Cambridge International A Level

Cambridge International Examinations
Cambridge International Advanced Level

CANDIDATE NAME

CENTRE NUMBER

CANDIDATE NUMBER

6550085963

COMPUTER SCIENCE **9608/32**

Paper 3 Advanced Theory **October/November 2015**

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 **11** printed pages and **1** blank page.

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

- 8 bits for the mantissa, followed by
- 4 bits for the exponent

Two's complement form is used for both mantissa and exponent.

(a) (i) A real number is stored as the following 12-bit binary pattern:

0	1	1	0	1	0	0	0	0	0	1	1

Calculate the denary value of this number. Show your working.

.....

.....

.....

.....

.....

.....[3]

(ii) Give the normalised binary pattern for +3.5. Show your working.

.....

.....

.....

.....

.....

.....[3]

(iii) Give the normalised binary pattern for –3.5. Show your working.

.....

.....

.....

.....

.....

.....[3]

The number of bits available to represent a real number is increased to 16.

(b) (i) If the system were to use the extra 4 bits for the mantissa, state what the effect would be on the numbers that can be represented.

.....

.....[1]

(ii) If the system were to use the extra 4 bits for the exponent instead, state what the effect would be on the numbers that can be represented.

.....

.....[1]

(c) A student enters the following expression into an interpreter:

```
OUTPUT (0.1 + 0.2)
```

The student is surprised to see the following output:

0.3000000000000001

Explain why this output has occurred.

.....

.....

.....

.....

.....

.....[3]

2 In this question, you are shown pseudocode in place of a real high-level language. A compiler uses a keyword table and a symbol table. Part of the keyword table is shown below.

- Tokens for keywords are shown in hexadecimal.
- All the keyword tokens are in the range 00 to 5F.

Keyword	Token
←	01
+	02
=	03
IF	4A
THEN	4B
ENDIF	4C
ELSE	4D
FOR	4E
STEP	4F
TO	50
INPUT	51
OUTPUT	52
ENDFOR	53

Entries in the symbol table are allocated tokens. These values start from 60 (hexadecimal).

Study the following piece of code:

```
Start ← 0.1
// Output values in loop
FOR Counter ← Start TO 10
    OUTPUT Counter + Start
ENDFOR
```

(a) Complete the symbol table below to show its contents after the lexical analysis stage.

Symbol	Token	
	Value	Type
`Start`	60	Variable
`0.1`	61	Constant

[3]

(b) Each cell below represents one byte of the output from the lexical analysis stage.

Using the keyword table and your answer to **part (a)** complete the output from the lexical analysis.

60	01												

[2]

(c) The compilation process has a number of stages. The output of the lexical analysis stage forms the input to the next stage.

(i) Name this stage.

.....[1]

(ii) State **two** tasks that occur at this stage.

.....

.....

.....

.....[2]

(d) The final stage of compilation is optimisation. There are a number of reasons for performing optimisation. One reason is to produce code that minimises the amount of memory used.

(i) State another reason for the optimisation of code.

.....[1]

(ii) What could a compiler do to optimise the following expression?

```
A ← B + 2 * 6
```

.....

.....

.....[1]

[Turn over

(iii) These lines of code are to be compiled:

```
X ← A + B
Y ← A + B + C
```

Following the syntax analysis stage, object code is generated. The equivalent code, in assembly language, is shown below:

```
LDD 436     //loads value A
ADD 437     //adds value B
STO 612     //stores result in X
LDD 436     //loads value A
ADD 437     //adds value B
ADD 438     //adds value C
STO 613     //stores result in Y
```

(iv) Rewrite the equivalent code, given above, following optimisation.

.....

.....

.....

.....

.....

.....[3]

3 **(a)** Explain what is meant by circuit switching.

.....

.....

.....

.....[2]

(b) There are many applications in which digital data are transferred across a network. Video conferencing is one of these.

For this application, circuit switching is preferable to the use of packet switching.

Explain why this is so.

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....[6]

(c) A web page is transferred from a web server to a home computer using the Internet.

Explain how the web page is transferred using packet switching.

.....

.....

.....

.....

.....

.....[3]

4 **(a)** Four descriptions and four types of computer architecture are shown below.

Draw a line to connect each description to the appropriate type of computer architecture.

Description	Computer architecture
A computer that does not have the ability for parallel processing.	SIMD
The processor has several ALUs. Each ALU executes the same instruction but on different data.	MISD
There are several processors. Each processor executes different instructions drawn from a common pool. Each processor operates on different data drawn from a common pool.	SISD
There is only one processor executing one set of instructions on a single set of data.	MIMD

[4]

(b) In a massively parallel computer explain what is meant by:

(i) Massive .....

.....

.....[1]

(ii) Parallel .....

.....

.....[1]

(c) There are both hardware and software issues that have to be considered for parallel processing to succeed.

Describe **one** hardware and **one** software issue.

Hardware .....

.....

.....

.....

Software .....

.....

.....

.....[4]

5 (a) (i) Complete the Boolean function that corresponds to the following truth table.

INPUT			OUTPUT
P	**Q**	**R**	**Z**
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	0
1	0	0	1
1	0	1	1
1	1	0	0
1	1	1	1

$Z = P \cdot \overline{Q} \cdot \overline{R} +$ .....[3]

The part to the right of the equals sign is known as the sum-of-products.

(ii) For the truth table above complete the Karnaugh Map (K-map).

		PQ			
		00	**01**	**11**	**10**
R	**0**				
	1				

[1]

The K-map can be used to simplify the function in **part(a)(i)**.

(iii) Draw loop(s) around appropriate groups of 1's to produce an optimal sum-of-products. [2]

(iv) Using your answer to **part (a)(iii)**, write the simplified sum-of-products Boolean function.

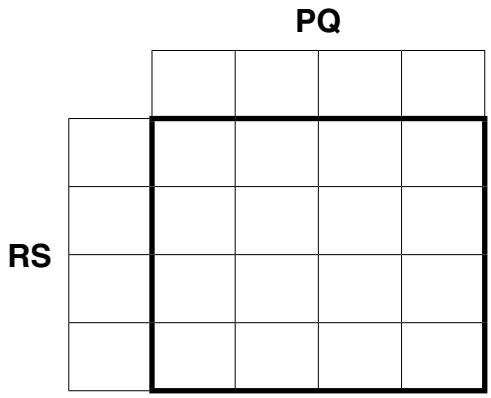
Z = ...[1]

[Turn over

(b) The truth table for a logic circuit with four inputs is given below:

INPUT				OUTPUT
P	**Q**	**R**	**S**	**Z**
0	0	0	0	0
0	0	0	1	0
0	0	1	0	0
0	0	1	1	0
0	1	0	0	0
0	1	0	1	1
0	1	1	0	0
0	1	1	1	1
1	0	0	0	0
1	0	0	1	1
1	0	1	0	0
1	0	1	1	0
1	1	0	0	0
1	1	0	1	1
1	1	1	0	0
1	1	1	1	1

(i) Complete the K-map corresponding to the truth table above.



[4]

(ii) Draw loop(s) around appropriate groups of 1's to produce an optimal sum-of-products. [2]

(iii) Using your answer to **part (b)(ii)**, write the simplified sum-of-products Boolean function.

Z = ...[2]

6 A number of processes are being executed in a computer.

A process can be in one of three states: running, ready or blocked.

(a) For each of the following, the process is moved from the first state to the second state. Describe the conditions that cause each of the following changes of state of a process:

From blocked to ready .....

.....

.....

.....

From running to ready .....

.....

.....

.....[4]

(b) Explain why a process cannot move directly from the ready state to the blocked state.

.....

.....

.....

.....

.....

.....[3]

(c) A process in the running state can change its state to something which is neither the ready state nor the blocked state.

(i) Name this state.

.....[1]

(ii) Identify when a process would enter this state.

.....[1]

(d) Explain the role of the low-level scheduler in a multiprogramming operating system.

.....

.....

.....

.....[2]

BLANK PAGE