or false. **[3]**

Statement	True	False
The condition is checked before the code inside the loop runs.		
The code inside a `while` loop always runs at least once.		
It is suitable when the number of repeats is not known before the loop starts.		

2 **Total: 6 marks**

A goods lift in a warehouse can carry a total of 100 kg. A program counts how many boxes can be loaded before the next box would take the total over 100 kg. Each box weight is entered in whole kilograms.

```
01 total = 0
02 boxes = 0
03 weight = int(input("Enter box weight"))
04 while total + weight <= 100
05     total = total + weight
06     boxes = boxes + 1
07     weight = int(input("Enter box weight"))
08 endwhile
09 print(boxes)
```

(a) Complete the trace table for the program when the box weights entered are **30, 45, 20, 15**. **[4]**
You may not need to use all the rows.

Line	total	boxes	weight	Output

(b) The program is run again. The first box weight entered is **120**. **[1]**
State what the program outputs.

(c) State why line 03 is needed **before** the loop starts. **[1]**

3 **Total: 4 marks**

An adult is saving 60 pounds each week to buy a 500 pound bicycle. This program should output how many weeks it takes to save at least 500 pounds.

```
01 const TARGET = 500
02 saved = 0
03 weeks = 0
04 while saved < TARGET
05     weeks = weeks + 1
06 endwhile
07 print("Weeks needed: " + str(weeks))
```

(a) When the program is run, it never outputs anything. Explain why. **[2]**

(b) Write the line of code that needs to be added inside the loop so that the program works correctly. **[1]**

(c) State the output of the corrected program. **[1]**

4 **Total: 3 marks**

A heating controller asks for a temperature setting in whole degrees. It must keep asking until a temperature from 10 to 30 inclusive is entered. Two lines of the program are missing.

```
01 temp = int(input("Enter temperature setting"))
02 while ..........................................
03     print("The setting must be from 10 to 30")
04     ..........................................
05 endwhile
06 print("Heating set to " + str(temp))
```

Write the missing code for line 02 and line 04. **[3]**

Line 02 `while`

Line 04

5 **Total: 5 marks**

A gym's rowing machine adds up the distance a member rows in one session. The member enters the distance of each row, one at a time, and enters 0 when they have finished.

Write a program that: **[5]**

- asks the user to enter the distance of a row in metres
- uses a `while` loop to keep asking for distances until 0 is entered
- adds each distance to a running total
- outputs the total distance with a suitable message after 0 is entered.

You must use either:

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

Total for this worksheet: 21 marks

MODULE M PROGRAMMING · M12

Post-condition loops, and choosing the right loop

Original practice questions · 22 marks · ~25 minutes · OCR J277 spec 2.2.1, 2.1.2

Name _____ Date _____

Instructions. Answer all questions in the spaces provided. The marks for each part are shown in brackets [ ]. The total for this worksheet is **22 marks**. All code on this sheet is written in OCR Exam Reference Language.

1

Total: 3 marks

OCR Exam Reference Language has two condition-controlled loops: `while ... endwhile` and `do ... until`.

(a) Tick (✓) **one** box in each row to show which loop each statement describes. **[3]**

Statement	`while`	`do ... until`	Both
The condition is checked before the code inside the loop runs.			
The code inside the loop always runs at least once.			
It is a condition-controlled loop.			

2 **Total: 6 marks**

A coffee roaster tips freshly roasted beans into a cooling tray. This program models the bean temperature, in °C, falling minute by minute.

```
01 temp = 96
02 minutes = 0
03 do
04     temp = temp - 18
05     minutes = minutes + 1
06     print(temp)
07 until temp < 40
08 print("Minutes to cool: " + str(minutes))
```

(a) Complete the trace table for the program. The first row has been done for you. You may not need to use all the rows. **[4]**

temp	minutes	Output
96	0	

(b) Line 01 is changed to `temp = 35`. The rest of the program is unchanged. **[2]**
State everything the program now outputs.

3

Total: 5 marks

(a) Complete the table by writing the most suitable loop for each task: `for`, `while` or `do ... until`. **[3]**

Task	Most suitable loop
A lighthouse lamp flashes exactly 6 times in each cycle.	
A quiz machine shows a question and reads an answer. It must do this at least once, and repeats until the answer is correct.	
A robot vacuum cleaner moves forward as long as its front bumper sensor is not pressed. The robot may already be touching a wall when the program starts.	

(b) Explain why a `do ... until` loop would **not** be suitable for the robot vacuum cleaner. **[2]**

4

Total: 3 marks

A lift display shows each level it passes on its way up.

```
01 for level = 1 to 4
02     print("Level " + str(level))
03 next level
```

(a) Rewrite this program using a `do ... until` loop instead of a `for` loop. Your program must give exactly the same output. **[3]**

5 **Total: 5 marks**

A smart heating controller lets a user set a target room temperature. The target must be a whole number from 10 to 30 inclusive.

(a) Write a program that: **[5]**

- asks the user to enter a target temperature
- uses a condition-controlled loop to keep asking until the target entered is from 10 to 30 inclusive
- then outputs `Target set to` followed by the target.

You may assume the user always enters a whole number.

You must use either: OCR Exam Reference Language, or a high-level programming language that you have studied.

Total for this worksheet: 22 marks

MODULE M PROGRAMMING · M14

Nested selection and nested loops

Original practice questions · 22 marks · ~25 minutes · OCR J277 spec 2.1.2, 2.2.1

Name _____ Date _____

Instructions. Answer all questions in the spaces provided. The marks for each part are shown in brackets [ ]. The total for this worksheet is **22 marks**. Code is written in OCR Exam Reference Language. All data on this sheet is made up.

1

Total: 3 marks

Look at these three pieces of code.

Code A

```
if temperature > 30 then
    if humidity > 80 then
        print("Alert")
    endif
endif
```

Code B

```
for day = 1 to 7
    for hour = 0 to 23
        print(day * hour)
    next hour
next day
```

Code C

```
for i = 1 to 3
    print(i)
next i
for j = 1 to 2
    print(j)
next j
```

(a) Tick (✓) **one** box in each row to show what each piece of code contains. **[3]**

Code	Nested selection	Nested iteration	No nesting
Code A			
Code B			
Code C			

2

Total: 4 marks

Look at this algorithm.

```
01 total = 0
02 for row = 1 to 2
03     for seat = 1 to 3
04         if seat != 2 then
05             total = total + row * seat
06         endif
07     next seat
08 next row
09 print(total)
```

(a) Complete the trace table for this algorithm. You may not need to use all the rows. **[4]**

row	seat	total	Output

3 **Total: 4 marks**

A parcel collection point uses this program to tell staff where to put each parcel.

```
01 weight = real(input("Enter weight in kg"))
02 size = input("Enter size S, M or L")
03 if weight <= 20 then
04     if size == "L" then
05         print("Use a large locker")
06     else
07         print("Use a standard locker")
08     endif
09 else
10     print("Take to counter")
11 endif
```

(a) Complete the table to show the output for each set of inputs. **[3]**

weight	size	Output
12	"M"	
25	"L"	
20	"L"	

(b) State why line 04 is **not** run when the weight entered is 25. **[1]**

4 **Total: 5 marks**

A warehouse has 4 aisles, numbered 0 to 3. Each aisle has 6 shelves, numbered 0 to 5. The 2D array `stock` holds the number of boxes on each shelf, so `stock[aisle, shelf]` is the number of boxes on one shelf.

This program should output, for every aisle, how many of its shelves are empty (hold 0 boxes). The program contains **two** errors.

```
01 for aisle = 0 to 3
02     empty = 0
03     for shelf = 0 to 6
04         if stock[aisle, shelf] == 0 then
05             empty = empty + 1
06         endif
07     next shelf
08     print("Aisle " + str(aisle) + ": " + str(empty) + " empty")
09 next shelf
```

(a) Identify the line number of each error and write the corrected line of code. **[4]**

Error 1 Line number ..... Corrected line

Error 2 Line number ..... Corrected line

(b) The errors have been corrected. State how many times line 04 is run when the program is run once. **[1]**

5 **Total: 6 marks**

A car park has 3 levels, numbered 1 to 3. Each level has 20 parking spaces, numbered 1 to 20. On every level, spaces 1 to 4 are badge spaces for disabled drivers.

(a) Write a program that: **[6]**

- uses nested loops to go through every space on every level
- outputs a code for each space, made of the level number, a dash and the space number, for example `2-15`
- adds `(badge)` after the code when the space number is from 1 to 4, for example `2-3 (badge)`
- counts the badge spaces and outputs this total after all the codes.

You must use either:

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

Total for this worksheet: 22 marks

MODULE M PROGRAMMING · M16

String handling: length, substring, case, concatenation and slicing

Original practice questions · 22 marks · ~25 minutes · OCR J277 spec 2.2.3

Name _____ Date _____

Instructions. Answer all questions in the spaces provided. The marks for each part are shown in brackets [ ]. The total for this worksheet is **22 marks**. Code is written in OCR Exam Reference Language, where the first character of a string is at index 0. All data on this sheet is made up.

1

Total: 5 marks

A ferry company stores the name of a stop in the variable `stop`:

```
stop = "Harbour Lights"
```

(a) Complete the table to show the value given by each expression. **[5]**

Expression	Value
`stop.length`	
`stop.substring(2, 5)`	
`stop.right(6)`	
`stop.upper`	
`stop.left(1) + str(stop.length)`	

2

Total: 2 marks

In ASCII, the character `"A"` has the code 65 and the character `"a"` has the code 97. The codes for the other letters follow on in alphabetical order.

(a) State the value returned by `ASC("D")` **[1]**

(b) State the value returned by `CHR(101)` **[1]**

3

Total: 4 marks

A warehouse scanner tidies up shelf references that have been typed with spaces in them.

```
01 ref = "K4 7T"
02 clean = ""
03 for i = 0 to ref.length - 1
04     letter = ref.substring(i, 1)
05     if letter != " " then
06         clean = clean + letter
07     endif
08 next i
09 print(clean.lower)
```

(a) Complete the trace table for this program. You may not need to use all the rows. **[4]**

i	letter	clean	Output

4 **Total: 6 marks**

A parts supplier only accepts a product code if **both** of these are true:

- the code is exactly 6 characters long
- the first character of the code is `"X"`.

The program below has **two** logic errors.

```
01 code = input("Enter product code")
02 first = code.substring(1, 1)
03 if code.length > 6 AND first == "X" then
04     print("Code accepted")
05 else
06     print("Code rejected")
07 endif
```

(a) Identify the line number of each logic error and write the corrected line of code. **[4]**

Error 1 Line number ..... Corrected line

Error 2 Line number ..... Corrected line

(b) After the errors are corrected, the program still rejects the code `x48213`, which starts with a lower case x. Describe how the `.upper` method could be used so that this code is accepted. **[2]**

5 **Total: 5 marks**

A bus company gives each route a code made from the town it starts in and a route number. For example, route 14 starting in Kelso has the code `KEL-14`

(a) Write a program that: **[5]**

- asks the user to enter the name of the town and stores it
- asks the user to enter the route number and stores it
- creates the route code by joining the first three characters of the town name in upper case, a hyphen (-) and the route number
- outputs the route code.

You must use either:

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

Total for this worksheet: 22 marks

MODULE M PROGRAMMING · M18

One-dimensional arrays

Original practice questions · 22 marks · ~25 minutes · OCR J277 spec 2.2.3 (with 2.1.2 trace tables)

Name _____ Date _____

Instructions. Answer all questions in the spaces provided. The marks for each part are shown in brackets [ ]. The total for this worksheet is **22 marks**. Code is written in OCR Exam Reference Language, where arrays are 0-indexed (the first element has index 0). All data on this sheet is made up.

1

Total: 3 marks

(a) Tick (✓) the **two** statements that are true for an array in OCR Exam Reference Language. **[2]**

Statement	Tick (✓)
Every element in an array must be the same data type.	
The first element of an array has the index 1.	
An array can store a number in one element and a string in another.	
Once an array has been declared, it cannot change size while the program runs.	
`array codes[4]` creates the elements `codes[1]` to `codes[4]`.	

(b) A program needs to store 50 temperature readings. Give **one** reason why the programmer should store them in an array rather than in 50 separate variables. **[1]**

2

Total: 2 marks

A bank of parcel lockers has five locker units. This line of code stores the number of parcels waiting in each unit:

```
array parcels = [12, 0, 7, 16, 9]
```

(a) State the value of `parcels[3]`. **[1]**

(b) The unit that has no parcels waiting now has 15 parcels. Write **one** line of code to change the value `0` in the array to `15`. **[1]**

3 **Total: 2 marks**

A small hotel has five rooms. A programmer writes these two lines of code:

```
01 array guests[5]
02 guests[5] = "Ola Mensah"
```

(a) Line 02 causes an error when the program runs. Explain why. **[2]**

4 **Total: 5 marks**

A warehouse has four delivery robots. The array `levels` stores the battery level of each robot. Any robot with a level below 50 is given a 20-point top-up.

```
01 array levels = [40, 85, 15, 60]
02 low = 0
03 for i = 0 to 3
04     if levels[i] < 50 then
05         levels[i] = levels[i] + 20
06         low = low + 1
07     endif
08 next i
09 print(low)
```

(a) Complete the trace table for the program. You may not need to use all the rows. **[3]**

i	low	Output

(b) Complete the table to show the contents of the array `levels` after the program has finished. **[2]**

Index	0	1	2	3
levels				

5 **Total: 4 marks**

A small cinema records the number of tickets sold on each of the 7 days of a week.

(a) Write a program that: **[4]**

- declares an array called `tickets` that has 7 elements
- uses a loop to ask the user to enter the number of tickets sold on each day, storing each value in the next element of `tickets`.

You must use either:

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

6 **Total: 6 marks**

A museum has six galleries, numbered 0 to 5. The array `visitors` stores the number of visitors in each gallery so far today. It has already been declared and filled:

```
array visitors = [120, 85, 230, 0, 64, 97]
```

(a) Write a program that: **[6]**

- asks the user to enter a gallery number from 0 to 5 and stores it
- adds 1 to the number of visitors stored for that gallery
- then uses a loop to output the number of visitors for every gallery.

You do not need to write the line that declares `visitors`. You do not need to validate the input.
You must use either:

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

Total for this worksheet: 22 marks

MODULE M PROGRAMMING · M19

Two-dimensional arrays and looping through them

Original practice questions · 21 marks · ~25 minutes · OCR J277 spec 2.2.3

Name _____ Date _____

Instructions. Answer all questions in the spaces provided. The marks for each part are shown in brackets [ ]. The total for this worksheet is **21 marks**. Code is written in OCR Exam Reference Language, where arrays are indexed from 0. All data on this sheet is made up.

1

Total: 3 marks

(a) Tick (✓) **one** box in each row to show whether each statement about 2D arrays is true or false. **[3]**

Statement	True	False
Every element in a 2D array must be of the same data type.		
After `array grid[4,6]`, the element `grid[4,6]` can be used to store a value.		
Extra rows can be added to an array while the program is running.		

2 **Total: 5 marks**

A sports centre stores details of its instructors in a 2D array with the identifier `instructors`. The array is used like a database table: each row is one record and each column is one field. Every element is a string. The first index is the row and the second index is the column, so `instructors[0,1]` is `"Okafor"`.

	0 (ID)	1 (Surname)	2 (Activity)	3 (Years)
0	"I01"	"Okafor"	"Swimming"	"12"
1	"I02"	"Lindqvist"	"Climbing"	"3"
2	"I03"	"Haddad"	"Yoga"	"7"
3	"I04"	"Moreau"	"Climbing"	"15"
4	"I05"	"Tanaka"	"Swimming"	"1"

(a) State the value stored in `instructors[2,1]` **[1]**

(b) Instructor `"I05"` now teaches Yoga instead of Swimming. Write **one** line of code to store this change in the array. **[1]**

(c) Write code that uses iteration to output the surname of every instructor whose activity is `"Climbing"`. **[3]**

You must use either:

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

3

Total: 7 marks

A plant nursery has three growing beds. A sensor in each bed takes four soil moisture readings a day. The readings are stored in the 2D array `moisture`. The first index is the bed (0 to 2) and the second index is the reading (0 to 3), so `moisture[0,3]` is `31`.

	Reading 0	Reading 1	Reading 2	Reading 3
Bed 0	42	38	27	31
Bed 1	55	49	44	29
Bed 2	33	26	35	40

(a) State the output when this code is run. **[1]**

```
01 total = 0
02 for reading = 0 to 3
03     total = total + moisture[1, reading]
04 next reading
05 print(total)
```

(b) State the output when this code is run. Write each output on a new line. **[3]**

```
01 for bed = 0 to 2
02     for reading = 0 to 3
03         if moisture[bed, reading] < 30 then
04             print(str(bed) + " " + str(reading))
05         endif
06     next reading
07 next bed
```

(c) Write code that uses iteration to calculate and output the average of reading 3 (the last reading of the day) for the three beds. **[3]**

You must use either:

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

4 **Total: 6 marks**

A self-storage company has 40 lockers in 5 rows of 8. The 2D array `rented` stores whether each locker is rented. The first index is the row (0 to 4) and the second index is the locker in that row (0 to 7). Each element is `True` if the locker is rented and `False` if it is free.

(a) Write a program that: **[6]**

- uses nested loops to check every locker in the array
- counts how many lockers are free
- outputs the number of free lockers
- also outputs `All lockers rented` if no lockers are free.

You must use either:

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

Total for this worksheet: 21 marks

MODULE M PROGRAMMING · M21

File handling: open, read, write and close

Original practice questions · 22 marks · ~25 minutes · OCR J277 spec 2.2.3

Name _____ Date _____

Instructions. Answer all questions in the spaces provided. The marks for each part are shown in brackets [ ]. The total for this worksheet is **22 marks**. Code on this sheet is written in OCR Exam Reference Language.

1

Total: 4 marks

Tick (✓) **one** box in each row to show which file handling operation each description matches. **[4]**

Description	Open	Read	Write	Close
Adds a line of text from the program to the end of the file				
Releases the file once the program has finished using it				
Connects the program to a named file so that it can be used				
Copies the next line of the file into the program, where it can be stored in a variable				

2

Total: 4 marks

A museum's ticket log, `tickets.txt`, holds two lines: `T101` then `T102`. This program then runs.

```
01 logFile = open("tickets.txt")
02 logFile.writeLine("T103")
03 logFile.writeLine("T104")
04 logFile.close()
```

(a) Show the contents of `tickets.txt` after the program has run. You may not need to use all the boxes. **[2]**

Contents of tickets.txt, one line in each box (line 1 on the left)				

(b) The museum also needs a file called `archive.txt`, which does not exist yet. Write the code to create `archive.txt` and then open it, storing the open file in a variable. **[2]**

3 **Total: 4 marks**

A weather station saves one temperature reading on each line of the text file `readings.txt`. Today the file holds these four lines:

```
14
17
9
21
```

This program uses the file.

```
01 total = 0
02 count = 0
03 tempFile = open("readings.txt")
04 while NOT tempFile.endOfFile()
05     reading = int(tempFile.readLine())
06     total = total + reading
07     count = count + 1
08 endwhile
09 tempFile.close()
10 print(total / count)
```

(a) State the value that line 10 outputs. **[1]**

(b) State how many times line 05 is run. **[1]**

(c) Describe the purpose of line 04. **[2]**

4 **Total: 4 marks**

A smart speaker stores its volume level on the first line of the text file `settings.txt`. This program should:

- open `settings.txt`
- read the first line of the file and store it in the variable `volume`
- output the message `Volume:` followed by the volume level
- close the file.

```
01 setFile = open(settings.txt)
02 volume = setFile.writeLine()
03 print("Volume: " + volume)
04 setFile.close()
```

The program contains **two** errors. Identify the line number of each error and write the corrected line of code. **[4]**

Line number	Corrected line of code

5 **Total: 6 marks**

A café records its orders in the text file `orders.txt`. The file is empty at the start of the day.

Write a program that: **[6]**

- opens `orders.txt`
- asks the user to enter an order, again and again, until the user enters `stop`
- writes each order to the file as a new line (`stop` must not be written to the file)
- closes the file
- outputs the number of orders written to the file.

You must use either:

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

Total for this worksheet: 22 marks

MODULE M PROGRAMMING · M22

Procedures

Original practice questions · 22 marks · ~25 minutes · OCR J277 spec 2.2.3, 2.3.1

Name _____ Date _____

Instructions. Answer all questions in the spaces provided. The marks for each part are shown in brackets [ ]. The total for this worksheet is **22 marks**. Code is written in OCR Exam Reference Language. All data on this sheet is made up.

1

Total: 4 marks

(a) Tick (✓) **one** box in each row to show whether each statement about procedures is true or false. **[3]**

Statement	True	False
A procedure must have at least one parameter.		
A procedure can be called more than once in the same program.		
The code inside a procedure only runs when the procedure is called.		

(b) State **one** difference between a procedure and a function. **[1]**

2 **Total: 6 marks**

A ticket printer at a craft fair runs this program.

```
01 procedure printLine()
02     print("----------")
03 endprocedure
04 procedure printHeader()
05     printLine()
06     print("Spring Craft Fair")
07 endprocedure
08 printHeader()
09 for ticket = 1 to 2
10     print("Ticket " + str(ticket))
11     printLine()
12 next ticket
13 print("Enjoy your visit")
```

(a) Write the output when this program is run. Write one line of output on each answer line. You may not need to use all the lines. **[4]**

(b) Describe what happens when line 08 is run. **[2]**

3 **Total: 4 marks**

A self-service machine in a library shows the same menu several times while it is being used. The menu is these three lines:

```
1 Borrow
2 Return
3 Renew
```

(a) Write a procedure called `showMenu()` that outputs the three lines of the menu, in order. **[3]**
You must use either:

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

(b) Write the line of code that calls the procedure `showMenu()`. **[1]**

4 **Total: 8 marks**

The entry gate at a gym is opened by a member card or by a code. The program that controls the gate contains three lines of code that are repeated.

```
01 method = input("Enter CARD or CODE")
02 if method == "CARD" then
03     print("Card accepted")
04     print("Gate opening")
05     print("Please walk through")
06     print("Gate closing")
07 elseif method == "CODE" then
08     print("Code accepted")
09     print("Gate opening")
10     print("Please walk through")
11     print("Gate closing")
12 else
13     print("Please ask at reception")
14 endif
```

(a) Rewrite the program so that the three repeated lines are in a procedure called `openGate()`, and the procedure is called wherever those lines are needed. The program must still produce the same output for every input. **[4]**

You must use either:

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

(b) Explain **two** benefits of using the procedure `openGate()` in this program. **[4]**

1

2

Total for this worksheet: 22 marks

MODULE M PROGRAMMING · M23

Functions and return values

Original practice questions · 22 marks · ~25 minutes · OCR J277 spec 2.2.3

Name _____ Date _____

Instructions. Answer all questions in the spaces provided. The marks for each part are shown in brackets [ ]. The total for this worksheet is **22 marks**. Code is written in OCR Exam Reference Language. All data on this sheet is made up.

1

Total: 3 marks

(a) Tick (✓) **one** box in each row to show whether each statement describes a function, a procedure, or both. **[3]**

Statement	Function	Procedure	Both
It is a named sub program that is defined once and can be called many times.			
It uses `return` to send a value back to the point where it was called.			
It carries out a task but does not return a value.			

2

Total: 4 marks

A print shop charges 5p for every page it prints. Its ordering program uses this function to work out the cost, in pence, of a print job.

```
01 function printCost(pages, copies)
02     totalPages = pages * copies
03     cost = totalPages * 5
04     return cost
05 endfunction
```

(a) Give the line number of the statement that sends a value back to the point where the function was called. **[1]**

(b) State the value returned by `printCost(3, 4)` **[1]**

(c) A customer orders two print jobs: **[2]**

- a 2-page leaflet, 5 copies
- a 4-page booklet, 3 copies.

Write **one** statement that uses the function `printCost()` to store the total cost of both jobs in the variable `orderTotal`

3

Total: 5 marks

A warehouse packs items into boxes. Each box holds up to 8 items. This function works out how many boxes an order needs.

```
01 function boxesNeeded(items)
02     boxes = items DIV 8
03     if items MOD 8 != 0 then
04         boxes = boxes + 1
05     endif
06     return boxes
07 endfunction
```

(a) Complete the table to show the value returned by each function call. **[3]**

Function call	Value returned
boxesNeeded(16)	
boxesNeeded(17)	
boxesNeeded(5)	

The main program uses the function like this:

```
10 orders = int(input("Enter the number of items"))
11 if boxesNeeded(orders) > 2 then
12     print("Use a trolley")
13 else
14     print("Carry by hand")
15 endif
```

(b) State what the program outputs when the user enters: **[2]**

20 .....

16 .....

4 **Total: 4 marks**

A water-quality sensor takes three readings. The program below should:

- use the function `meanReading()` to **return** the mean of three readings
- store the mean of the readings 4, 9 and 8 in the variable `result`, then output it.

The program contains **two** errors.

```
01 function meanReading(first, second, third)
02     total = first + second + third
03     mean = total / 3
04     print(mean)
05 endfunction
06 meanReading(4, 9, 8)
07 print("Mean reading is " + str(result))
```

(a) Identify the line number of each error and write the corrected line of code. **[4]**

Error 1 Line number ..... Corrected line

Error 2 Line number ..... Corrected line

5 **Total: 6 marks**

A hotel heating system checks whether a room is at a comfortable temperature. A temperature, in °C, is comfortable if it is from 18 to 24 inclusive. Temperatures can include decimals, such as 21.5

(a) Write a program that: **[6]**

- defines a function called `isComfortable()` that takes a temperature as a parameter and returns `True` if the temperature is comfortable, or `False` if it is not
- asks the user to enter a room temperature and stores it
- calls `isComfortable()` with the temperature entered
- outputs `Comfortable` if the function returns `True`, otherwise outputs `Adjust heating`.

You must use either:

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

Total for this worksheet: 22 marks

MODULE M PROGRAMMING · M24

Parameters and arguments

Original practice questions · 20 marks · ~25 minutes · OCR J277 spec 2.2.3

Name _____ Date _____

Instructions. Answer all questions in the spaces provided. The marks for each part are shown in brackets [ ]. The total for this worksheet is **20 marks**. Code is written in OCR Exam Reference Language. All data on this sheet is made up.

1

Total: 4 marks

A café till uses this program to add a service charge to a bill.

```
01 function addService(bill, percent)
02     extra = bill * percent / 100
03     return bill + extra
04 endfunction
05
06 total = addService(40, 10)
07 print(total)
```

(a) Identify the **two** parameters of the function `addService`. **[1]**

(b) Identify the line number where the function is called. **[1]**

(c) State the value that the parameter `percent` holds when the function is called. **[1]**

(d) State the output when the program is run. **[1]**

2

Total: 3 marks

(a) Tick (✓) **one** box in each row to show whether each statement about parameters is true or false. **[3]**

Statement	True	False
Inside a sub program, a parameter can be used in the same way as a variable.		
A sub program can only ever have one parameter.		
When a sub program is called, the values in the brackets are matched to the parameters in the order they are written.		

3

Total: 5 marks

A hardware warehouse uses this procedure to check its stock.

```
01 procedure stockMessage(item, quantity, minimum)
02     if quantity < minimum then
03         print(item + " low: order " + str(minimum - quantity))
04     else
05         print(item + " OK")
06     endif
07 endprocedure
08
09 stockMessage("bolts", 12, 20)
10 stockMessage("nuts", 30, 25)
11 stockMessage("washers", 15, 15)
```

(a) Complete the table to show what is output by each of the procedure calls on lines 09, 10 and 11. **[3]**

Call on line	Output
09	
10	
11	

(b) A programmer adds this line to the program:

```
12 stockMessage(20, "bolts", 12)
```

Explain why line 12 will **not** work as the programmer intended. **[2]**

4

Total: 2 marks

A hotel needs a program that prints a door sign for each of its 40 rooms. The programmer writes **one** procedure, `printSign(roomNumber)`, and calls it 40 times with a different room number each time.

(a) Give **two** benefits of using one procedure with a parameter, instead of writing separate code to print each room's sign. **[2]**

1

2

5 **Total: 6 marks**

A weather station stores a week of daily maximum temperatures, as whole numbers, in an array. It needs a function that counts how many values in an array are above a limit.

(a) Complete the function `countAbove` so that it: **[4]**

- takes an array of whole numbers and a limit as its two parameters
- counts how many values in the array are **greater than** the limit
- returns the count.

You must use either:

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

```
function countAbove(readings, limit)
```

```
endfunction
```

(b) The temperatures are stored in an array called `dayTemps`. Write **one** line of code that calls `countAbove` to count how many of the temperatures are above 30, and stores the result in a variable called `hotDays`. **[2]**

Total for this worksheet: 20 marks

MODULE M PROGRAMMING · M25

Local and global variables

Original practice questions · 22 marks · ~25 minutes · OCR J277 spec 2.2.3

Name _____ Date _____

Instructions. Answer all questions in the spaces provided. The marks for each part are shown in brackets [ ]. The total for this worksheet is **22 marks**. Code is written in OCR Exam Reference Language. All data on this sheet is made up.

1

Total: 3 marks

(a) Tick (✓) **one** box in each row to show whether each statement describes a local variable or a global variable. **[3]**

Statement	Local variable	Global variable
It can be used in the main program and inside every sub program.		
It is declared inside a procedure or function.		
It only exists while the sub program it belongs to is running.		

2 **Total: 9 marks**

A pay machine in a multi-storey car park charges for each stay and keeps a running total of the money taken.

```
01 global totalTakings = 0
02
03 procedure takePayment(hours)
04     const RATE = 2
05     fee = hours * RATE
06     totalTakings = totalTakings + fee
07     print("Fee: " + str(fee))
08 endprocedure
09
10 takePayment(3)
11 takePayment(5)
12 print("Total: " + str(totalTakings))
```

(a) Identify the global variable in the program. **[1]**

(b) Identify the local constant in the program. **[1]**

(c) State the **three** lines of output when the program is run. **[3]**

1

2

3

(d) A programmer adds this line to the end of the program:

```
13 print(fee)
```

Explain why line 13 causes an error. **[2]**

(e) Explain why `totalTakings` needs to be a global variable in this program. **[2]**

3 **Total: 4 marks**

A team of programmers is writing a large booking system for a theatre. Each programmer writes their own procedures and functions.

(a) Explain **two** benefits of the team using local variables, rather than global variables, inside their sub programs. **[4]**

1

2

4 **Total: 6 marks**

A café gives customers loyalty stamps. The main program already contains this line:

```
global totalStamps = 0
```

(a) Write a procedure called `addStamps` that: **[5]**

- takes the amount a customer spends, in whole pounds, as a parameter
- calculates the number of stamps earned, which is one stamp for every full £5 spent, and stores it in a **local** variable
- adds the stamps earned to the global variable `totalStamps`
- outputs the number of stamps earned.

You must use either:

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

(b) `totalStamps` starts at 0. The procedure is called twice: `addStamps(23)` and then `addStamps(12)`. State the value of `totalStamps` after both calls. **[1]**

Total for this worksheet: 22 marks

MODULE M PROGRAMMING · M26

Random numbers

Original practice questions · 20 marks · ~25 minutes · OCR J277 spec 2.2.3 (random number generation)

Name _____ Date _____

Instructions. Answer all questions in the spaces provided. The marks for each part are shown in brackets [ ]. The total for this worksheet is **20 marks**. Code is written in OCR Exam Reference Language, where `random(a, b)` gives a random number from a to b inclusive. No calculator.

1

Total: 4 marks

A cinema booking program and a weather simulation each use one of the statements below.

(a) Complete the table. For each statement, give the data type of the value stored, and the smallest and largest value that could be stored. **[4]**

Statement	Data type	Smallest value	Largest value
`seat = random(1, 40)`			
`wind = random(-3.0, 3.0)`			

2

Total: 2 marks

A building's door-entry system creates a temporary code for each delivery driver. Its lights then flash after a short random pause.

(a) Write one statement that stores a random whole number from 100 to 999 in the variable `code`. **[1]**

(b) Write one statement that stores a random real number from 0.5 to 2.5 in the variable `pause`. **[1]**

3

Total: 3 marks

A museum's welcome screen shows visitors one fact chosen at random. The array `facts` holds 6 facts, stored at indexes 0 to 5. The programmer has written this code.

```
01 pick = random(1, 6)
02 print("Did you know? " + facts[pick])
```

(a) Line 01 contains a logic error. Describe **two** problems this error causes when the program is run. **[2]**

(b) Rewrite line 01 so that any of the 6 facts can be shown. **[1]**

4

Total: 5 marks

A gardening simulation models how a seedling grows. Each day it grows by a random whole number of centimetres. The program stops when the seedling is at least 15 cm tall.

```
01 height = 5
02 days = 0
03 do
04     growth = random(1, 4)
05     height = height + growth
06     days = days + 1
07 until height >= 15
08 print("Ready after " + str(days) + " days")
```

(a) When the program is run, `random(1, 4)` returns **3**, then **4**, then **1**, then **2**. **[4]**
Complete the trace table. The first row has been done for you. You may not need to use all the rows.

growth	height	days	Output
	5	0	

(b) State the **smallest** number of times the loop could run on any run of the program. **[1]**

5 **Total: 6 marks**

A coffee shop runs a reward game at the till. Each customer is given a random whole number from 1 to 20. A customer who is given 20 wins a free drink.

(a) Write a program that: **[6]**

- asks the user to enter the number of customers
- for each customer, generates a random whole number from 1 to 20
- outputs "Free drink" if the number is 20, otherwise outputs the number
- after all the customers, outputs how many free drinks were won.

You must use either OCR Exam Reference Language or a high-level programming language that you have studied.

Total for this worksheet: 20 marks

MODULE M PROGRAMMING · M27

Records

Original practice questions · 20 marks · ~25 minutes · OCR J277 spec 2.2.3, 2.2.2

Name _____ Date _____

Instructions. Answer all questions in the spaces provided. The marks for each part are shown in brackets [ ]. The total for this worksheet is **20 marks**. Code on this sheet is written in OCR Exam Reference Language. OCR Exam Reference Language has no keyword for records, so in Question 3 a field is written as `recordName.fieldName`.

A tool-hire shop lends out tools such as hedge trimmers and floor sanders by the day. Its program stores the details of each tool as a **record**.

1

Total: 5 marks

(a) Describe what is meant by a **record**. **[2]**

(b) State what is meant by a **field**. **[1]**

(c) Give **two** differences between how data is stored in a record and how data is stored in a 1D array. **[2]**

2 **Total: 4 marks**

One tool record is shown. Complete the table to give the most appropriate data type for each field. The first row has been done for you. **[4]**

Field	Example data	Data type
`toolID`	TH204	String
`description`	Pressure washer	
`dailyRate`	24.00	
`minDays`	2	
`available`	True	

(`dailyRate` is the cost in pounds to hire the tool for one day. `minDays` is the smallest number of whole days the tool can be hired for.)

3 **Total: 4 marks**

This program stores one tool in a record called `hireTool` and decides whether a customer can hire it.

```
01 hireTool.toolID = "TH204"
02 hireTool.description = "Pressure washer"
03 hireTool.dailyRate = 24.00
04 hireTool.minDays = 2
05 hireTool.available = True
06 days = 3
07 if hireTool.available == True AND days >= hireTool.minDays then
08     print(hireTool.description)
09     print("Cost: " + str(hireTool.dailyRate * days))
10 else
11     print("Cannot hire")
12 endif
```

(a) State the output of the program. **[2]**

(b) Line 06 is changed to `days = 1`. State the output of the program. **[1]**

(c) The shop puts up the price of this tool. Write **one** statement that adds 1.50 to the daily rate stored in `hireTool`. **[1]**

4 **Total: 7 marks**

The shop stores the records of five tools in a 2D array called `tools`, like a table. Each row is one record and each column is one field. An array can only store one data type, so every value is stored as a string.

`tools[row, column]` gives the value in that row and column. Both indexes start at 0.

	Column 0 `toolID`	Column 1 `description`	Column 2 `dailyRate`	Column 3 `minDays`	Column 4 `available`
Row 0	`"TH118"`	`"Hedge trimmer"`	`"12.50"`	`"1"`	`"False"`
Row 1	`"TH204"`	`"Pressure washer"`	`"24.00"`	`"2"`	`"True"`
Row 2	`"TH311"`	`"Tile cutter"`	`"18.00"`	`"1"`	`"True"`
Row 3	`"TH427"`	`"Leaf blower"`	`"9.75"`	`"1"`	`"True"`
Row 4	`"TH509"`	`"Floor sander"`	`"31.50"`	`"2"`	`"False"`

(a) State the value stored in `tools[2, 1]`. **[1]**

(b) Write down all of the data in the record for the tool with the ID `TH427`. **[1]**

(c) The hedge trimmer (`TH118`) has been returned. Write **one** statement that updates `tools` to show that it is now available. **[1]**

(d) Use the contents of `tools` shown in the table (ignore the change in part (c)). **[4]**
Write an algorithm that outputs the description of every tool that is available **and** has a daily rate of less than 20.00.
You must use either: OCR Exam Reference Language, or a high-level programming language that you have studied.

Total for this worksheet: 20 marks

MODULE M PROGRAMMING · M28

Writing maintainable code: naming, comments, indentation and sub programs

Original practice questions · 20 marks · ~25 minutes · OCR J277 spec 2.3.1, 2.2.3

Name _____ Date _____

Instructions. Answer all questions in the spaces provided. The marks for each part are shown in brackets [ ]. The total for this worksheet is **20 marks**. All code on this sheet is written in OCR Exam Reference Language.

1

Total: 4 marks

A programmer is told to make a program more **maintainable**.

(a) Tick (✓) **two** boxes to show which actions would make the program more maintainable. **[2]**

Action	Tick (✓)
Adding comments that explain what each section of code does	
Checking that every number input is within a sensible range	
Asking the user to enter a username and password	
Indenting the statements inside each loop and `if` statement	
Running the program with boundary test data	

(b) Give **two** reasons why comments are useful in a program. **[2]**

2 **Total: 7 marks**

A water company records the rainfall at a reservoir each day for a week. A day with more than 25 mm of rain is a heavy-rain day. A programmer has written this program, which works correctly.

```
01 t = 0
02 h = 0
03 for d = 1 to 7
04 r = float(input("Enter rainfall in mm"))
05 t = t + r
06 if r > 25 then
07 h = h + 1
08 endif
09 next d
10 print("Total rainfall: " + str(t))
11 print("Heavy-rain days: " + str(h))
```

(a) Explain how indenting lines 04 to 08 would make this program easier to maintain. **[2]**

(b) Rewrite the program so that it is easier to maintain. Your program must: **[5]**

- do exactly the same job as the program above
- use meaningful identifiers that follow a naming convention
- use indentation
- include at least one suitable comment.

3 **Total: 7 marks**

A museum sells adult tickets at £12.50 and senior tickets at £9.00. If the cost of one type of ticket comes to more than £100, 10% is taken off that cost. This is part of the program used at the ticket desk.

```
01 adults = int(input("Number of adult tickets"))
02 adultCost = adults * 12.50
03 if adultCost > 100 then
04     adultCost = adultCost * 0.9
05 endif
06 seniors = int(input("Number of senior tickets"))
07 seniorCost = seniors * 9.00
08 if seniorCost > 100 then
09     seniorCost = seniorCost * 0.9
10 endif
11 print("Total to pay: " + str(adultCost + seniorCost))
```

Lines 03 to 05 and lines 08 to 10 do the same job. The programmer decides to replace them with a function called `applyDiscount()`.

(a) Complete the function `applyDiscount()`. The function: **[3]**

- takes a cost as a parameter
- takes 10% off the cost if the cost is more than £100
- returns the cost.

You must use either: OCR Exam Reference Language, or a high-level programming language that you have studied.

```
function applyDiscount(cost)

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

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

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

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

endfunction
```

(b) Write the statement that should replace lines 03 to 05 so that the program uses the function. **[1]**

(c) Explain how using the function `applyDiscount()` makes this program easier to maintain. **[3]**

4 **Total: 2 marks**

A software company has a team of programmers. Every programmer must follow the same naming convention for identifiers, for example `camelCase` for all variables, such as `ticketsSold`.

(a) Explain why using the same naming convention helps the team to maintain its programs. **[2]**

Total for this worksheet: 20 marks