Cambridge Assessment International Education

Cambridge International AS & A Level

CANDIDATE NAME

CENTRE NUMBER

CANDIDATE NUMBER

6943099556

COMPUTER SCIENCE **9608/33**

Paper 3 Advanced Theory **May/June 2020**

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 **16** pages. Blank pages are indicated.

© UCLES 2020

1 In a particular computer system, real numbers are stored using floating-point representation with:

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

(a) Calculate the normalised floating-point representation of +192.5 in this system. Show your working.

Mantissa **Exponent**



Working .....

.....

.....

.....

..... [3]

(b) Calculate the normalised floating-point representation of –192.5 in this system. Show your working.

Mantissa **Exponent**



Working .....

.....

.....

.....

..... [3]

(c) The floating-point representation has changed. There are now 12 bits for the mantissa and 4 bits for the exponent as shown.



Explain why +192.5 cannot be accurately represented in this format.

.....

.....

.....

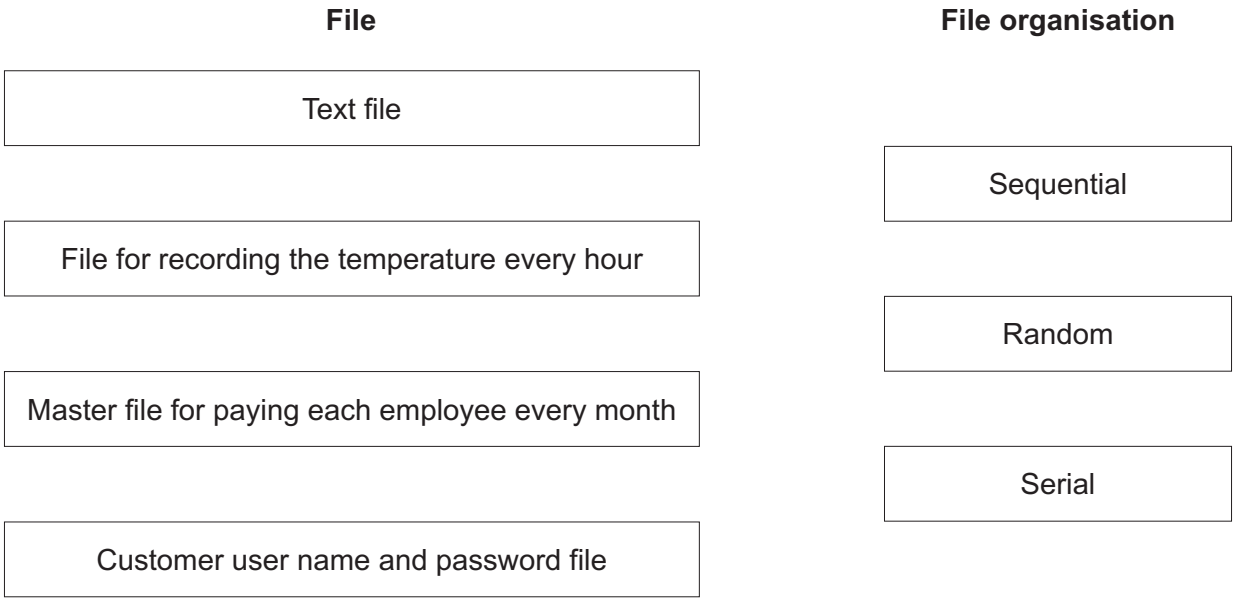
.....

.....

..... [3]

2 The diagram shows four files and three methods of file organisation.

Draw **one** line to match each file with its most appropriate method of file organisation.



[4]

3 A mobile phone company uses circuit switching for voice calls and packet switching to send and receive other data.

(a) (i) Describe circuit switching.

.....

.....

.....

.....

.....

..... [3]

(ii) Explain why the company uses circuit switching for voice calls.

.....

.....

.....

..... [2]

(b) (i) Describe packet switching.

.....

.....

.....

.....

.....

..... [3]

(ii) Explain why the company uses packet switching to send and receive other data.

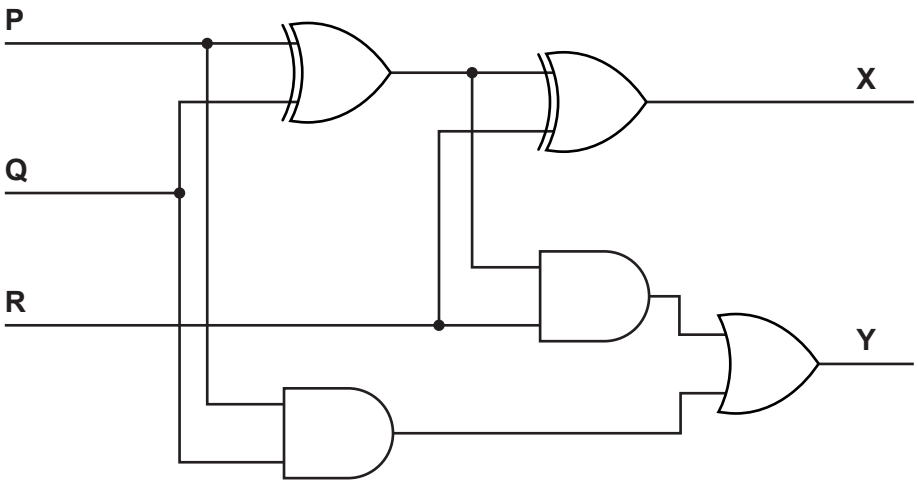
.....

.....

.....

..... [2]

4 **(a)** Write the Boolean algebraic expressions for the following logic circuit.



X = ...

.....

Y = ...

.....

[5]

(b) The logic circuit given in **part (a)** is a full adder.

(i) Give the purpose of outputs **X** and **Y** in this circuit.

X .....

Y .....

[2]

(ii) Give the use of the input **R** in this circuit.

..... [1]

5 Complete these **three** statements about computer processors.

A processor with a few simple fixed-length instructions that have a small number of instruction formats is called a ..... processor.

A processor with many complex variable-length instructions that has many instruction formats is called a ..... processor.

Instruction-level parallelism, applied to the execution of instructions during the fetch-execute cycle, is called ..... .

[3]

6 Duraid writes a short program in a high-level programming language. An interpreter executes the program.

The following is part of Duraid's program.

```
DECLARE P, Q, R, X, Y : INTEGER
CONSTANT M = 10

P = 4
Q = 2
R = 1
X = (P + Q) * (P – Q)
Y = (M / Q) * (P + Q - R)
```

(a) Write the Reverse Polish Notation (RPN) for the following expression from Duraid's program.

```
(P + Q) * (P – Q)
```

..... [2]

(b) The interpreter is executing Duraid's program. The expressions are in infix form.

The interpreter converts the infix to RPN.

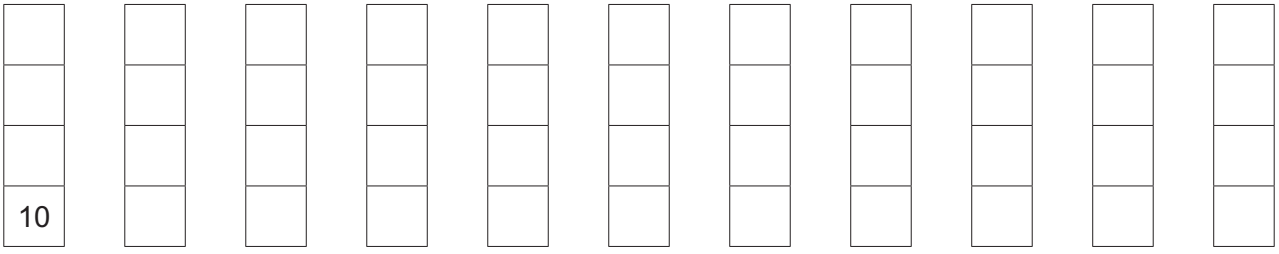
The RPN expression for `Y` is:

```
M Q / P Q + R – *
```

The interpreter evaluates this RPN expression using a stack.

(i) Show the changing contents of the stack, as the interpreter evaluates the expression for `Y`.

Use the values of the variables and constant given in the program. The first entry has been done for you.



[4]

(ii) Convert the following RPN expression back to its infix form.

```
P Q + M * R P - -
```

.....

..... [2]

(c) Explain how RPN is used by an interpreter to evaluate expressions.

.....

.....

.....

..... [2]

7 A computer at a remote weather station is performing three tasks:

- measuring and recording the temperature every 10 seconds
- measuring and recording the wind speed every 10 seconds
- sending the previous day's temperature and wind speed readings to a scientist at another location via the Internet.

The operating system is managing the multitasking of these tasks.

(a) At one point in time:

- the temperature measuring and recording task is idle
- the wind speed is being recorded
- the task to send the previous day's temperature and wind speed readings is waiting for an internet connection.

Identify the process state for each task. Give a reason why each task is in that process state.

Temperature measuring and recording process state .....

Reason .....

.....

.....

Wind speed measuring and recording process state .....

Reason .....

.....

.....

Sending process state .....

Reason .....

.....

.....

[6]

(b) The weather station computer uses an operating system.

Explain how this operating system uses interrupts to schedule the measuring and recording tasks.

.....

.....

.....

.....

.....

.....

.....

..... [4]

8 Martha wants to send a private message to Joshua over the Internet.

(a) Martha and Joshua's computers have already exchanged digital certificates.

Identify **three** items that could be contained in a digital certificate.

1 .....

.....

2 .....

.....

3 .....

.....

[3]

(b) Joshua and Martha's digital certificates are used to ensure that Martha's message has not been altered during transmission.

Explain how asymmetric encryption uses the contents of the digital certificates to ensure that the message has not been altered during transmission.

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

[6]

9 A train cannot move if any of the eight automatic train doors are open. The train door monitoring system, set out below, checks that all the doors are closed before the train can move.

- If a monitoring system detects that a door is open, it sets a specific bit in address `500` to `1`.
- If the bit for door one is equal to `1`, the binary value for hexadecimal `FF` is sent to address `501`. The contents of address `501` are changed to make door 1's light flash when the door is open.
- If the bit for door two is equal to `1`, the binary value for hexadecimal `FF` is sent to address `502`. The contents of address `502` are changed to make door 2's light flash when the door is open.

  This is repeated for each door from 3 to 8.
- Each door sets its bit in address `500` to zero when the door closes, and the contents of the corresponding door address are set to zero.
- The train manager can identify which door is open from the flashing light.

The current contents of address `500` are:

	Door number							
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**
Address `500`	1	0	0	1	0	0	1	0

(a) Complete the following table by writing the values stored in addresses `503` to `508`. Use the contents of address `500` shown above. Note that addresses `501` and `502` are complete.

`501`	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**	Door 1
`502`	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	Door 2
`503`									Door 3
`504`									Door 4
`505`									Door 5
`506`									Door 6
`507`									Door 7
`508`									Door 8

[2]

(b) The following table shows assembly language instructions for the processor controlling the train door monitoring system that has one general purpose register, the Accumulator (ACC).

	Instruction		Explanation
Label	**Op code**	**Operand**	
	LDM	&n	Load the hexadecimal number n to ACC
	LDD	<address>	Load the contents of the location at the given address to ACC
	STO	<address>	Store the contents of ACC at the given address
	AND	&n	Bitwise AND the contents of ACC with the hexadecimal number n
	CMP	&n	Compare the contents of ACC with the hexadecimal number n
	JPE	<address>	Following a compare instruction, jump to <address> or <label> if the compare was True
<label>:	<op code>	<operand>	Labels an instruction
	WAIT		Macro to wait one second before the next instruction is executed

After rechecking the doors, address 500 now contains 10101010.

(i) Complete the table by writing the values of the Accumulator (ACC) and the contents of address 501 as these instructions are executed **once** to check door **1**.

	Instruction		ACC	501
Label	**Op code**	**Operand**		
CHECK1:	LDD	500		
	AND	&80		
	CMP	&00		
	JPE	DOOR1		
	LDM	&FF		
DOOR1:	STO	501		
	WAIT			
	LDM	&00		
	STO	501		
	WAIT			
	JMP	CHECK1		

[4]

© UCLES 2020

(ii) Write the assembly language instructions to check door **2**.

	Instruction	
Label	**Op code**	**Operand**

[4]

(c) Explain how the check door routines show a flashing light or no light.

.....

.....

.....

..... [2]

© UCLES 2020

BLANK PAGE

BLANK PAGE

© UCLES 2020

BLANK PAGE

Permission to reproduce items where third-party owned material protected by copyright is included has been sought and cleared where possible. Every reasonable effort has been made by the publisher (UCLES) to trace copyright holders, but if any items requiring clearance have unwittingly been included, the publisher will be pleased to make amends at the earliest possible opportunity.

To avoid the issue of disclosure of answer-related information to candidates, all copyright acknowledgements are reproduced online in the Cambridge Assessment International Education Copyright Acknowledgements Booklet. This is produced for each series of examinations and is freely available to download at www.cambridgeinternational.org after the live examination series.

Cambridge Assessment International Education is part of the Cambridge Assessment Group. Cambridge Assessment is the brand name of the University of Cambridge Local Examinations Syndicate (UCLES), which itself is a department of the University of Cambridge.

© UCLES 2020