1. (a) *Good answers should earn marks even if they do not comply with the markscheme.*

New INDEX / pass-by-reference / variable parameters pass changes back to the calling routine / main program ***[1 mark]***.
NUMBERS / pass-by-value parameters do **not** pass any changes back ***[1 mark]***.

(Accept an answer along the lines of INDEX uses less memory space than NUMBERS. This would get ***[2 marks]*** even though it is not fully explained.)

(b) *Award marks as follows:*

POS	NUMBERS[POS] < VALUE	VALUE	INDEX[1]
		5000	
1	true	27	1
2	false	27	1
3	true	15	3
4	true	2	4
5	false	2	4

*Award **[2 marks]** for **each** correct column of VALUE, INDEX [1], and NUMBERS[POS]<VALUE (give **[1 mark]** if only one error, and then correct follow through). (A maximum of **[6 marks]**).*

(c) The two main ways of doing this, are to set values to 5000 once their subscripts have been transferred to INDEX, or to record the fact that a subscript has been used in INDEX by using a separate Boolean array of order 5, and set the corresponding entry to true when a subscript has been used in INDEX:

```
procedure SORT
   declare I integer

   for I <-- 1 upto 5 do
      VALUE <-- 5000

      for POS <-- 1 upto 5 do
         if NUMBERS[POS] < VALUE then
            VALUE <-- NUMBERS[POS]
            INDEX[I] <-- POS
         endif
      endfor

      NUMBERS[INDEX[I]] <-- 5000
   endfor
endprocedure SORT
```

```
procedure SORT
   declare I integer
   declare USED boolean array [1..5]

   for I <-- 1 upto 5 do
      USED [I] <-- false
   endfor

   for I <-- 1 upto 5 do
      VALUE <-- 5000

      for POS <-- 1 upto 5 do

         if notUSED[POS] then
            if NUMBERS[POS] < VALUE then
               VALUE <-- NUMBERS[POS]
               INDEX[I] <-- POS
            endif
         endif
      endfor

      USED [I] <-- true
   endfor

endprocedure SORT
```

Award marks as follows:

[2 marks] for outer loop; (***[1 mark]*** for any loop, ***[1 mark]*** for `1 upto 5`)
[1 mark] for setting `INDEX[I] <-- POS` (*i.e.* change from `INDEX[1]`)
[3 marks] for not re-testing a value once the subscript has been used in `INDEX`:
(*e.g.* setting `NUMBERS[INDEX[I]] <-- 5000`, or setting `USED[I] <-- true` **and** testing `if notUSED[POS] then`)

- ***[1 mark]*** for any attempt at this
- ***[2 marks]*** for a good, but incorrect, attempt
- ***[3 marks]*** for a totally correct solution (NOTE: if the "Boolean" method is used, do not deduct marks for not including the initialisation to `true` or declaring the equivalent of `USED`)

(d)
```
procedure SORTED
   declare POS integer

   for POS <-- 1 upto 5 do
      SORTED[POS] <-- NUMBERS[INDEX[POS]]
   endfor
endprocedure SORTED
```

Award marks as follows:

[1 mark] for correct loop **and** declaration of variable as an integer
[1 mark] for a correct array and subscript for SORTED in the assignment
[2 marks] for value in the assignment (*[1 mark]* for a good, but incorrect, attempt at using INDEX and NUMBERS to get the value). The candidate may have separated the values, which is acceptable, *e.g:*

```
for POS <-- 1 upto 5 do
   SUB <-- INDEX[POS]
   SORTED[POS] <-- NUMBERS[SUB]
endfor
```

(e) Two solutions are:

```
procedure TALLY
   declare POS, COUNT, CURRENT integer

   POS <-- 1
   COUNT <-- 1
   CURRENT <-- ORDERED[1]
   repeat
      while POS < 600 and ORDERED[POS+1] = CURRENT do
         COUNT <-- COUNT+1
         CURRENT <-- ORDERED[POS+1]
         POS <-- POS+1
      endwhile

      output CURRENT, " is stored ", COUNT, " times "

      if POS < 600 then
         CURRENT <-- ORDERED[POS+1]
         COUNT <-- 1
      else
         output ORDERED[600], " is stored once "
      endif
   until POS = 600
endprocedure TALLY
```

Or:

```
procedure TALLY
   declare POS, COUNT, LAST integer

   COUNT <-- 1
   LAST <-- ORDERED[1]
   for POS <-- 2 upto 600 do
      if ORDERED[POS] = LAST then
         COUNT <-- COUNT+1
      else
         output LAST, " is stored ", COUNT, " times "
         LAST <-- ORDERED[POS]
         COUNT <-- 1
      endif
   enddo

   output LAST, " is stored ", COUNT, " times "
endprocedure TALLY
```

Award marks as follows:

[2 marks] for initialisations (***[1 mark]*** for an incomplete list).
[1 mark] for outer loop until 600 locations have been tested
[2 marks] for a correct test of same values (*e.g.* `ORDERED[POS+1] = CURRENT` in the **`while`**, or `ORDERED[POS] = LAST` in the **`if`** *etc.*); (***[1 mark]*** for a reasonable, but incorrect, attempt)
[2 marks] for a **correct** increment of a counter (eg `COUNT`) in the **correct** place (***[1 mark]*** for a reasonable, but incorrect, attempt)
[2 marks] for correctly updating test value (*i.e.* `CURRENT<--ORDERED[POS+1]` in the first algorithm, or `LAST<--ORDERED[POS]` in the second) in the **correct** place (***[1 mark]*** for a reasonable, but wrong, attempt in the correct place, or correct statements in the wrong place; ***[0 marks]*** for a reasonable, but wrong, attempt in a incorrect place)
[2 marks] for the **`output`** (the wording is **not** important, even though it is given in the question, but it **must** contain the equivalent of `CURRENT/LAST` and `COUNT` **and** be in the **correct** position for both marks, deduct a mark for any of these points that are missing (but do not give ***[-1 mark]*** if **all** are missing!))
[1 mark] for a good attempt at the final **`output`** required (i.e. as part of the **`else`** in algorithm 1 for a single value in location 600, or in the final display in algorithm 2)

(a) *Award **[1 mark]** for any feasible sensor.*

e.g. Temperature, moisture, pressure, barometric *etc.*

(b) *Award marks as follows, up to **[2 marks]** max:*

- new supercomputers would have predicted bad storm ***[1 mark]***, whereas less powerful ones didn't ***[1 mark]***.
- newer computers will be able to process equations faster ***[1 mark]***, giving forecasts earlier ***[1 mark]***.
- new computers will be able to process more complex equations (*i.e.* more than 7 variables) ***[1 mark]***, giving more accurate predictions ***[1 mark]***.

(c) *Award **[1 mark]** for defining archive data, and **[2 marks]** for a clear description of its use (**[1 mark]** for a reasonable attempt), up to **[3 marks]** max:*

- Data kept after initial use / for long-term store not required for on-line access ***[1 mark]***

- Used for research / tracking history ***[1 mark]*** to test for patterns ***[1 mark]***

(d) *Award marks as follows:*

(i) - Data collection site ⇒ National Weather Service ***[1 mark]***

(ii) - forecasting based on it ***[1 mark]***, so important for accuracy ***[1 mark]***

(e) *Award **[1 mark]** for a correct identification, and **[2 marks]** for a clear reason, for two reasons, giving a maximum of **[6 marks]**.*

new media will take less space ***[1 mark]***. As archive data continues to increase ***[1 mark]***, it will keep space used to a minimum if more compact media is used ***[1 mark]***

current media will become out-of-data / obsolete ***[1 mark]***. New media is always being developed ***[1 mark]*** and if data is not changed with the new media, it may not be able to be read ***[1 mark]***.

(f) (i) *Award **[1 mark]** for:*

the bad snow storm ***[1 mark]***

(ii) *Award up to **[3 marks]** maximum for a discussion on trusting computers more than people, eg:*

peoples' skills become devalued / not trusted ***[1 mark]*** so less people will have those skills ***[1 mark]***, and since people program computers ***[1 mark]*** forecasts will get worse ***[1 mark]***

(g) *Award **[1 mark]** for a suitable form and up to **[2 marks]** for a clear outline (**[1 mark]** for a partial outline), for two forms of output, up to **[6 marks]** maximum, e.g:*

Text ***[1 mark]***, a forecast in printed form ***[1 mark]***, so that a weather forecaster can read it out ***[1 mark]***.

Graphic ***[1 mark]***, a forecast in pictorial form ***[1 mark]*** so that a copy can be shown on the television as a map so that viewers can see it ***[1 mark]***.

Note: the question does not require the candidate to explicitly state the form (*e.g.* text or graphic), so if the format is clear from the description, allocate this mark as well.

3. (a) *Allow other answers that involve data storage that is input and storage of data that may change.*

To store the data from the sensors ***[1 mark]***

(b) *Award **[1 mark]** for any feasible input device, and **[1 mark]** for a good attempt at a description / reason. Award **[1 mark]** for any feasible output device, and **[1 mark]** for a good attempt at a description / reason. To a maximum of **[4 marks]**. If it is feasible, possible and sensible, and if the description works award the marks even if its not likely e.g:*

keyboard / keypad ***[1 mark]*** to allow entry (code) of destination *[1 mark]*
presence sensor / any feasible sensor ***[1 mark]*** so that it doesn't hit any objects / people ***[1 mark]***

Speaker / sound device ***[1 mark]*** to warn people of approaching buggy ***[1 mark]***
(Remember, no marks for motor as this is given in the question)

(c) *Award **[1 mark]** for identifying a suitable implication, and **[2 marks]** for a good discussion for **[3 marks]** maximum. e.g:*

Loss of work time ***[1 mark]***. Without buggy heavy objects cannot be transported / have to be done by hand ***[1 mark]*** which will slow / halt production ***[1 mark]***.

Possible danger ***[1 mark]***. If any sensor malfunctions / doesn't work ***[1 mark]*** then if there is not an auto-shut off ***[1 mark]*** the buggy may crash into objects / people ***[1 mark]*** (causing injury).

Note: the candidate does not have to identify the implication explicitly. If it is clear from the description, this mark is to be awarded.

(d) *Award up to **[3 marks]** maximum by giving **[1 mark]** per valid point. e.g:*

the new layout must be recorded into the buggy's processor ***[1 mark]***
the ROM must be reprogrammed ***[1 mark]***
by creating a new one ***[1 mark]***

if movement is by sensing (*e.g.* following white line on floor, reading barcodes around the factory *etc.*) ***[1 mark]***
these will have to be relocated ***[1 mark]***

time will be required to do this ***[1 mark]***
therefore must be planned / tested beforehand / or time lost ***[1 mark]***

must be tested thoroughly, otherwise collisions may occur ***[1 mark]***

Do **not** give any marks for a statement along the lines of 'the buggy will crash', unless justified (similar to the final point above).