Cambridge International AS & A Level

Cambridge International Examinations
Cambridge International Advanced Subsidiary and Advanced Level

CANDIDATE NAME

CENTRE NUMBER

CANDIDATE NUMBER

0586807379

COMPUTER SCIENCE **9608/32**

Paper 3 Advanced Theory **October/November 2017**

1 hour 30 minutes

Candidates answer on the Question Paper.

No Additional Materials are required.

No calculators allowed.

READ THESE INSTRUCTIONS FIRST

Write your Centre number, candidate number and name in the spaces at the top of this page.
Write in dark blue or black pen.
You may use an HB pencil for any diagrams, graphs or rough working.
Do not use staples, paper clips, glue or correction fluid.
DO **NOT** WRITE IN ANY BARCODES.

Answer **all** questions.
No marks will be awarded for using brand names of software packages or hardware.

At the end of the examination, fasten all your work securely together.
The number of marks is given in brackets [ ] at the end of each question or part question.

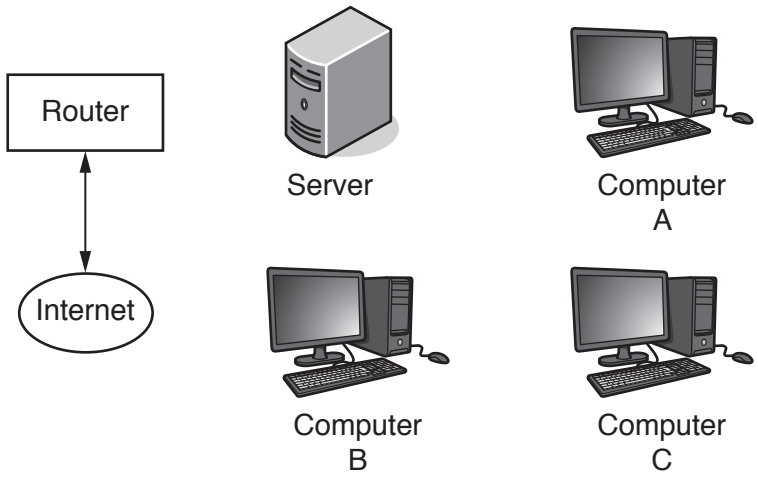
The maximum number of marks is 75.

This document consists of **15** printed pages and **1** blank page.

CAMBRIDGE International Examinations

1 A Local Area Network (LAN) consists of three computers, one server and a router connected to the Internet. The LAN uses a bus topology.

(a) Complete the following diagram to show how the computers, the server and the router could be connected.



[2]

(b) There are four statements in the following table. For each statement, place a tick (✓) in the appropriate column to indicate whether it is true or false.

Statement	True	False
The server can send packets to Computer B and the router at the same time.		
Computer C uses the IP address of a web server to send a request for a web page on the web server.		
Computer B can read a packet sent from Computer A to Computer C.		
The server can read all incoming packets from the Internet.		

[4]

(c) The user on Computer A and the user on Computer B are both using the Internet at the same time. On a few occasions, Computer A and Computer B start transmitting packets to the router at exactly the same time. This causes a problem called a collision.

(i) Explain what is meant by a **collision** in this context.

.....

.....

.....

.....[2]

(ii) As a result of the collision, both Computer A and Computer B stop transmitting.

Computer A must carry out a number of steps to ensure the successful transmission of its packet.

Give **two** of the steps.

Step 1 .....

Step 2 .....
[2]

(d) The LAN topology is redesigned.

(i) Describe the changes that could be made to the LAN topology to overcome the problem identified in **part (c)**.

.....

.....

.....

.....[2]

(ii) Explain how the redesign has overcome the problem.

.....

.....

.....

.....[2]

2 **(a)** The following diagram shows four descriptions and two types of processor.

Draw lines to connect each description to the appropriate type of processor.

Description	Type of processor
It has a simplified set of instructions.	
Emphasis is on the hardware rather than the software.	CISC
It makes extensive use of general purpose registers.	RISC
Many instruction formats are available.	

[4]

(b) In a RISC processor, instructions are processed using pipelining.

(i) Explain what is meant by **pipelining**.

.....

.....

.....

.....[2]

(ii) The following table shows the five stages that occur when instructions are fetched and executed. The table also shows a number of time intervals.

Two instructions, D followed by E, are fetched and executed. The 'E' in the incomplete table shows that instruction E has been fetched in time interval 2.

Complete each row of the table.

	Time interval							
Stage	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**
Fetch instruction		E						
Read registers and decode instruction								
Execute instruction								
Access operand in memory								
Write result to register								

[3]

(c) The instruction set for a RISC processor that allows pipelining includes the following instruction.

Instruction		Explanation
Op code	**Operands**	
`ADD`	`<dest>, <op1>, <op2>`	Add the integers in registers `op1` and `op2`. Place the result in register `dest`.

A program contains the following three instructions.

```
ADD r3, r2, r1

ADD r5, r4, r3

ADD r10, r9, r8
```

(i) Explain why pipelining fails for the first two instructions.

.....

.....

.....

.....[2]

(ii) The instructions were produced by a compiler after translation of a high-level language program.

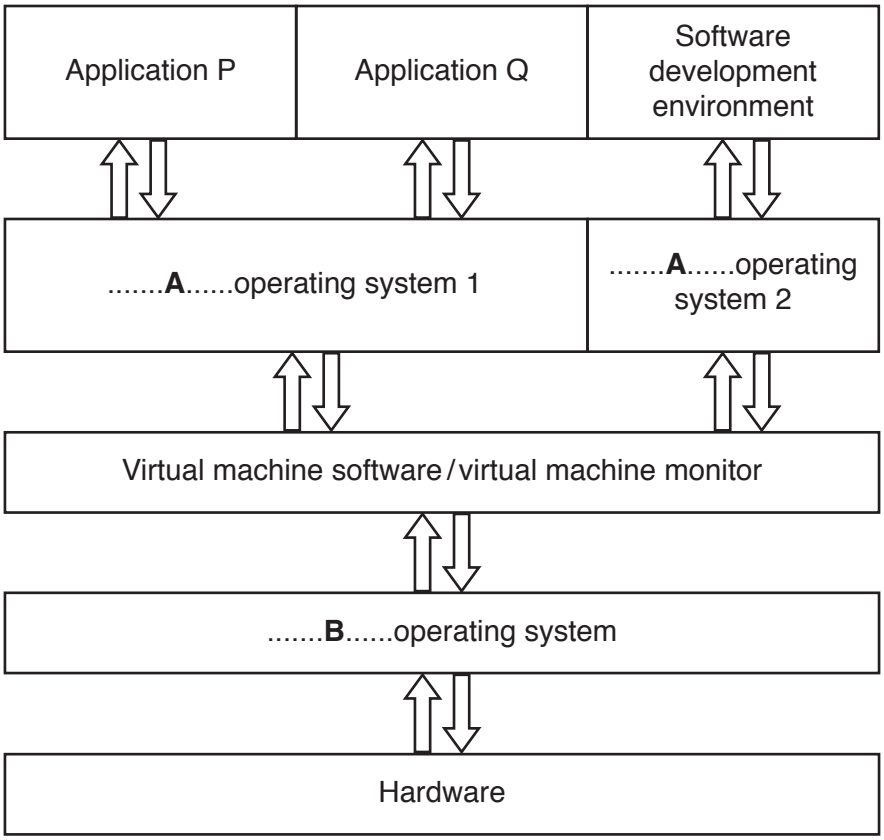
The compiler is not capable of code optimisation.

State how the code from the compiler could have been optimised to overcome the problem in **part (c)(i)**.

.....

.....[1]

3 **(a)** This diagram shows how applications P, Q and a software development environment can be run on a virtual machine system.



(i) State the operating systems labelled **A** and **B** in the diagram.

A .....

B .....
[2]

(ii) Application P is executing and requests data from a file.

Describe what happens after .....**A**.....operating system 1 has received the data request from the application.

.....

.....

.....

.....

.....

.....[3]

(b) A software development company uses virtual machines to produce software.

(i) State **one** benefit to the company.

.....

.....[1]

(ii) Explain **two** limitations of this approach.

Limitation 1 .....

.....

.....

.....

Limitation 2 .....

.....

.....

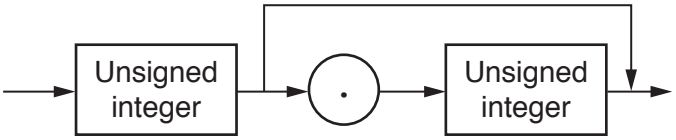
.....

[4]

4 The following syntax diagrams for a particular programming language show the syntax of:

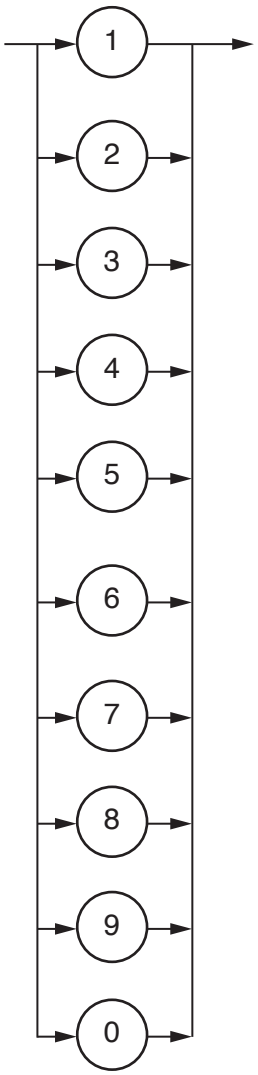
- an unsigned number
- an unsigned integer
- a digit.

Unsigned number



Unsigned integer

Digit



(a) (i) Explain why 32 is a valid unsigned integer.

.....

.....

.....

.....[2]

(ii) Explain why 32.5 is a valid unsigned number.

.....

.....

.....

.....[2]

(b) Complete the Backus-Naur Form (BNF) for the syntax diagrams shown.

<unsigned_number> ::= .....

.....

<unsigned_integer> ::= .....

.....

<digit> ::= .....

.....

[5]

The format of an unsigned number is amended to include numbers with possible exponents.

If an unsigned number has an exponent, then the exponent part:

- will start with an 'E'
- be followed by an optional '+' or '–' sign
- and be completed by an unsigned integer.

Examples of unsigned numbers with exponents include: 3E2, 3E+3, 3E–32, 3.45E–2

(c) **(i)** Redraw the syntax diagram for unsigned number to include numbers that might have exponents.

[4]

(ii) Use your syntax diagram from **part (c)(i)** to write the BNF for an unsigned number to include numbers with exponents.

<unsigned_number> ::= .....

.....

.....

.....

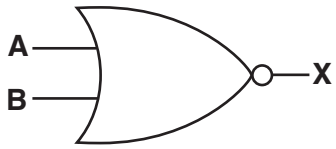
.....

.....

.....

.....[4]

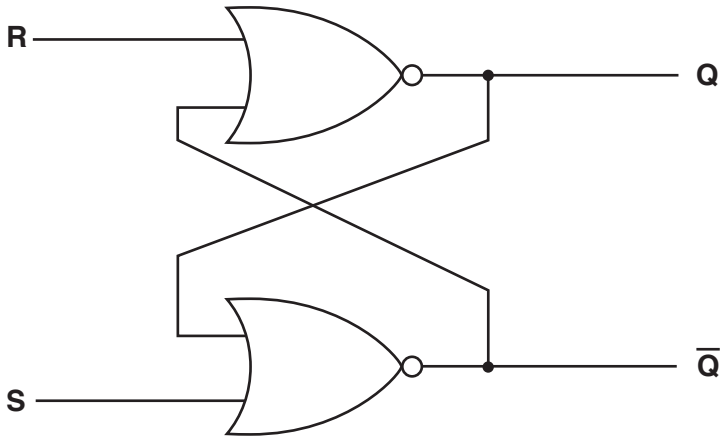
5 **(a)** Complete the truth table for this NOR gate:



A	B	X
0	0	
0	1	
1	0	
1	1	

[1]

A SR flip-flop is constructed using two NOR gates.



(b) Complete the truth table for the SR flip-flop:

	S	R	Q	$\overline{Q}$
Initially	1	0	1	0
S changed to 0	0	0		
R changed to 1	0	1		
R changed to 0	0	0		
S and R changed to 1	1	1		

[4]

[Turn over

Another type of flip-flop is the JK flip-flop. The JK flip-flop is an improvement on the SR flip-flop.

(c) (i) The JK flip-flop has three inputs. Two of the inputs are the Set (J) and the Reset (K).

State the third input.

.....[1]

(ii) There are **two** problems with the SR flip-flop that the JK flip-flop overcomes.

State each problem and state why it does not occur for the JK flip-flop.

Problem 1 .....

.....

.....

.....

Problem 2 .....

.....

.....

.....

[4]

6 The environment in a very large greenhouse is managed by a computer system. The system uses a number of different sensors that include temperature sensors. In addition, the system controls a number of heaters, windows and sprinklers.

(a) State **one** other type of sensor that could be used with this system.

Justify your choice.

Sensor .....

Justification .....

.....

[2]

(b) Describe why feedback is important in this system.

.....

.....

.....

.....

.....

.....[3]

(c) **(i)** The system makes use of a number of parameters. These parameters are used in the code that runs the system.

State **one** of the parameters used in controlling the temperature in the greenhouse.

.....[1]

(ii) Explain how the parameter identified in **part (c)(i)** is used in the feedback process.

.....

.....

.....

.....[2]

(d) There are eight temperature sensors numbered 1 to 8. Readings from these sensors are stored in four 16-bit memory locations. The memory locations have addresses from `4000` to `4003`. Each memory location stores two sensor readings as two unsigned binary integers.

Sensor 1 reading is stored in bits 8 to 15 of address `4000`; Sensor 2 reading is stored in bits 0 to 7 of address `4000` and so on. The diagram shows that the current sensor 1 reading has a value of 97.

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
4000	0	1	1	0	0	0	0	1	0	0	1	1	1	0	0	1
4001	1	1	0	0	0	0	0	0	1	0	1	0	0	0	0	0
4002	0	0	0	1	0	1	0	0	0	0	0	0	1	1	0	1
4003	1	0	0	0	0	0	1	0	1	1	0	0	0	1	0	1

(i) Give the denary value of the current reading for Sensor 5.

.....

.....

.....

.....[1]

(ii) The following table shows part of the instruction set for a processor. The processor has one general purpose register, the Accumulator (ACC).

Instruction		Explanation
Op code	**Operand**	
LDD	<address>	Direct addressing. Load the contents of the location at the given address to ACC.
AND	#n	Bitwise AND operation of the contents of ACC with the operand.
AND	<address>	Bitwise AND operation of the contents of ACC with the contents of <address>.
XOR	#n	Bitwise XOR operation of the contents of ACC with the operand.
XOR	<address>	Bitwise XOR operation of the contents of ACC with the contents of <address>.
OR	#n	Bitwise OR operation of the contents of ACC with the operand.
OR	<address>	Bitwise OR operation of the contents of ACC with the contents of <address>. <address> can be an absolute address or a symbolic address.
LSL	#n	Bits in ACC are shifted n places to the left. Zeros are introduced on the right hand end.
LSR	#n	Bits in ACC are shifted n places to the right. Zeros are introduced on the left hand end.

The reading for Sensor 5 is used in a calculation. The calculation is carried out by two assembly language instructions.

The first instruction loads the contents of the 16-bit location that contains the value for Sensor 5.

The second instruction moves the bits in Sensor 5 so that the 16-bit value is the value of Sensor 5.

Complete the two instructions in the following code. Use the instruction set provided.

```
LDD ..... // load the contents of the 16-bit location
          containing the value for Sensor 5 into the
          Accumulator

..... // move the bits in the Accumulator so that the
      Accumulator stores the value of Sensor 5 as an
      unsigned 16-bit binary integer
```

[3]

BLANK PAGE