**Algorithm Modelling Tools : Flowchart , Pseudo-code, Trace Table**

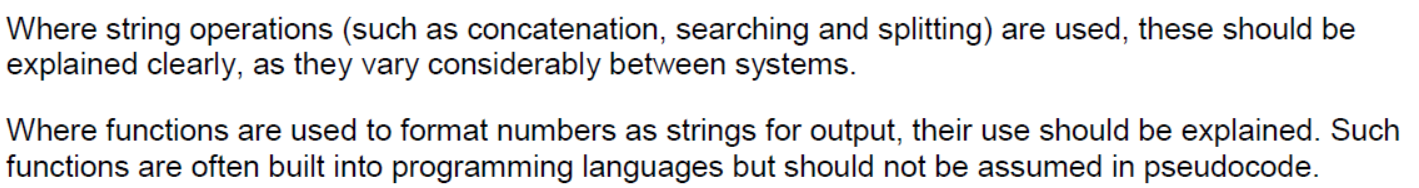
|  |  |  |  |
| --- | --- | --- | --- |
|  | **Structured English** | **Pseudocode** | **Flowchart** |
| **Sequence and Assignment** | SET A TO 100  INCREMENT A BY 1 | DECLARE A: INTEGER  A <- A + 1 |  |
| **Input** | GET INPUT B FOR TEMPERATURE IN CELCIUS | OUTPUT "Enter Temp. in Cels"  INPUT B |  |
| **Output** | OUTPUT "TEMPERATURE IN CELSIUS IS ", B | OUTPUT "Temp in Cels =", B |  |
| **Selection** | IF A IS GREATER THAN B  THEN EXECUTE FUNCTION Factorial(A)  ELSE EXECUTE FUNCTION SqRoot(B) | DECLARE A, B : INTEGER  INPUT A, B  IF A > B THEN  factorial(A)  ELSE IF A = B THEN  SqRoot(B)  ELSE  SqRoot(A)  ENDIF |  |
| **Iteration** | WHILE A IS GREATER THAN B INCREMENT B by 1 | WHILE A > B DO  B <- B + 1  ENDWHILE |  |
| FOR I = 1 TO 10 OUTPUT I\*I | FOR I <- 1 TO 10  OUTPUT I\*I  ENDFOR |  |

**FLOWCHART RULES:**

* Except for the DECISION SYMBOL, Only 1 arrow in and 1 arrow out
* DECISION SYMBOL has 1 arrow in and 2 arrows out (Y and N)
* LINES cannot cross each other
* START terminator may be a FUNCTION declaration
* END terminator may be a RETURN
* SYNTAX follows Pseudo code conventions

|  |  |
| --- | --- |
| SYMBOLS |  |
|  | TERMINATOR (START, END, FunctionName, RETURN) |
|  | PROCESS |
|  | INPUT,OUTPUT |
|  | DECISION |

Pseudo-code Notes:



|  |  |
| --- | --- |
| Declaring variables | DECLARE MyScore : INTEGER |
| Declaring a 1-D array | DECLARE Marks: ARRAY[0:9] OF INTEGER  DECLARE Names: ARRAY[0:9] OF STRING |
| Declaring a 2-D array | DECLARE Marks: ARRAY[0:9] OF (ARRAY[0:3] OF STRING) |
| Using an Abstract Data Type like List, Dictionary | DECLARE Marks : LIST  DECLARE Scores : DICTIONARY  Marks.APPEND(10) // Describe APPEND method  Marks.SUBLIST(2,4) // Describe SUBLIST to implement slicing operator in Python List  Scores["John"] = 58 // Store a dictionary object with a key "John" |
| Declaring a function/method | FUNCTION Factorial(n : INTEGER) RETURNS INTEGER  DECLARE Result : INTEGER  Result ← 1  FOR i ← 1 TO n  Result ← Result \* i  ENDFOR  RETURN Result  ENDFUNCTION |
|  |  |

Practice:

1. Create a flowchart to model the following Python code:

|  |
| --- |
| a = 1  while a % 7 != 0:      if a % 2 == 0:          print("O" , end = "")      if a == 2:          print("X" , end = "")      a = a + 1  print("") |

1. Write pseudo code for the following Python code:

|  |
| --- |
| divisor = 2  for i in range(0, 10, 2):      print(i // divisor) |

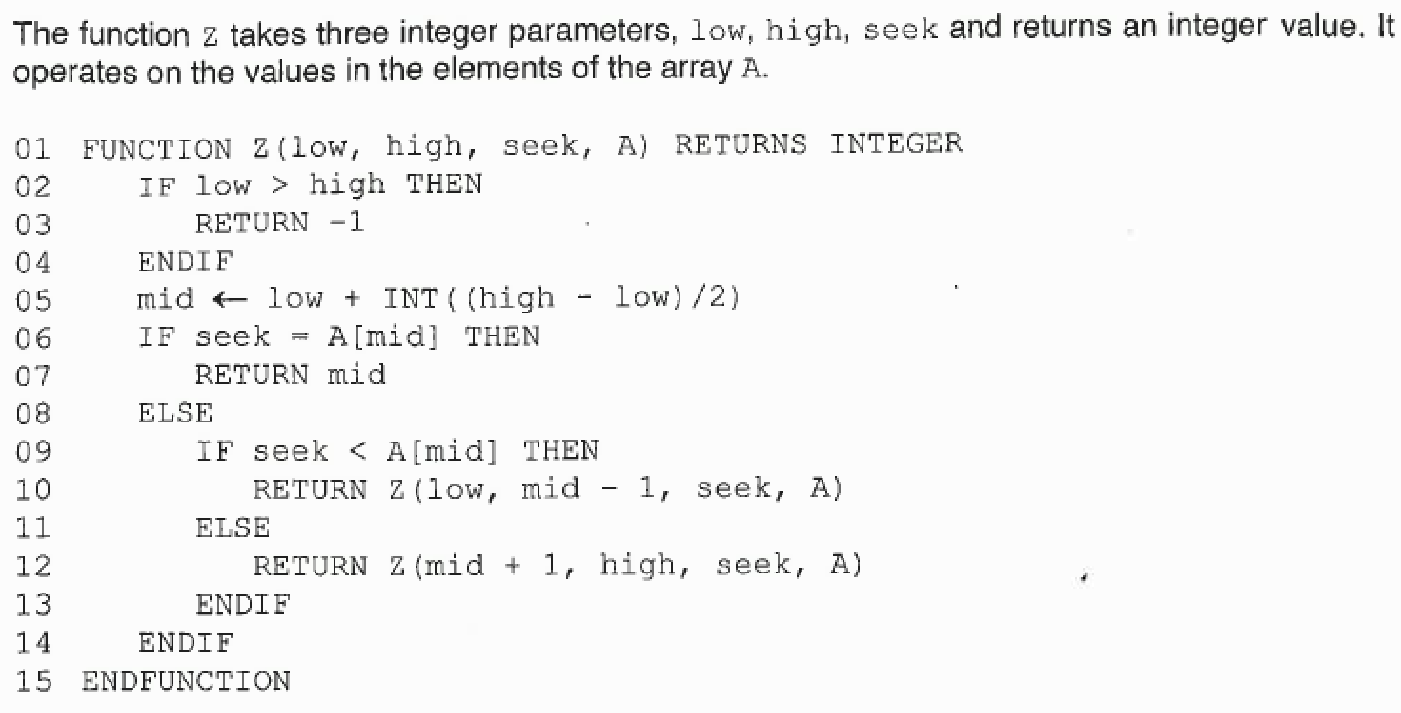
|  |
| --- |
|  |

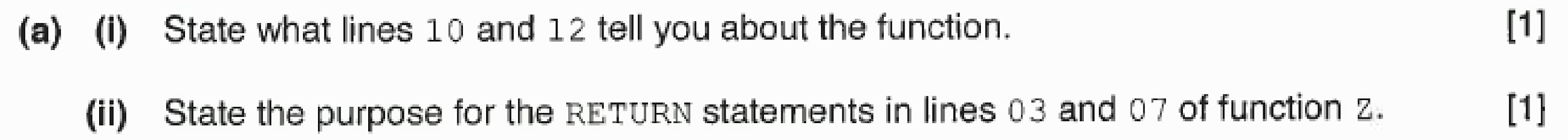
1. You are tasked to find the smallest and second smallest integers in an array of integers in a single pass. You are not allowed to sort the array. You can assume that the array has already been initialised with a list of distinct integers.

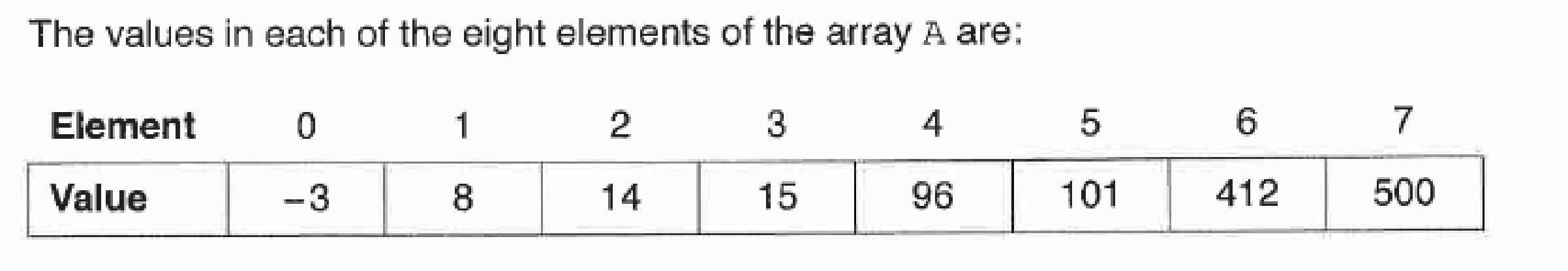
Express your algorithm using pseudocode.

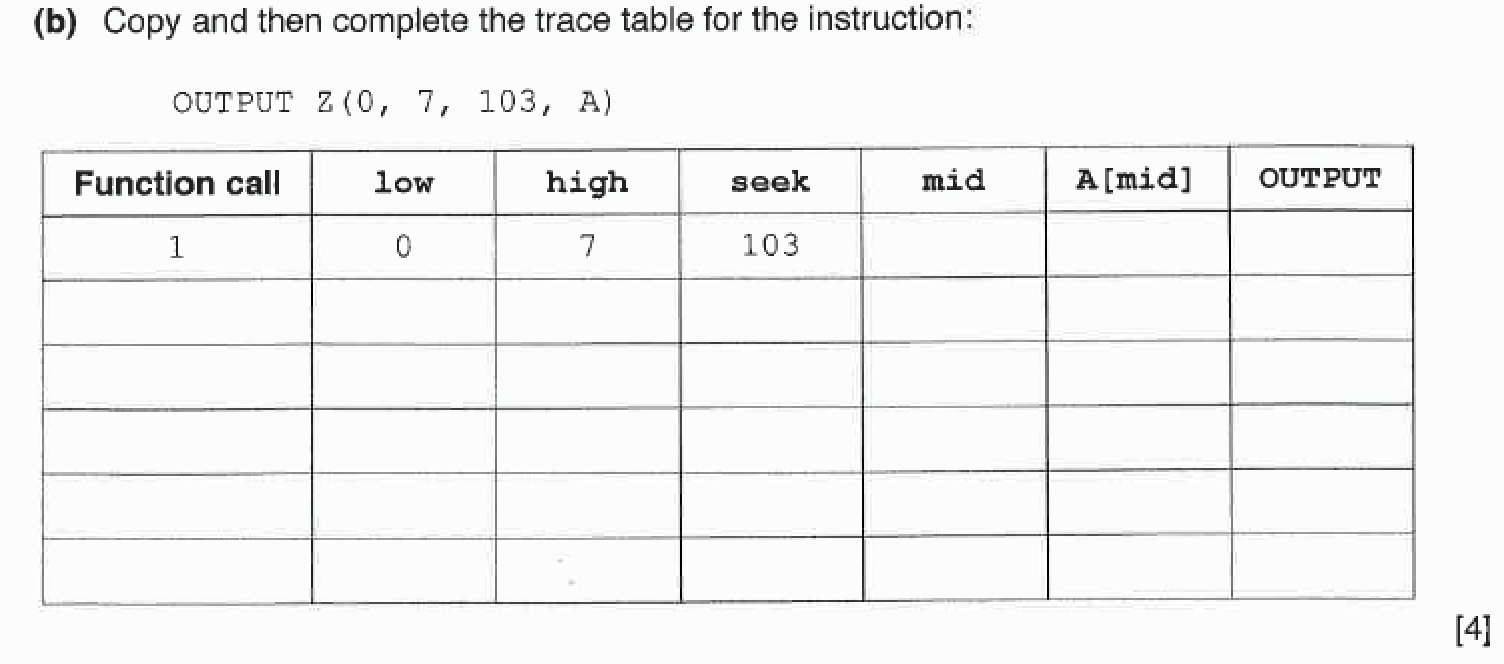
|  |
| --- |
|  |

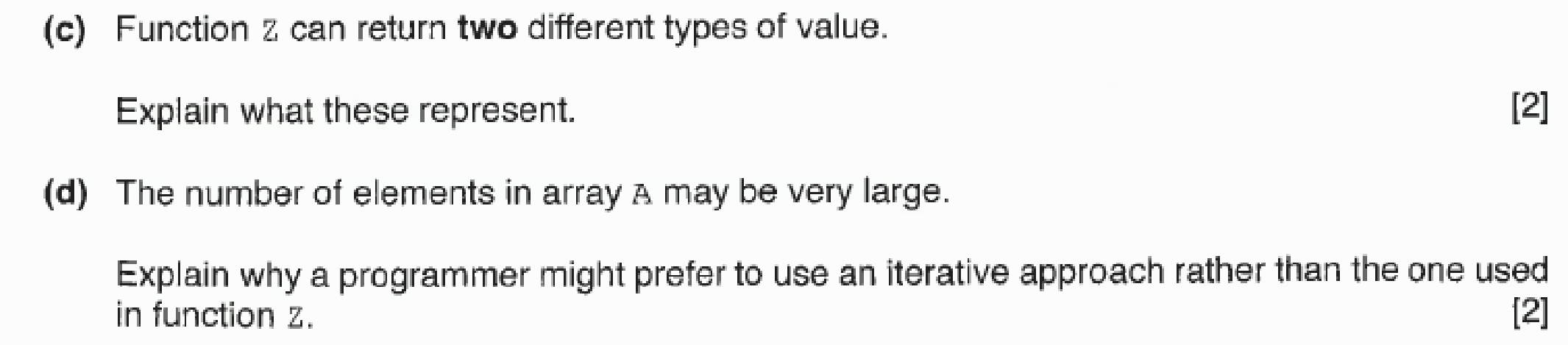
## **Exercise 13.4 2019/A Level/P1/Q5 H2 Computing (Modified)**











**Algorithm Modelling - References**

1. **General Algorithm Design**

* **Github Notes Chapter 08., Exercise 13**
* **Cambridge International AS and A Level - Computer Science (9806)**
  + Chapter 11
* **AQA Computing – AS Computing**
  + Unit 1.4
* **BBC Bitesize Revision**
  + https://www.bbc.com/bitesize/guides/z3bq7ty/revision/1

**1. Pseudocode**

* **Cambridge Pseudocode Guide**
  + <https://drive.google.com/file/d/1vBm4iswJF66zYW-4oNj1NkRwC1DHbZ3o/view?usp=sharing>

**2. Flow Charts**

* **Creately Tutorial on Drawing Flow Charts**
  + <https://creately.com/blog/diagrams/flowchart-guide-flowchart-tutorial/>
* **Online Flow Chart Drawing Tool**
  + <https://www.draw.io/>