



GCE A LEVEL

1500U30-1

MONDAY, 10 JUNE 2024 – AFTERNOON

COMPUTER SCIENCE – A2 unit 3

Programming and System Development

2 hours plus your additional time allowance

ADDITIONAL MATERIALS

A WJEC pink 16-page answer booklet.

A calculator.

INSTRUCTIONS TO CANDIDATES

Use black ink, black ball-point pen or your usual method.

Answer ALL questions.

Write your answers in the separate answer booklet provided.

Write the question number in the left-hand margin at the start of each answer.

INFORMATION FOR CANDIDATES

The number of marks is given in brackets at the end of each question or part-question; you are advised to divide your time accordingly.

The total number of marks available is 100.

Assessment will take into account the quality of written communication used in your answers.

(Turn over)

Answer ALL questions.

1 (a) Explain, giving a suitable example for each, the difference between a two-dimensional and a three-dimensional array. [4 marks]

(b) This is a graphical representation of a two-dimensional string array:

july[n,n]

	0	1	2	3	4	5	6
0	Mon	Tue	Wed	Thu	Fri	Sat	Sun
1	01	02	03	04	05	06	07
2	08	09	10	11	12	13	14
3	15	16	17	18	19	20	21
4	22	23	24	25	26	27	28
5	29	30	31	X	X	X	

Describe how you would update the empty element in the array to x. [2 marks]

(Turn over)

- 1 (c) This is a representation of a two-dimensional integer array containing the values 1 to 9:

$2D = [[1, 2, 3], [4, 5, 6], [7, 8, 9]]$

Describe how you would change the value 8 to 10 in the 2D array. [2 marks]

2. Computer languages such as HTML and CSS are standardised.

(a) Describe the need for the standardisation of computer languages. [4 marks]

(b) Explain, giving a suitable example, the potential difficulties involved in agreeing and implementing standards. [4 marks]

3. Clearly showing each step, simplify the following Boolean expressions using Boolean algebra, identities and De Morgan's Law where appropriate.
- (a) $(1.X + \overline{Y.Y}).\overline{X} + X.Z$ [6 marks]
- (b) $(\overline{A.A} + A.A).1 + B.C.0$ [6 marks]
4. Write an algorithm to search an unsorted one-dimensional array of strings and replace any duplicate values with the string "X" and then output the array. [8 marks]

5. The algorithm on the separate A3 sheet validates a password for a given string.
- (a) Using a table, select **THREE** appropriate pieces of test data to dry-run the algorithm and trace the variables `x`, `y`, `z`, `valid` and the expected output. [6 marks]
- (b) Suggest suitable text for message 1 and message 2. [2 marks]

6. When evaluating computer-based solutions, there are several criteria that can be considered.

Describe the following criteria when evaluating computer-based solutions and give suitable examples:

- (a) Usability [2 marks]
- (b) Performance [2 marks]
- (c) Scalability [2 marks]
- (d) Security [2 marks]

7. A website URL is made up of a protocol, a domain name, and an optional file path.
- The protocol can only be “http” or “https”.
 - The domain name can only consist of alphanumeric characters, hyphens and full stops.
 - The protocol and domain name must be separated by a colon and two forward slashes.
 - The optional file path must start with a forward slash and can only contain alphanumeric characters and forward slashes.

Example: `https://www.wjec.co.uk/home/`

Produce a Backus-Naur form (BNF) definition for a valid website URL. [6 marks]

8. Explain program version management. [6 marks]
9. Explain, using suitable examples, recursive and non-recursive sorting algorithms. [8 marks]

(Turn over)

10. This is a signed 8-bit integer:

00001111_2

Include this integer in a worked example to demonstrate how masking can be used to determine the sign of the integer. [3 marks]

```
1  declare i,j as integer
2  declare myArray1[n]
3  declare myArray2[n]
4  declare myArray3[]
5
6  set i = 0
7  set j = 0
8
9  for i = 0 to len(myArray1) - 1
10     set myArray3[num(myArray3)] =
        myArray1[i]
11 next i
12
13 for j = 0 to len(myArray2) - 1
14     set myArray3[num(myArray3)] =
        myArray2[j]
15 next j
16
17 output myArray3
```

11. This algorithm (opposite) merges two one-dimensional arrays of length n . Assume the arrays (`myArray1` and `myArray2`) have already been populated with data.

`num(array)` returns the number of elements currently in the array.

- 11 (a) Evaluate the efficiency of the algorithm and, using Big O notation, determine the growth rate for time performance. [5 marks]
- (b) Determine the growth rate of memory space during a single run of the algorithm. [2 marks]
- (c) Identify the type of time complexity and draw a graph of the algorithm to illustrate the order of time performance. Graph paper is not required. [2 marks]
12. Explain, giving a suitable example, the shortest path algorithm. [4 marks]

- 13. Discuss the potential use of natural language interfaces in human-computer communication to address the problem of communicating with computers.**

You should draw on your knowledge, skills and understanding from a number of areas across your computer science course when answering this question. [12 marks]

END OF PAPER

```
1 declare password as string
2 declare digits as char[]
3 declare special as char[]
4 declare length, i as integer
5 declare valid, w, x, y, z as Boolean
6 set digits = {'0','1','2','3','4','5','6','7','8','9'}
7 set special = {'!','@','#','$','%','&','*','?'}
8 set length = 8
9 set i, j = 0
10 set valid, x, y, z = FALSE
11
12 do
13     output "Enter a password: "
14     input password
15     if len(password) >= length
16         set x = TRUE
17     end if
18     for i = 0 to len(password) - 1
19         for j = 0 to len(digits) - 1
20             if password[i] = digits[j]
21                 set y = TRUE
22             end if
23         next j
24     next i
25     set j = 0
26     for i = 0 to len(password) - 1
27         for j = 0 to len(special) - 1
28             if password[i] = special[j]
29                 set z = TRUE
30             end if
31         next j
32     next i
33     if NOT x OR NOT y OR NOT z then
34         output "message 1"
35     else
36         output "message 2"
37         set valid = TRUE
38 while valid = FALSE
```