Cambridge Assessment International Education

Cambridge International AS & A Level

CANDIDATE NAME Parth

CENTRE NUMBER | | | | | |

CANDIDATE NUMBER | | | | |

2579440869

COMPUTER SCIENCE **9618/31**

Paper 3 Advanced Theory **May/June 2021**

1 hour 30 minutes

You must answer on the question paper.

No additional materials are needed.

INSTRUCTIONS

- Answer **all** questions.
- Use a black or dark blue pen.
- Write your name, centre number and candidate number in the boxes at the top of the page.
- Write your answer to each question in the space provided.
- Do **not** use an erasable pen or correction fluid.
- Do **not** write on any bar codes.
- You may use an HB pencil for any diagrams, graphs or rough working.
- Calculators must **not** be used in this paper.

INFORMATION

- The total mark for this paper is 75.
- The number of marks for each question or part question is shown in brackets [ ].
- No marks will be awarded for using brand names of software packages or hardware.

This document has **12** pages.

1 Real numbers are stored in a computer system using floating-point representation with:

- 10 bits for the mantissa
- 6 bits for the exponent
- Two's complement form for both the mantissa and the exponent.

(a) Calculate the normalised floating-point representation of –7.25 in this system.
Show your working.

Mantissa

1	0	0	0	1	1	0	0	0	0

Exponent

0	0	0	1	1	1

Working 00000111.01

Exp = 3

0.1110100000

1 000101111

+ 1

1.0001 10000

[3]

(b) Calculate the denary value of the given binary floating-point number.
Show your working.

Mantissa

1	0	1	1	0	0	0	1	1	1

Exponent

4 2 1

0	0	0	1	1	1

Working 128 64 32 16 8 4 2 1

1 0 1 1 0 0 0 1 1 1

1 0 1 1 0 0 0 1 · 1 1

−128 + 32 + 16 + 1 + 0.75

7 1
128
−49.75 32
78.25 + 16
48
49

Answer 78.25

[3]

(c) The given binary floating-point number is not normalised.

Normalise the floating-point number. Show your working.

Mantissa										Exponent					
										-32			4	2	1
0	0	0	0	0	0	0	1	1	1	1	0	0	1	1	1

Mantissa										Exponent					
0	1	1	1	0	0	0	0	0	0	1	0	0	0	0	1

Working 0.0000001 1 1

0.11100000... Exp -6 7 to 32

Exp = -25

Exp = -25 - 6 = -31

[3]

(d) The denary number 513 cannot be stored accurately as a normalised floating-point number in this computer system.

(i) Explain the reason for this.

- There are not enough bits allocated to the mantissa.

[3]

(ii) Describe an alteration to the way floating-point numbers are stored to enable this number to be stored accurately using the same total number of bits.

- Allocate more 1 bit to the mantissa and - Reduce the exponent's number of

[2]

2 **(a)** Describe the purpose of a user-defined data type.

.....

.....

.....

.....

..... [2]

(b) Define, using pseudocode, the following enumerated data types:

(i) `SchoolDay` to hold data about the days students are usually in school.

TYPE SchoolDay ← (Monday, Tuesday, Wednesday, Thursday, Friday)

..... [1]

(ii) `WeekEnd` to hold data about the days that are not school days.

TYPE WeekEnd ← (Saturday, Sunday)

..... [1]

(c) Define, using pseudocode, the composite data type `ClubMeet`. This will hold data about club members that includes:

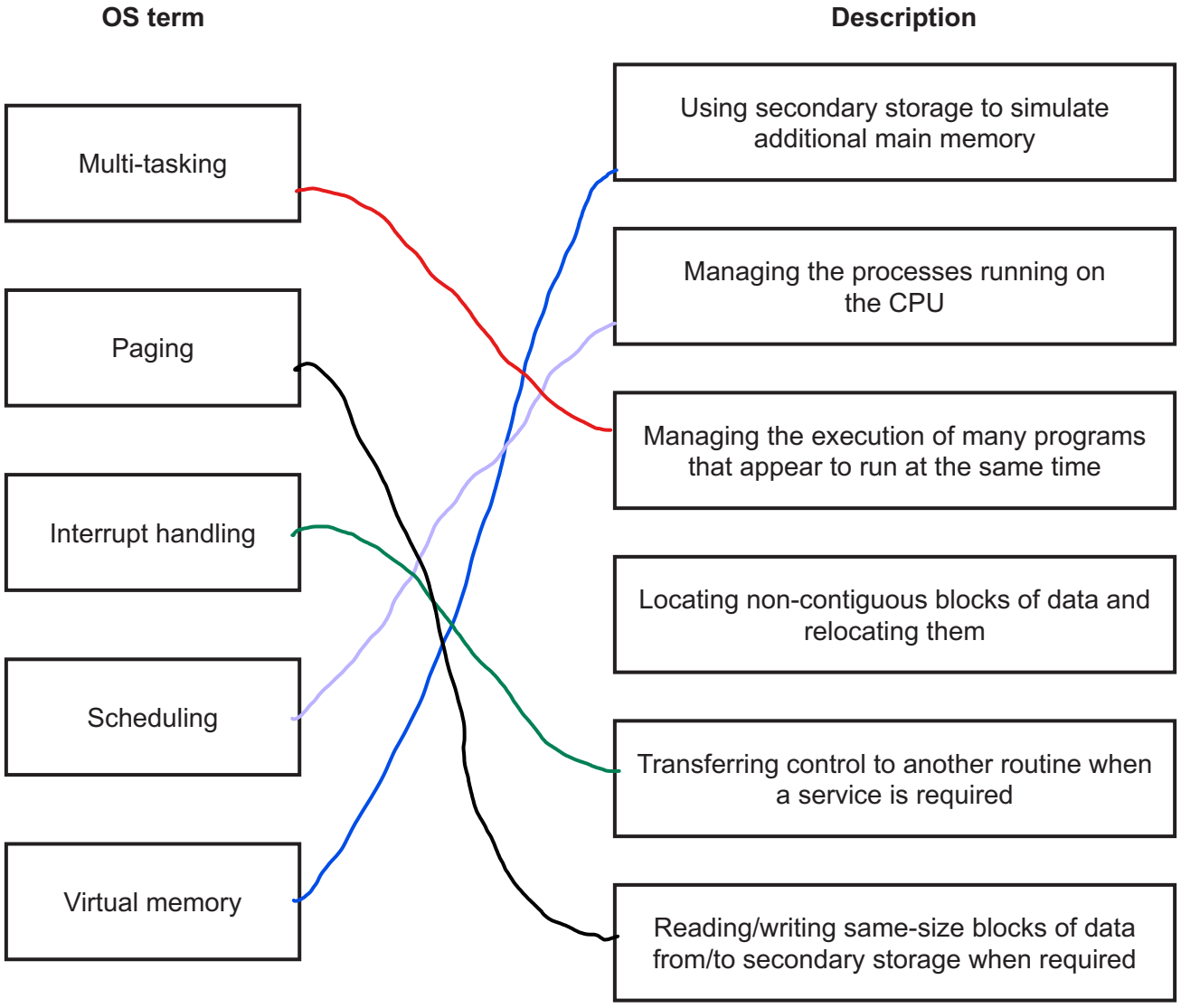
- first name and last name
- the two days they attend:
  - one on a school day
  - one not on a school day.

Use the enumerated types you created in **part (b)**.

```
TYPE ClubMeet
    DECLARE FirstName : STRING
    DECLARE Surname : STRING
    DECLARE SDay : SchoolDay
    DECLARE WDay : WeekEnd
ENDTYPE
```

..... [4]

3 **(a)** Draw **one** line to connect each **Operating System (OS)** term to the **most appropriate** description about it.



[5]

(b) Explain how an interpreter executes a program without producing a complete translated version of it.

.....

.....

.....

.....

.....

.....

.....

..... [4]

4 **(a)** **(i)** Explain why Reverse Polish Notation (RPN) is used to carry out the evaluation of expressions.

No need for brackets
All expressions can be expressed without the need for the rule of preceedence.

[2]

(ii) Identify, with reasons, a data structure that could be used to evaluate an expression in RPN.

The stack; it handles data in reverse order of input (First In Last Out).

[2]

(b) Write the infix expression in RPN.

```
(a - b) * (a + c) / 7
```

ab - ac + * 7 /

[1]

(c) Write the RPN expression as an infix expression.

```
a b / 4 * a b + -
```

(a/b) * 4 - (a+b)

[1]

(d) Evaluate the RPN expression:

```
a b + c d / /
```

where `a = 17, b = 3, c = 48 and d = 12.`

Show your working.

(17+3) / (48/12)

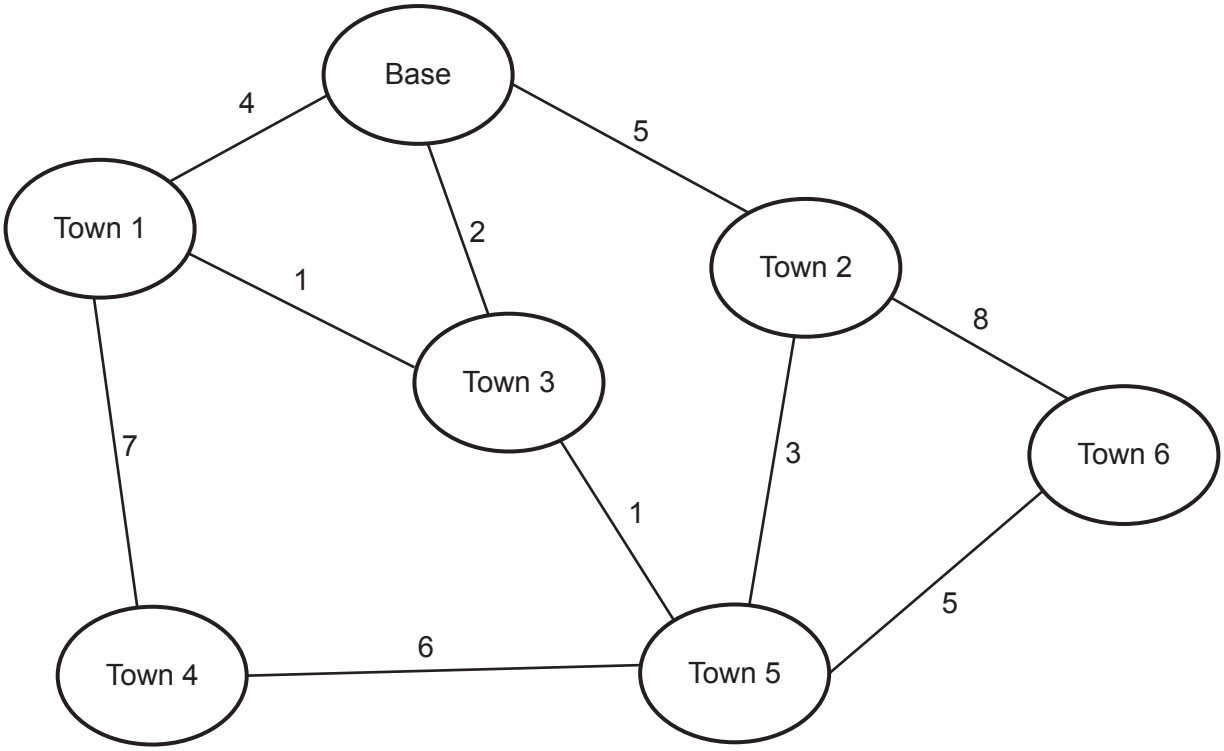
20 / 4

5

[2]

5 **(a)** Calculate the shortest distance between the base and each of the other towns in the diagram using Dijkstra's algorithm.

Show your working **and** write your answers in the table provided.



Working

Town 1: 4, 2+1 Town 6: 5+8, 5+3+5, 2+1+5

Town 2: 5, 2+1+3

Town 3: 2, 4+1

Town 4: 7+4, 4+1+1+6, 4+2+1+6

Town 5: 2+1, 5+3, 4+1+1

Answers

Town 1	Town 2	Town 3	Town 4	Town 5	Town 6
3	5	2	11	3	8

[5]

(b) Explain the use of graphs to aid Artificial Intelligence (AI).

.....

.....

.....

.....

.....

..... [3]

6 Give **two** benefits **and two** drawbacks of packet switching.

Benefit 1 Multiple users can use it at the same time

.....

Benefit 2 Make the most use of resources as compared to circuit switching.

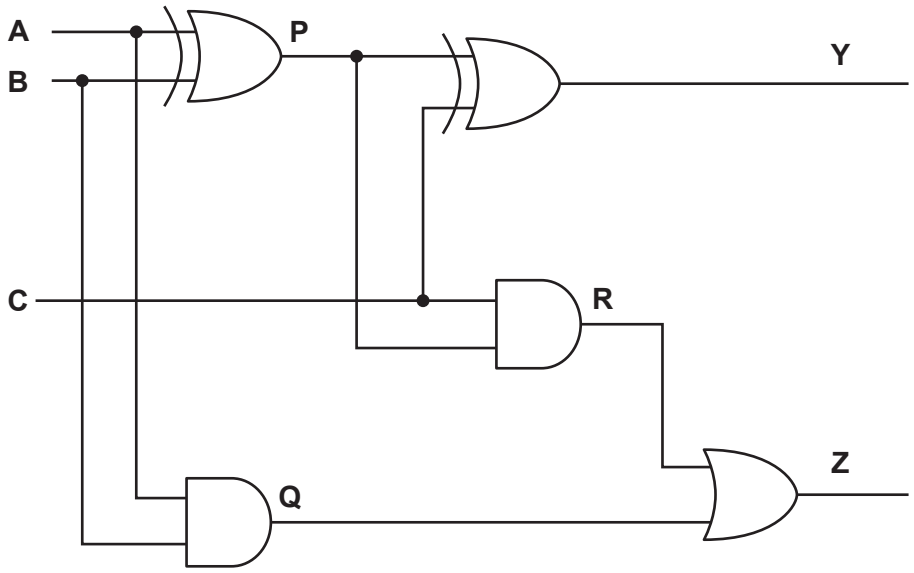
Drawback 1 Becomes slower as the number of users increases

Drawback 2 Not suitable for Real-Time services

.....

[4]

7 The diagram shows a logic circuit.



(a) Complete the truth table for the given logic circuit. Show your working.

Inputs			Working space			Outputs	
A	B	C	P	Q	R	Y	Z
0	0	0	0	0	0	0	0
0	0	1	0	0	0	1	0
0	1	0	1	0	0	1	0
0	1	1	1	0	1	0	1
1	0	0	1	0	0	1	0
1	0	1	1	0	1	0	1
1	1	0	0	1	0	0	1
1	1	1	0	1	0	1	1

[3]

(b) State the name of the logic circuit.

Full Adder ..... [1]

(c) Write the Boolean expressions for the two outputs **Y** and **Z** in the truth table as sum-of-products **and** state the purpose of each output.

Y = $\bar{A} \cdot \bar{B} \cdot C + \bar{A} \cdot B \cdot \bar{C} + A \cdot \bar{B} \cdot \bar{C} + A \cdot B \cdot C$

Purpose Sum Bit

Z = $\bar{A} \cdot B \cdot C + A \cdot \bar{B} \cdot C + A \cdot B \cdot \bar{C} + A \cdot B \cdot C$

Purpose Carry Bit

[4]

8 **(a)** State **two** factors that may affect the performance of a sorting algorithm.

.....

.....

.....

..... [2]

(b) The given algorithm is a simple bubble sort that arranges a set of scores stored in a one-dimensional array into **descending** order, and orders the corresponding students' names stored into a two-dimensional array in the same order as the scores. All the arrays are indexed from 1.

The contents of both arrays after sorting are shown.

	Score
1	98
2	97
…	
248	5
249	3

	Name	
	1	**2**
1	Smithfield	Tom
2	Johnson	Jane
…		
248	Peters	Jade
249	Allen	John

```
YearSize ← 249
Flag ← TRUE
WHILE Flag = TRUE
    Flag ← FALSE
    FOR Student ← 1 TO YearSize - 1
        IF Score[Student] < Score[Student + 1] THEN
           Temp1 ← Score[Student]
           Temp2 ← Name[Student,1]
           Temp3 ← Name[Student,2]
           Score[Student] ← Score[Student + 1]
           Name[Student,1] ← Name[Student + 1,1]
           Name[Student,2] ← Name[Student + 1,2]
           Score[Student + 1] ← Temp1
           Name[Student + 1,1] ← Temp2
           Name[Student + 1,2] ← Temp3
           Flag ← TRUE
        ENDIF
    NEXT Student
ENDWHILE
```

Write an algorithm, using pseudocode, that will perform the same task using an insertion sort.

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

..... [6]

9 **(a)** Describe what is meant by **an imperative (procedural)** programming language.

.....

.....

.....

..... [2]

(b) Describe what is meant by **a declarative** programming language.

.....

.....

.....

..... [2]

(c) Identify the programming paradigm for each of these program code examples.

Program code example	Programming paradigm

`male(john).`
`female(ethel).`
`parent(john, ethel).`
	Declarative

`FOR Counter = 1 TO 20`
`  X = X * Counter`
`NEXT Counter`
	Procedural

`Start: LDD Counter`
`  INC ACC`
`  STO Counter`
	Low Level

`public class Vehicle`
`{`
`  private speed;`
`  public Vehicle()`
`  {`
`  speed = 0;`
`  }`
`}`
	Object Oriented

[4]