# Cambridge IGCSE

| CANDIDATE NAME | |
| --- | --- |

| CENTRE NUMBER | | | | | | CANDIDATE NUMBER | | | | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |

**COMPUTER SCIENCE** **0478/22**

Paper 2  Problem-solving and Programming **October/November 2019**

**1 hour 45 minutes**

Candidates answer on the Question Paper.

No Additional Materials are required.

No calculators allowed.

## READ THESE INSTRUCTIONS FIRST

Write your centre number, candidate number and name in the spaces at the top of this page.
Write in dark blue or black pen.
You may use an HB pencil for any diagrams, graphs or rough working.
Do not use staples, paper clips, glue or correction fluid.
DO **NOT** WRITE IN ANY BARCODES.

Answer **all** questions.

**DO NOT ATTEMPT TASKS 1, 2 AND 3** in the pre-release material; these are for information only.

You are advised to spend no more than **40 minutes** on **Section A** (Question 1).

No marks will be awarded for using brand names of software packages or hardware.

Any businesses described in this paper are entirely fictitious.

At the end of the examination, fasten all your work securely together.

The number of marks is given in brackets [ ] at the end of each question or part question.

The maximum number of marks is 50.

This document consists of **13** printed pages and **3** blank pages.

Cambridge Assessment
International Education

**[Turn over**

## Section A

**You are advised to spend no longer than 40 minutes answering this section.**

**Here is a copy of the pre-release material.**

**DO NOT** attempt Tasks 1, 2 and 3 now.

Use the pre-release material and your experience from attempting the tasks before the examination to answer Question 1.

**Pre-release material**

A company supplies concrete slabs for paving. The slabs are made to order in batches of 20; all slabs in a batch are identical. Customers can order from a small range of standard sizes and colours. All measurements are given in millimetres. The price is calculated at $0.05 for a volume of $100\,000\,mm^3$ of grey concrete; red and green concrete are charged at 10% more. Customers can choose their own colours; a custom colour has an initial set up cost of $5 then 15% more than the price for grey.

| Colour of slab |
|---|
| Grey |
| Red |
| Green |
| Custom |

| Depth of slab |
|---|
| 38 |
| 45 |

| Shapes | Sizes for each shape |
|---|---|
| Square | 600 × 600 or 450 × 450 |
| Rectangular | 600 × 700 or 600 × 450 |
| Round | Diameter 300 or 450 |

Write and test a program or programs for the concrete slab company.

* Your program or programs must include appropriate prompts for the entry of data; data must be validated on entry.
* Error messages and other output need to be set out clearly and understandably.
* All variables, constants and other identifiers must have meaningful names.

You will need to complete these **three** tasks. Each task must be fully tested.

**Task 1** – Price for a batch of 20 slabs

Using the information above set up a routine that allows a customer to choose the concrete slab they require and calculate a price for a batch of 20 slabs. The details of the slab chosen and the price for a batch of 20 should be displayed on the screen.

**Task 2** – Customer places an order

Using the information from TASK 1, the customer places an order for the number of slabs they require. Orders for fewer than 20 slabs or more than 100 slabs are not accepted; orders that are not a multiple of 20 are rounded up to the next multiple of 20 slabs. Display the order price and the number of slabs to be produced.

**Task 3** – Flexible pricing

The cost of concrete is variable. The cost for $100\,000\,mm^3$ of grey concrete can be input and two grades are available; basic at the cost input and best at 7% more. Use a copy of your program for TASK 1 to develop TASK 3 to input the cost and grade of concrete before calculating the price for 20 slabs.

**1** **(a)** All variables, constants and other identifiers must have meaningful names.

**(i)** Describe the data structures that you have used to store the data for the concrete slabs in **Task 1**, include the name, data type and use for each data structure.

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

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

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

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

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

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

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

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

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

.................................................................................................................... [5]

**(ii)** State **one** variable that you have used in **Task 3**.
Give the data type for the variable. State what it is used for.

Variable name .................................................................................................................

Data type .......................................................................................................................

Use .................................................................................................................................

.......................................................................................................................................
[3]

**(b)** Explain how you calculated the volume of a concrete slab in **Task 1**.

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

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

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

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

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

.................................................................................................................... [3]

**(c)** Write an algorithm for **Task 2**, using **either** pseudocode, programming statements **or** a flowchart. Assume that **Task 1** has been completed.

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

.......................................................................................................................................... [5]

**(d)** Explain how you changed your program for **Task 1** to meet the requirements for **Task 3**. Include and fully explain any altered or additional programming statements.

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

...................................................................................................................................... [4]

**Section B**

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]

**3** **(a)** The flowchart checks the lengths of a batch of 10 ropes. For the batch to be accepted 90% of the lengths need to be between 24.9 and 25.1 metres.

```
                              ┌─────────────┐
                              │    START     │
                              └─────────────┘
                                     │
                                     ▼
                         ┌─────────────────────┐
                         │   Reject  ←  0       │
                         │   Count   ←  0       │
                         └─────────────────────┘
                                     │
                                     ▼
                              ╱─────────────╲            Yes
                            ╱   Is Count =    ╲──────────────────┐
                            ╲     10 ?        ╱                   │
                              ╲─────────────╱                    │
                                     │ No                        ▼
                                     ▼                    ╱─────────────╲      No
                         ┌─────────────────────┐       ╱   Is Reject     ╲──────────┐
                         │   INPUT Length       │       ╲    <= 1 ?       ╱          │
                         └─────────────────────┘         ╲─────────────╱            │
                                     │                          │ Yes               ▼
                                     ▼                          │          ┌────────────────┐
                              ╱─────────────╲                   │          │ OUTPUT "Batch  │
                   Yes      ╱   Is Length <    ╲                │          │   rejected"    │
           ┌──────────────  ╲    25.1 AND      ╱                │          └────────────────┘
           │                ╲  Length > 24.9 ? ╱                │                   │
           │                  ╲─────────────╱                   │                   │
           │                         │ No                       ▼                   │
           │                         ▼                 ┌────────────────┐           │
           │              ┌─────────────────────┐      │ OUTPUT "Batch  │           │
           │              │  Reject ← Reject + 1 │      │   accepted"    │           │
           │              └─────────────────────┘      └────────────────┘           │
           │                         │                          │                   │
  ┌─────────────────────┐            │                          ▼                   │
  │ Count ← Count + 1    │◄───────────┘                   ┌─────────────┐           │
  └─────────────────────┘                                │     END      │◄──────────┘
           │                                              └─────────────┘
           └──────────────────────────────────────────►
```
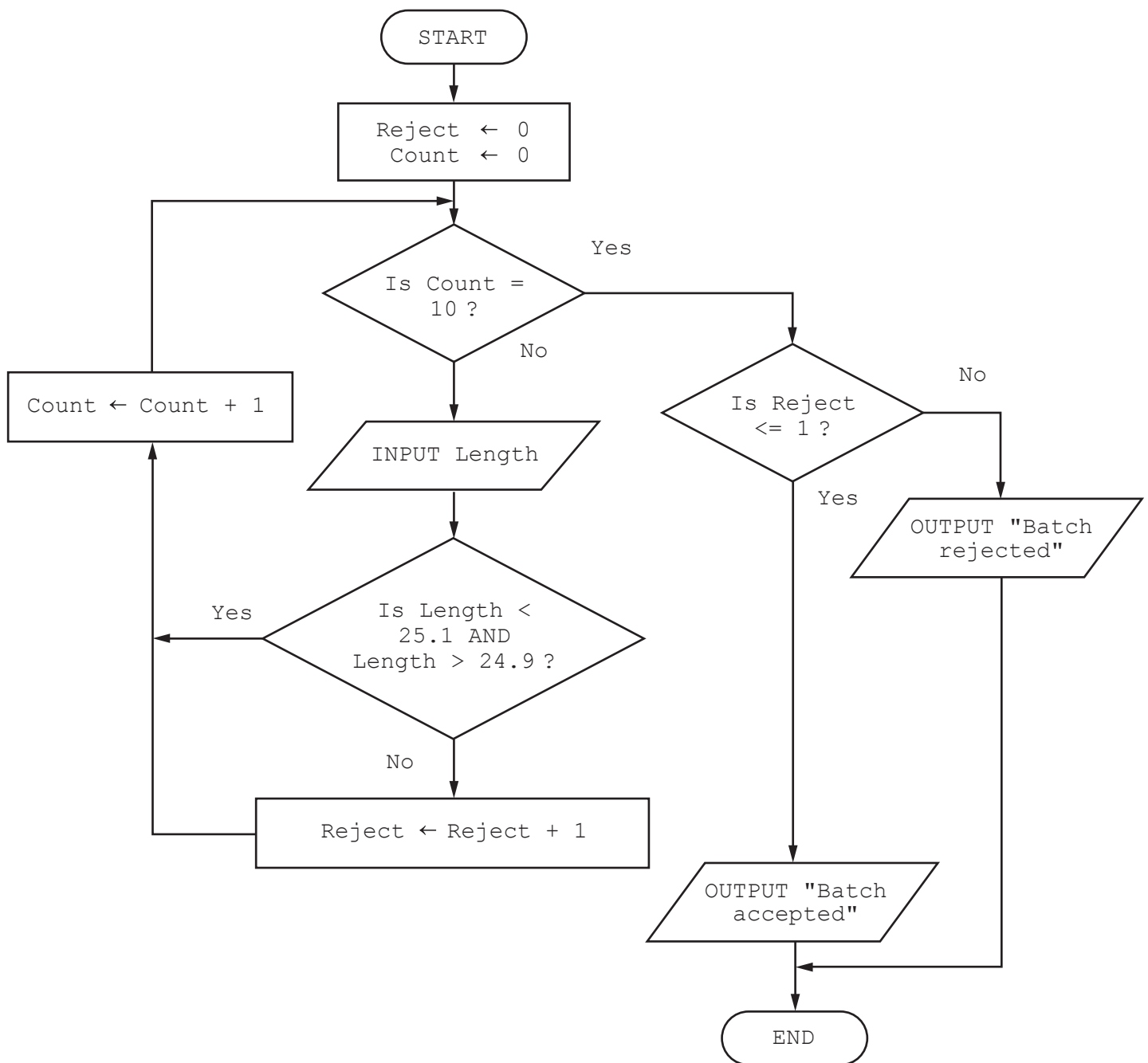
Complete the trace table for the input data:

24.88, 25.01, 24.98, 25.00, 25.05, 24.99, 24.97, 25.04, 25.19, 25.07

| Reject | Count | Length | OUTPUT |
|--------|-------|--------|--------|
|        |       |        |        |
|        |       |        |        |
|        |       |        |        |
|        |       |        |        |
|        |       |        |        |
|        |       |        |        |
|        |       |        |        |
|        |       |        |        |
|        |       |        |        |
|        |       |        |        |
|        |       |        |        |

[4]

**(b) (i)** It has been decided to only reject batches of rope that contain ropes that are too short.

State the change required to the algorithm.

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

.......................................................................................................................................... [1]

**(ii)** Explain how the algorithm to reject batches could be improved to make it more effective.

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

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

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

.......................................................................................................................................... [2]

**4** **Four** validation checks and **four** descriptions are shown.

Draw a line to connect each validation check to the correct description.

| **Validation Check** | **Description** |
| --- | --- |
| Range check | Checks that some data is entered. |
| Presence check | Checks for a maximum number of characters in the data entered. |
| Length check | Checks that the characters entered are all numbers. |
| Type check | Checks that the value entered is between an upper value and a lower value. |

[3]

**5** A programmer writes a program to weigh baskets of fruit in grams, keeping a total of the weight and counting the number of baskets. The total weight is stored in a variable `Total` and the number of baskets is stored in a variable `BasketCount`.

Explain, including examples of programming statements, how totalling and counting could be used in this program.

Totalling ...................................................................................................................................................

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

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

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

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

Counting ................................................................................................................................................

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

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

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

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

[4]

6   Explain why constants, variables and arrays are used in programming.

Constants ..................................................................................................................................................

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

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

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

Variables ..................................................................................................................................................

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

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

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

Arrays ..................................................................................................................................................

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

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

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

[6]

**7** A database table, SALES, is used to keep a record of items made and sold by a furniture maker.

| Item number | Order number | Notes | Amount | Status |
|---|---|---|---|---|
| CH001 | 1921 | Smith – six dining chairs | 6 | Delivered |
| TB003 | 1921 | Smith – large table | 1 | In progress |
| CH001 | 1924 | Hue – extra chairs | 4 | In progress |
| CH003 | 1925 | For stock | 2 | Cancelled |
| BN001 | 1927 | Patel – replacement bench | 1 | Not started |
| ST002 | 1931 | Sola – small table | 1 | Delivered |
| CH003 | 1927 | Patel – eight dining chairs with arms | 8 | Not started |
| TB003 | 1927 | Patel – large table | 1 | Not started |

**(a)** Explain why the field **Item number** could not be used as a primary key.

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

................................................................................................................................... [1]

**(b)** A query-by-example has been written to display only the order number and item numbers of any items in progress or not started.

| Field: | Item number | Order number | Amount | Status |
|---|---|---|---|---|
| Table: | SALES | SALES | SALES | SALES |
| Sort: | | | | |
| Show: | ☐ | ✓ | ✓ | ☐ |
| Criteria: | | | | Not Like "Delivered" |
| or: | | | | |

Explain why the query-by-example is incorrect, and write a correct query-by-example.

Explanation ........................................................................................................................

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

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

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

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

| Field: | | | | |
|---|---|---|---|---|
| Table: | | | | |
| Sort: | | | | |
| Show: | ☐ | ☐ | ☐ | ☐ |
| Criteria: | | | | |
| or: | | | | |

[5]

**BLANK PAGE**

15

**BLANK PAGE**

# BLANK PAGE

**COMPUTER SCIENCE**         **0478/22**

Paper 2         **October/November 2019**

MARK SCHEME

Maximum Mark: 50

---

**Published**

---

This mark scheme is published as an aid to teachers and candidates, to indicate the requirements of the examination. It shows the basis on which Examiners were instructed to award marks. It does not indicate the details of the discussions that took place at an Examiners' meeting before marking began, which would have considered the acceptability of alternative answers.

Mark schemes should be read in conjunction with the question paper and the Principal Examiner Report for Teachers.

Cambridge International will not enter into discussions about these mark schemes.

Cambridge International is publishing the mark schemes for the October/November 2019 series for most Cambridge IGCSE™, Cambridge International A and AS Level components and some Cambridge O Level components.

This document consists of **7** printed pages.

## Generic Marking Principles

These general marking principles must be applied by all examiners when marking candidate answers. They should be applied alongside the specific content of the mark scheme or generic level descriptors for a question. Each question paper and mark scheme will also comply with these marking principles.

| |
|---|
| GENERIC MARKING PRINCIPLE 1:<br><br>Marks must be awarded in line with:<br><br>• the specific content of the mark scheme or the generic level descriptors for the question<br>• the specific skills defined in the mark scheme or in the generic level descriptors for the question<br>• the standard of response required by a candidate as exemplified by the standardisation scripts. |
| GENERIC MARKING PRINCIPLE 2:<br><br>Marks awarded are always **whole marks** (not half marks, or other fractions). |
| GENERIC MARKING PRINCIPLE 3:<br><br>Marks must be awarded **positively**:<br><br>• marks are awarded for correct/valid answers, as defined in the mark scheme. However, credit is given for valid answers which go beyond the scope of the syllabus and mark scheme, referring to your Team Leader as appropriate<br>• marks are awarded when candidates clearly demonstrate what they know and can do<br>• marks are not deducted for errors<br>• marks are not deducted for omissions<br>• answers should only be judged on the quality of spelling, punctuation and grammar when these features are specifically assessed by the question as indicated by the mark scheme. The meaning, however, should be unambiguous. |
| GENERIC MARKING PRINCIPLE 4:<br><br>Rules must be applied consistently e.g. in situations where candidates have not followed instructions or in the application of generic level descriptors. |
| GENERIC MARKING PRINCIPLE 5:<br><br>Marks should be awarded using the full range of marks defined in the mark scheme for the question (however; the use of the full mark range may be limited according to the quality of the candidate responses seen). |
| GENERIC MARKING PRINCIPLE 6:<br><br>Marks awarded are based solely on the requirements as defined in the mark scheme. Marks should not be awarded with grade thresholds or grade descriptors in mind. |

**Section A**

| Question | Answer | Marks |
|---|---|---|
| 1(a)(i) | Data Structure(s) maximum two marks<br>•   Array<br>•   Variable<br>•   Constant<br><br>Description maximum three marks<br>•   Name(s) one or more e.g. Colour<br>•   Data type(s) one or more e.g. String<br>•   Use(s) one or more e.g. To store the colour of the slab<br><br>Additional data structure description using the same data structure type maximum one mark<br>•   Two or more full descriptions of the data structure including name, data type and use | **5** |
| 1(a)(ii) | •   Variable name e.g. ConcretePrice<br>•   Data type e.g. Real<br>•   Use e.g. To store the price of the grade of concrete | **3** |
| 1(b) | **Three** from:<br>•   Using the shape, size and depth …<br>•   … the area of the shape is found …<br>•   … for all except round multiply length by breadth/show example …<br>•   … for round multiply Pi by half the diameter/radius squared …<br>•   … multiply the area by the depth to give the volume | **3** |

| Question | Answer | Marks |
|---|---|---|
| 1(c) | **Five** from:<br>MP1    Enter number of slabs to purchase with prompt<br>MP2    Check if less than 20 or greater than 100 …<br>MP3    … if so, reject the number and re-enter a value<br>MP4    Attempt to round the number of slabs to the next 20 …<br>MP5    Accurate attempt to round <u>up to the next 20</u><br>MP6    Calculate the number of batches<br>MP7    Calculate price to pay (number of batches × price per batch from Task 1)<br>MP8    Display price and number of slabs produced …<br>MP9    … Suitable output message(s)<br><br>Sample answer<br>`REPEAT`<br>`    PRINT "Enter Number of slabs to purchase"`<br>`    INPUT Number`<br>`UNTIL Number >= 20 and Number <= 100`<br>`PurchaseNo ← Number`<br>`IF Number > 20 AND Number <= 40 THEN PurchaseNo ← 40 ENDIF`<br>`IF Number > 40 AND Number <= 60 THEN PurchaseNo ← 60 ENDIF`<br>`IF Number > 60 AND Number <= 80 THEN PurchaseNo ← 80 ENDIF`<br>`IF Number > 80 THEN PurchaseNo ← 100 ENDIF`<br>`BatchNo ← PurchaseNo / 20`<br>`PriceToPay ← BatchNo * BatchPrice //BatchPrice calculated in Task 1`<br>`PRINT "Price for ", PurchaseNo, " Slabs is $", PriceToPay` | 5 |
| 1 (d) | **Four** from explanations:<br>•   Enter the price of the concrete<br>•   Store the price of the concrete<br>•   Set up variable(s) for grades<br>•   Input the grade<br>•   The price that is input is used in the final price calculation instead of 0.05<br>•   The price calculation depends on the grade input e.g. 1.07 × price or price + price × 0.07 for Best<br><br>If no programming code seen to support at least one explanation maximum of three marks can be awarded. | 4 |

**Section B**

| Question | Answer | Marks |
|---|---|---|
| 2 | **One** mark for each error identified and suggested correction:<br>• `IF TotalTry > Number` should be `IF Guess > Number`<br>• `IF Guess > Number` should be `IF Guess < Number`<br>• `TotalTry ← Guess + 1` should be `TotalTry ← TotalTry + 1`<br>• `UNTIL Guess <> Number` should be `UNTIL Guess = Number` | **4** |

| Question | Answer | | | | Marks |
|---|---|---|---|---|---|
| 3(a) | | | | | **4** |

| Reject | Count | Length | OUTPUT |
|---|---|---|---|
| 0 | 0 | | |
| 0 | 1 | 24.88 | |
| 1 | 2 | 25.01 | |
| 1 | 3 | 24.98 | |
| 1 | 4 | 25.00 | |
| 1 | 5 | 25.05 | |
| 1 | 6 | 24.99 | |
| 1 | 7 | 24.97 | |
| 1 | 8 | 25.04 | |
| 1 | 9 | 25.19 | |
| 2 | 10 | 25.07 | Batch rejected |
| | | | |

**One** mark for each correct column **max. 4**

| Question | Answer | Marks |
|---|---|---|
| 3(b)(i) | • Remove `Length < 25.1 AND` | **1** |
| 3(b)(ii) | **Two** from:<br>• Check the reject counter after each incrementation/remove reject check after counter = 10 …<br>• … as soon as Reject = 2 / >1 ...<br>• … reject batch and end | **2** |

| Question | Answer | Marks |
|---|---|---|
| 4 |  One mark for each correct line, up to maximum of three marks. | **3** |

| Question | Answer | Marks |
|---|---|---|
| 5 | Totalling:<br>• Adding the weight of each basket to the total weight as each weight is entered<br>• `Total = Total + Weight`<br><br>Counting:<br>• Adding one to/incrementing the number of baskets as each weight is entered<br>• `BasketCount = BasketCount + 1` | **4** |

| Question | Answer | Marks |
|---|---|---|
| 6 | Constants<br>**Two** from:<br>• The value cannot be changed accidentally …<br>• … during the **execution** of the program<br>• Value only needs to be changed once if circumstances change/during the initialisation process<br><br>Variables<br>**Two** from:<br>• Stores a value that can change …<br>• … during the **execution** of the program<br>• Can use a variable without knowing its value<br><br>Arrays<br>**Two** from:<br>• A list of items of the same data type …<br>• … stored under a single name<br>• To reduce the number of variables used<br>• Any item can be found using an index number to show its place in the list | **6** |

| Question | Answer | Marks |
|---|---|---|
| 7(a) | • Number is repeated/not unique | **1** |
| 7(b) | • Item number not displayed/Amount column not required<br>• Not Like 'Delivered' will also show cancelled items | **5** |

| Field: | Item number | Order number | Status | |
|---|---|---|---|---|
| Table: | SALES | SALES | SALES | |
| Sort: | | | | |
| Show: | ☑ | ☑ | ☐ | ☐ |
| Criteria: | | | Like "Not started" | |
| or: | | | Like "In progress" | |

• Correct Item number column
• Correct Order number column and any additional column not shown
• Correct status column