LCO Summer Revision Questions

May 2018

Contents

Topics

This list is not exhaustive and it appears in no particular order.

The written summer exam is (as your final exam will be) *calculator* but will **not** include Python or C#.

In terms of the Computing text book, you should expect questions from Chapters 13–36 inclusive. For the Discrete Maths, expect questions from Chapters 1–4.

The syllabus will be similar to the AQA AS Level (7516)[1] and elements drawn from the OCR Further Mathematics A (H235/Y534)[2]

Computing — AQA 7516

Data representation
: Number systems; Bits, bytes and binary; Binary arithmetic and the representation of fractions; Bitmapped graphics; Digital representation of sound; Compression, Hashing, Encryption; Floating point form

Computer Systems
: Hardware and software; Role of an operating system; Programming language classification; Programming language translators; Logic Gates; Boolean algebra; Adders and D-type flip flops

Computer organisation and architecture
: Internal computer hardware; The processor; The processor instruction set; Assembly language; Input-output devices; Secondary storage devices

Communication and networking
: Communication methods; Network topology; Client-server and peer-to-peer; Wireless networking, CSMA and SSID; Communication and privacy; The challenges of the digital age

Discrete Mathematics — OCR Y534

Mathematical Preliminaries
: counting methods; ${}^nC_r$; pigeonhole

Graphs and networks
: terminology, Euleran, Hamiltonian, planarity, digraphs

Algorithms
: measuring complexity; sorting algorithms; next-fit, first fit, first-fit descending binpacking problems.

Network algorithms
: least-weight paths; minimum spanning trees; Prim's, Kruskal's algorithms.

References

[1] AQA Computer Science (7516) AS Level Specification *http://www.aqa.org.uk/subjects/computer-science-and-it/as-and-a-level/computer-science-7516-7517/subject-content-as* AQA, 2015.

[2] OCR Further Mathematics A *http://www.ocr.org.uk/qualifications/as-a-level-gce/as-a-level-gce-further-mathematics-a-h235-h245-from-2017/specification-at-a-glance/#as-level* OCR, 2017.

1 Binary, Hex, Numbers

1. Explain how you would turn a positive value into a two's complement number storing its negative equivalent. Use the denary values 34 and -34 and 8 bit binary values as an example.
2. The binary bit pattern `11000010` ...
   (a) What is the value of this number if it is an *unsigned binary integer*?
   (b) What is the value of this number if it is an *unsigned binary fixed point number* with 4 bits before and 4 bits after the binary point?
   (c) What is the hex equivalent of this value?
   (d) What is the denary equivalent of this value if it represents a *two's complement binary integer*?
   (e) If this number has passed parity, what style of parity is being used?
   (f) What character is being represented by this number if it represents a 7-bit ASCII code with the most significant bit being used as the parity bit?
3. Represent the following denary numbers in binary using 8 bits:
   (a) 234
   (b) 197
   (c) 57
4. How many denary values can be represented using 16 bits?
5. What is the hexadecimal equivalent of the following denary numbers:
   (a) 289
   (b) 154
   (c) 57005
6. Showing your working, multiply the binary value `1101` by `101`.
7. Convert the following decimal numbers to fixed point binary:
   (a) 0.625
   (b) 0.3
   (c) 0.734375
8. Define the terms mantissa and exponent and their role in converting a binary floating point number into denary.
9. Convert the following decimal numbers to 8 bit two's complement, fixed point binary in which 3 bits are allocated for the fractional part:
   (a) -1.75
   (b) -6.5
   (c) -14.125
10. What is the maximum and minimum value — in both binary and denary — which can be stored in 8 bit signed, fixed point binary numbers in which 3 bits are allocated for the fractional part?
11. Convert the following floating point binary numbers which store the mantissa in 4 bits in two's complement form and the exponent in 4 bits, two's complement form, into denary. The binary point is between the most significant bit and the next ost significant bit of the mantissa.
    (a) 01100101
    (b) 10010100
    (c) 01001001
    (d) 01111101

12. What does it mean for a fixed point binary number (e.g. a mantissa in an exponent) to be normalised?
13. Explain the terms underflow and overflow when used with floating points numbers.
14. With a suitable example, explain the difference between absolute and relative errors with floating point representation.

2 Information Coding

1. How many different ASCII codes are there?
2. What is the range of hex values for
   (a) upper case letters?
   (b) lower case letters?
   (c) numbers?
3. Why has Unicode gained traction in preference to ASCII?
4. What would the parity bit be for the ASCII code for the letter 'A' to ensure that it passed odd parity?
5. Under majority voting what bits would be sent to communicate the ASCII for the number 7?
6. Define *checkdigit* and outline either the ISBN-10, ISBN-13 or Kuhn algorithm.
7. In computer graphics, what is meant by the colour depth?
8. Relate an image's resolution to PPI. Consider the Jobsian 'Retinal Display' in your answer.
9. Give 4 things that could be found in an image's metadata.
10. In a bitmapped image showing 24 bits of colour and a 1024×768 image size, how many bytes would be required?
11. For what purposes would a vector graphic format be used? Why?
12. What attributes might you need to define to draw a circle?
13. Why might anti-aliasing be used on a bitmapped image after resizing?
14. Describe the MIDI format.
15. What do ADC and DAC stand for?
16. What aspects of sampling are required to estimate the file size?
17. With a bit depth of 16 bits, sampling at 44.1KHz, how many byters are required for a 60 second, single channel recording?
18. What does Nyquist's Theorem say about the relationship between sample rate and frequency?

3 Compression, Hashing and Encryption

3.1 Compression

1. Contrast lossy and lossless forms of compression.
2. Describe run length encoding and state whether it is lossy or lossless.
3. Is the storing of analogue data in digital form compression? How might such data be compressed?
4. Give two uses of
   (a) lossy compression
   (b) lossless compression
5. Give three reasons why compression might be used.
6. Video when streamed may start being pixellated and then growing progressively smoother looking. Explain why this is the case.
7. What is a CODEC?

3.2 Hashing

8. What is the purpose of a hashing algorithm?
9. With the context of hashing, define:
   (a) collision
   (b) clustering
   (c) load factor
10. Describe two alternative methods for dealing with collisions.

3.3 Encryption

11. Why might data be encrypted?
12. Define:
    (a) key
    (b) plaintext
    (c) Caesar cipher
    (d) frequency analysis
    (e) one-time pad
    (f) symmetric and asymmetric encryption
    (g) computational security
13. Modulo arithmetic
    (a) What is $21 + 15 \bmod 26$?
    (b) Today is Friday, what day will it be in 506 days' time? What was it 212 days ago?
    (c) If you head due East, turn through 130 deg, what is your new bearing?
14. For every integer b and positive integer m there is exactly one integer q and exactly one positive integer r such that
    $$b = q \times m + r$$
    , the *quotient and remainder theorem*. Find r and q when
    (a) $b = 37, m = 12$

    (b) $b = 76, m = 60$
    (c) $b = -37, m = 12$

15. Encipher a short message (¡ 50 characters) using a Caesar cipher. Swap with a colleague. Attempt to decipher their message.

16. Explain why a Vernam cipher is considered unbreakable. What conditions must be observed?

17. How would you set about breaking a non-trivial cipher?

18. How would public key crypto be used to
    (a) encrypt
    (b) authenticate

    messages?

4 Software, Operating Systems, Languages

4.1 Software

1. Contrast application software with system software.
2. Define, with example where appropriate:
   (a) translator;
   (b) compiler;
   (c) assembler;
   (d) interpreter;
   (e) library modules;
   (f) utility program;
   (g) virtual machine.

4.2 Operating Systems

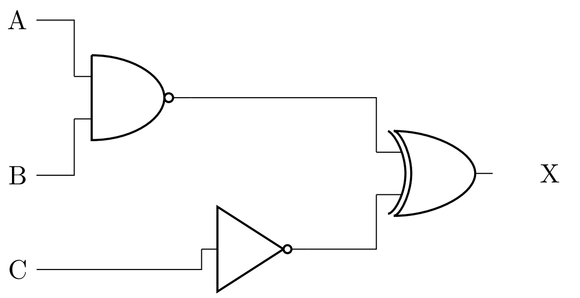
3. An operating systems is typified by several capabilities, define
   (a) resource management;
   (b) I/O management;
   (c) virtual memory;
   (d) file management;
   (e) UAC.
4. Why might a user choose one operating system over another?

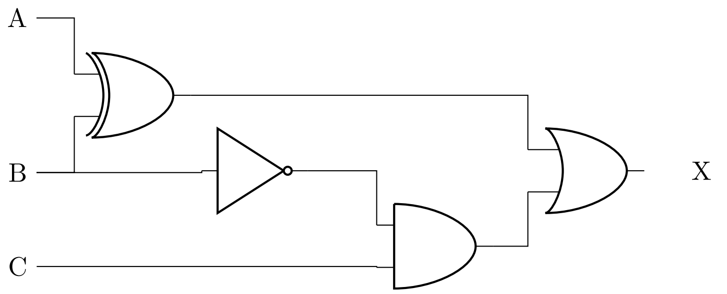
4.3 Programming Languages

5. Contrast machine code with assembly language.
6. Why might a programmer choose to code in assembly language?
7. Why might a programmer choose to code in a high level language? What might guide their choice of language?
8. With the context of high level langauges:
   (a) imperative;
   (b) functional;
   (c) object-oriented;
   (d) bytecode;
   (e) events.

5 Boolean Logic and Logic Gates

1. Enumerate the 6 logic gates you are expected to know. How would they be expressed in symbols?
2. Quote the two forms of De Morgan's Laws.
3. Complete the following identities:
   (a) $A \cdot 0 \equiv$
   (b) $A + 1 \equiv$
   (c) $A + 0 \equiv$
   (d) $A \cdot 1 \equiv$
   (e) $A \cdot \overline{A} \equiv$
   (f) $\overline{\overline{A}} \equiv$
   (g) $A \cdot A \equiv$
   (h) $A \cdot (B + C) \equiv$
4. Draw the truth table and circuit for the Boolean statement. $\overline{B} + (\overline{\overline{A} \cdot \overline{B}})$
5. Can you simplify the previous statement?
6. Simplify $(A + B) \cdot C \cdot \overline{C} + (A + \overline{A}) \cdot B$
7. Simplify $\overline{A + \overline{A} \cdot \overline{B}}$
8. Draw the circuit and truth table for the statement $Q = (A + B) \oplus \overline{C}$
9. Give Boolean statements for the following circuit diagrams:

   (a) 

   (b) 

10. (a) Draw a half adder.
    (b) Explain — with the aid of truth tables — how to combine half adders together into a full adder.
11. Draw a logic diagram, using only NAND operations, for
    (a) $A + B$
    (b) $A \cdot B$
    (c) $A \oplus B$

6 Internal Computer Architecture

1. System bus facts:
   (a) How are the number of wires on the bus better known? What is a common value for this quantity?
   (b) What are the names for the three busses?
   (c) What direction will the data travel down these three busses?
2. How much address space is directly addressable by a 32-bit address bus?
3. Within the context of main memory, define volatile.
4. What does RAM stand for; explain the importance of R in this acronym.
5. Contrast the von Neumann and Harvard architectures.
6. Give two reasons why I/O devices are handled by controllers rather than being connected directly to the processor.
7. Explain why we would need ROM, RAM, a SSD and, potentially, a magnetic HDD within the same PC.
8. On a games console, why might it take a minute to load a game?
9. Explain what is meant by the *Stored Program Concept.*
10. Give a pro and a con for programs and data being considered as the same thing in memory.
11. What is meant by the clock speed?
12. Explain what purposes are served by the CIR, PC, MBR, MAR, SR in the fetch execute cycle.
13. What are the operands and the opcodes?
14. What are the two addressing modes?
15. What might cause an interrupt to be fired?
16. Describe the effect of an ISR on the fetch execute cycle.
17. Write assembly language instructions that would perform the following pseudocode, use registers r_1 to r_n as necessary to store variables:

    if $A = 1$ **then**
        $B \leftarrow 2$
    else
        $A \leftarrow A + 1$
    end if

7 Input/Output & Secondary Storage

7.1 Digital Cameras

1. Contrast a CCD and a CMOS sensor.
2. Describe the working of the sensor: include transistors, pixels, RGB, file formats (e.g. RAW, TIFF), flash memory.
3. What is meant by pixellation?
4. Define resolution.

7.2 RFID Readers

7.3 Laser Printers

7.4 Barcode Scanners

7.5 Magnetic, Solid State & Optical Disks

5. Contrast a magnetic hard drive's method of operation with an SSD.
6. Pick a suitable secondary storage medium — or other methodology if more appropriate — for the following tasks, outlining reasons for your choice, also suggest approximate sizes for each task:
   (a) Transferring a 90 minute 4K video file from a laptop at home to a school machine;
   (b) Backing up your project documentation, as either MS Word or LaTeX;
   (c) Sending a short (e.g. less than an hour) audio file to a colleague;

8 Communication

9 Networking

10 Discrete Mathematics

10.1 Definitions

1. Define:
   (a) derangement
   (b) digraph
   (c) walk
   (d) trail
   (e) tree
   (f) hamiltonian cycle
   (g) bipartite graph
   (h) heuristic algorithm

10.2 Classification of problems

2. You need to be at school by 0837 but don't want to leave your house before 0730. Within the context of this example, can you explain the following terms:
   (a) existence
   (b) construction
   (c) enumeration
   (d) optimisation

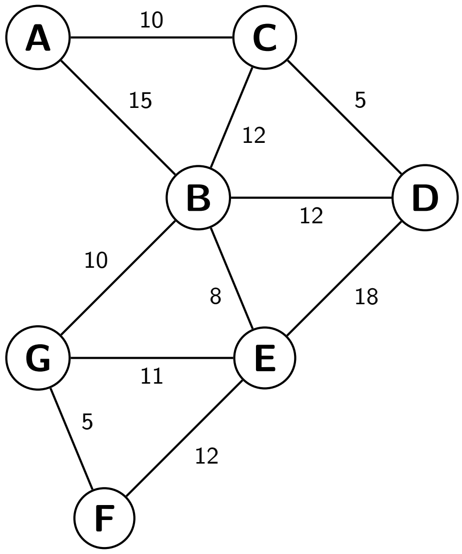
10.3 Sets and Counting

3. Three sets, A, B and C have 28, 28 and 22 members, $n(A \cup B \cup C) = 42$, $n(A \cap B) = 17$, $n(A \cap C) = 13$ and $n(B \cap C) = 15$.
   (a) What is $n(A \cap B \cap C)$?
   (b) If $n(\varepsilon) = 50$, what is $n(B')$?
4. Prove that if you pick three integers at random, a pair of them will add up to an even number.
5. In a race involving 8 athletes, how many different ways can the medals for first, second and third be awarded?
6. In a class of 12 students, how many ways can they be split into
   (a) 4 teams of 3;
   (b) 3 teams of 4?
7. How many derangements are there of the word "NOTE"?

10.4 Graphs

8. A graph has nodes with degree $2, 2, 3, 3, 4, 4$:
   (a) Draw a possible layout for this graph, the nodes should be labelled A–F.
   (b) Identify a semi-Eulerian route through your graph.
   (c) Explain whether or not your graph is simple.
9. Show that there are exactly six non-isomorphic trees on six vertices.

10. Using this graph:



(a) Apply Prim's algorithm and starting at 'A', listing the order in which the nodes should be selected. What is the weight of your minimum spanning tree?

(b) Apply Kruskal's algorithm, listing the order in which the nodes should be selected. What is the weight of your minimum spanning tree?