UNIVERSITY OF BRISTOL

Practice Paper 1A (practical section)

School of Computer Science

First In-class Test for the Degree of Master of Science in Computer Science (conversion)

COMSM1302
Overview of Computer Architecture

TIME ALLOWED:
2 Hours

This section contains **four** questions.
All questions will be marked.
The maximum for this section is **50 marks**.

Other Instructions

1. **You are not permitted to visit any web sites other than those in the "In-class test 1" sidebar of the unit Blackboard page.**
2. **You are permitted to use the Calculator app in Basic or Advanced mode only, or the KCalc app in Simple or Science mode only.**
3. **You are not permitted the use of physical calculators.**
4. **You are permitted the use of the exam version of Logisim linked on the unit Blackboard page, and no other versions of Logisim.**
5. **This is not a real exam paper.**

TURN OVER ONLY WHEN TOLD TO BEGIN WORK

For each question, build the circuit within the Logisim skeleton file provided. Each question subcircuit will already contain every input and output pin required. The components permitted to build the circuit are listed at the end of each question — do **not** use any components other than those explicitly permitted and those in the "Wiring" folder (e.g. constants and splitters). You may create further subcircuits, but they should only consist of permitted components for that question. Your final design (using any extra subcircuits you've built) should be built in the given subcircuit e.g. your design for Question 1 should be built with the Q1 subcircuit.

Full marks will be given to circuits that display the correct behaviour while only using permitted components. You are not required to use all permitted components. Partial marks will be available. Complexity and neatness will not be marked, unless otherwise stated.

In the actual test, you should submit your completed Logisim file to the "In-Class Test 1 (Practical component) submission point" on Blackboard. Multiple submissions are allowed, but only the last one will be marked.

Question 1 (10 marks)

Create a circuit that implements the Boolean expression:

$$(A \wedge B \wedge C) \vee \neg(\neg B \vee C) \vee \neg A.$$

Permitted components: 2-input AND gates, 2-input OR gates, and NOT gates.

Question 2 (10 marks)

Create a circuit that implements the below truth table, using at most 7 logic gates. Partial credit will be given for any solution that implements the truth table with more than 7 gates, depending on the number of gates used.

A	B	C	D	Out
0	0	0	0	1
0	0	0	1	1
0	0	1	0	1
0	0	1	1	1
0	1	0	0	1
0	1	0	1	1
0	1	1	0	1
0	1	1	1	1
1	0	0	0	0
1	0	0	1	0
1	0	1	0	1
1	0	1	1	1
1	1	0	0	0
1	1	0	1	0
1	1	1	0	1
1	1	1	1	0

Permitted components: Any logic gates with 2 or fewer inputs.

Question 3 (15 marks)

Create a circuit that implements a rising-edge D flip-flop with inputs D and CLK and outputs Q and Q'.

Permitted components: 2-input NAND gates.

Question 4 (15 marks)

The Logisim skeleton subcircuit for this question contains a pre-built RAM component with data RAM[0], ..., RAM[63], a button labelled input, and a 8-bit output. Create a circuit which behaves as follows.

While the input button is pressed: The output should become RAM[1] at the first rising clock edge after the input button is pressed. At the next rising clock edge, the output should become RAM[2]. It should continue cycling through RAM in this way until reaching RAM[31]. On the next rising clock edge, the output should become RAM[0]. On the next rising clock edge, the output should become RAM[1] again and the cycle should repeat.

While the input button is not pressed: The output should become RAM[0] at the first rising clock edge and stay at this value.

Permitted components: All components are permitted.