May 2015-2016 tbc

ECS404U **Computer Systems and Networks** Duration: $2\frac{1}{2}$ hours

SOLUTIONS AND MARKING SCHEME

YOU ARE NOT PERMITTED TO READ THE CONTENTS OF THIS QUESTION PAPER UNTIL INSTRUCTED TO DO SO BY AN INVIGILATOR.

Instructions: This paper contains FOUR questions. **Answer ALL questions.** Cross out any answers that you do not wish to be marked.

Calculators are permitted in this examination. Please state on your answer book the name and type of machine used.

Complete all rough workings in the answer book and cross through any work that is not to be assessed.

Possession of unauthorised material at any time when under examination conditions is an assessment offence and can lead to expulsion from QMUL. Check now to ensure you do not have any notes, mobile phones, smartwatches or unauthorised electronic devices on your person. If you do, raise your hand and give them to an invigilator immediately.

It is also an offence to have any writing of any kind on your person, including on your body. If you are found to have hidden unauthorised material elsewhere, including toilets and cloakrooms it will be treated as being found in your possession. Unauthorised material found on your mobile phone or other electronic device will be considered the same as being in possession of paper notes. A mobile phone that causes a disruption in the exam is also an assessment offence.

Exam papers must not be removed from the exam room.

Examiners: Dr A Alomainy and Prof E Robinson

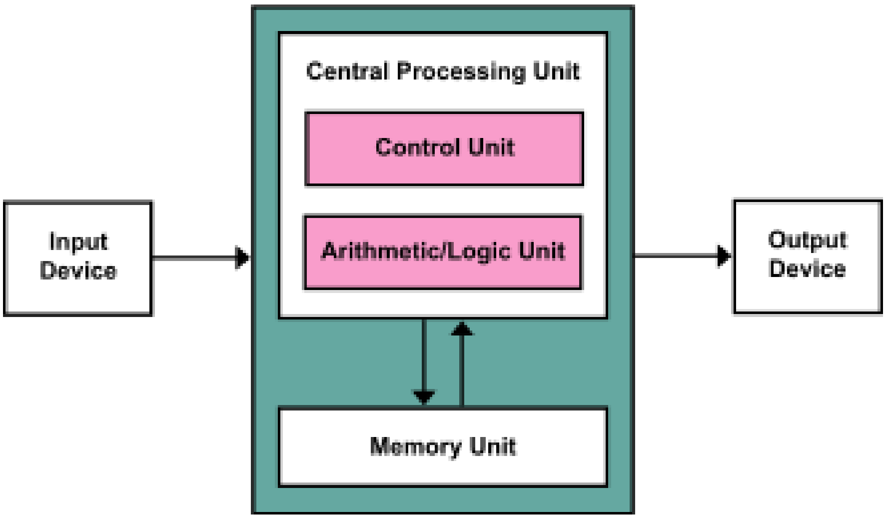
Question 1

This question is about Computer Architecture

(a) Draw an architecture diagram of a von Neumann machine and label the parts.

[5 marks]

Solution: A standard bookwork question and predictable. Expect the standard von Neumann picture with processor containing control and arithmetic/logical units, a memory and input and output.



Neatly drawn and includes all components and IO correctly related: 5

Messy but otherwise correct: 4

Notable deficiencies in structure: 3

Messy with notable deficiencies: 2

Indicates some knowledge of what is being asked: 1

(b) Give one example of a way in which the architecture of a modern PC conforms to that of a von Neumann Machine and one way in which it differs. Include a sentence or two in justification of each. **[4 marks]**

Solution: Integrating two pieces of bookwork. Students have not previously been asked this directly, though they have been asked to describe how PC and other architectures differ. There are many possible answers. An example answer is:

A modern PC conforms to the von Neumann architecture in that it includes a central processing unit that does (almost all) the computation, and this is linked by a single pathway to the rest of the machine, much of which is some form of storage (eg main memory, or disk). It differs in may significant details. One example it is not clear that a modern cpu can be divided into separate control and arithmetic/logic units, and hence the processing part does not really conform to von Neumann's picture. Another is that the memory part would certainly contain a hierarchy of devices (including main memory and disk as above).

2 marks for each with 1 mark if a reasonable point is being made, and 2 if it is made clearly with justification.

(c) Give an example of a modern machine that most people would not normally think of as a computer, but which can nevertheless be regarded as one. Justify your answer both in terms of the way it functions, and its internal architecture. **[4 marks]**

Solution: Students have discussed these examples, and therefore simply need to reproduce earlier work.

An example would be a modern television, which receives digital information (from broadcast, or over the internet if it is a smart tv), decodes it and processes it to display on the screen. Internally it contains microprocessor and memory, much like a computer or tablet.

There are not many points for this. I am looking for a reasonable example and justification.
Example: 1

Justification: 3 if well expressed, 2 if not really convincing, and 1 if very flawed but containing something of relevance.

(d) State *Moore's Law*. For maximal credit give the property that Moore originally formulated. Explain how you can observe the continuing effects of this law by reference to the processor chips used on mobile phones.

[6 marks]

Solution: Standard definition with the kind of effect that has been discussed in class. A very full answer is:
Moore's Law states that the size (in the sense of number of components) on a chip that is manufactured at minimum cost per component grows exponentially over time. Moore originally had it as doubling every year. Intel's website now has it as doubling every eighteen months.

A less full answer that would still get full marks is:
The number of transistors on a cpu doubles every eighteen months.

You can see its continuing effects in that the number of cores on a high-end mobile phone processor is currently doubling every one or two generations of phone, with eight cores now being available.

4 marks for Moore's Law. 3 if not well expressed. 2 if it is unclear that the growth is exponential or that it deals with numbers of components/transistors. 1 if both those problems.

2 marks for the continuing effects. 2 if any reasonable answer clearly expressed. 1 if a reasonable attempt, but either inaccurate or unclear.

(e) Define the concepts of:

(i) latency

(ii) bandwidth

For each of these give the units in which it is measured and an example of a communication situation where it is the key limiting factor. **[6 marks]**

Solution: Definitions and units are standard bookwork:

Latency: the time between a request and the receipt of the start of the response, measured in units of time such as milliseconds.
Bandwidth: the rate at which information can be communicated over a network expressed in units such as bits per second.

An example where latency is where a program is trying to access information that is widely distributed across a traditional magnetic hard disk. An example where bandwidth is critical would be trying to watch video over a 3G mobile phone network.

One mark each component (definition, unit, example).

Question 2

This question is about forms of digital representation.

(a) This part of the question is about spotting a representation in the wild.
A new computer game you have bought comes with a key: “97af e4b3 0651 3cd2”. You are told to omit the spaces when you type it in. What representation do you think is being used? How many bits are represented? And what bit sequence does it represent? **[4 marks]**

> **Solution:** Students have done this task, but not been asked to recognise it. This is a textual way of representing a 64-bit bit sequence using hexadecimal. The sequence represented is: 1001 0111 1010 1111 1110 0100 1011 0011 0000 0110 0101 0001 0011 1100 1101 0010.
>
> One mark for getting this is hex and one for 64 bits. 2 marks for the sequence, with one if clear understanding shown but with errors.

(b) This part of the question is about the 2’s complement representation of signed integers. It uses 8-bit 2’s complement as an example.

(i) What is the largest positive number that can be represented in 8-bit 2’s complement, and what is its representation?

(ii) What is the smallest negative (i.e. most negative) number that can be represented and what is its representation?

(iii) What number is represented by 10010101? Explain your reasoning.

[5 marks]

> **Solution:** This should be familiar material, but the question is in a slightly different form to those the students are used to.
>
> (i) 127 represented by 01111111.
>
> (ii) -128 represented by 10000000
>
> (iii) There are many ways of answering this. One way is that the initial 1 represents -128. The remaining 0010101 represent $1+4+16 = 21$. The whole therefore represents $-128 + 21 = -107$.
>
> 1 each for first two parts. 3 for second. 1 for answer, and 2 for reasoning, with 2 awarded for complete answer and 1 if reasonable but flawed.

(c) This part of the question is about floating point representation. It uses the IEEE 64-bit standard as an example, but any necessary details of that representation are summarised here. The IEEE standard representation has three components. One component is the *sign* of the representation, given by a single bit representing ’+’ or ’-’. The next component is the *exponent*, consisting of 12 bits, representing the numbers from -1022 to 1023. The final component represents the *significand* and it consists of 51 bits, where $b_1 b_2 \ldots b_{51}$ represents the binary $1.b_1 b_2 \ldots b_{51}$, so that $00 \ldots 0$ represents $1.00 \ldots 0$ and $10100 \ldots 0$ represents $1.10100 \ldots 0$.

(i) If we write the largest positive number that can be represented in this format as $2^x * y$, where y is a binary expansion between 1 and 2, then what is x and what is y?

(ii) What is the smallest number greater than 0 that can be represented? Give it in the form $2^x * y$.

(iii) What happens if we add 3 to 2^{52}? What whole numbers might we get as a result? Explain why we get these results.

[6 marks]

Solution:

(i) $x = 1023$ and $y = 1.111 \ldots 1$

(ii) $x = -1022$ and $y = 1.000 \ldots 0$

(iii) 2^{52} is represented as exponent 52 and significand $1.000 \ldots 0$ (51 zeros). 3 is represented as exponent 1 and significand 1.1. We need to translate to the same exponent as 2^{52}, which would mean moving the point 51 spaces to the left, and so would be significand $0.000 \ldots 011$ where there are 50 zeros after the point and so 53 places overall. Adding these together we get $1.000 \ldots 011$ where there are 50 zeros and so 53 places overall. This then has to be rounded to 52 places and so we get either $1.000 \ldots 01$ or $1.000 \ldots 010$, which would then represent either $2^{52} + 2$ or $2^{52} + 4$.

1 mark for first part (which should be easy)

2 marks for second (expected to be harder), with 1 for each component

3 marks final part: 1 for getting right general idea, another for demonstrating ability to get at least some of the details, and the final one for a clear complete explanation.

(d) This part is about text representation.

(i) How many bits does the ASCII character set use?

(ii) The character 'A' has number 65 in the ASCII character set, similarly 'a' has number 97 and '0' (zero) has number 48. Which characters have numbers: 67, 102, 54? Explain how you know the answers.

(iii) Give the key reason why other character representation systems, such as Unicode, were developed.

[6 marks]

Solution:

(i) 7

(ii) Upper case, lower case and digits are all in sequence. So 67 represents the third upper case character 'C', 102 the sixth lower case 'f', and 54 the seventh digit '6'.

(iii) With 7 digits we can only represent 128 characters, and the we use many more than that.

(e) This part is about MP3 audio and other multimedia representation.

(i) Explain the difference between *lossy* and *non-lossy* compression?

(ii) Why does MP3 (like almost all other multimedia compression algorithms) use lossy compression?

[4 marks]

Solution:

(i) Non-lossy compression is precisely reversible. Uncompression produces exactly the same data as the original input. Lossy compression is not. Uncompression produces data that is similar to the original input, but not necessarily identical.

(ii) Lossy compression algorithms can typically produce higher compression ratios than non-lossy ones. This is important for multimedia applications where the input files are large, and there is a need for much smaller compressed files.

2 marks for the first part. 1 mark if unclear or flawed. 0 if nothing of substance.

2 marks for the second. One for getting the higher compression ratios and one for linking it to multimedia in some way.

Question 3

This question is about Assembly Language

(a) Explain the concept of *Interpreters* and how they are linked to Assembly Language and in turn Machine Coding.

[5 marks]

Solution: A standard bookwork question and predictable. Expect and explanation similar to: Programmers developed little symbolic codes which represented the binary versions of instructions and the data the acted on and they wrote programs using these codes which they translated into zeroes and ones using a computer program, called an interpreter or an assembler. These codes were still very close to the binary representation of programs so translating them into binary was easy to do, and assemblers were easy to write. The symbolic representation of machine language is called assembly language. The binary version of it is called machine code.

Explanation similar to the above that they link between programs and binary 3 marks and highlighting that assembly language means symbolic representation (1 mark) and machine code for binary (2 marks)

(b) Explain what a register is, and also where arithmetic operations are performed in a computer. Explain what the MIPS instruction

```
lw $t0 x
```

does, where $t0 is a register, and how the MIPS instruction lw differs from the instruction li. Explain also what the MIPS instruction

```
sw $t2 z
```

does, where $t2 is a register and z is a memory location. What do instructions lw and sw stand for?

[6 marks]

Solution: A register is a special piece of fast memory inside the CPU where all arithmetic operations take place. The instruction lw $t0 x loads a word from the memory space x onto the register $t0 while li loads an immediate value from the input into a register.

The instruction sw $t2 z stores a value in register $t2 in memory space z in the main RAM. sw stands for 'store word while lw stands for 'load word'.

1 mark for explaining registers and 1 mark for mentioning the arithmetic operations happen in the CPU. 1 mark for explaining each of the instructions given above. 1 mark for specifying the difference between li and lw and 1 mark for mentioning what sw and lw stand for.

(c) Explain what the MIPS instruction

```
mult $t0 $t1; mflo $t2
```

does, where $t0, $t1, and $t2 are registers. Explain whether the results obtained from this arithmetic operation using the instruction above will be the complete expected outcome available from register $t2.

[7 marks]

Solution: The instruction multiplies the two values in register $t0 and $t1 and moves the lower 32 bits of the multiplication outcome into register $t2.

As mentioned above only the 32 bits of the multiplication operation stored into the special register space LO would be moved into register $t2 and to access the higher 32 bits you will need to use instruction mfhi

2 marks for identifying it is multiplication operation and 2 marks for specifying that only the lower 32 bits will be moved to register $t2. 3 marks for explaining that only the lower 32 bits will be presented. Mentioning mfhi is a plus to compensate for any missing marks within this question.

(d) Now write a MIPS program that will load the numbers 20 and 35 into registers $t0 and $t1, respectively, and then multiplies the two numbers and returns the whole 64-bit result (both lower and higher 32 bits) into registers $t2 and $t3.

[7 marks]

Solution:

```
li $t0 20
li $t1 35
mult $t0 $t1; mflo $t2; mfhi $t3
```

1 mark for each correct loading instruction. 1 mark for the multiplication instruction and 2 marks each for mfhi and mflo

Question 4

This question is about Computer Networks

The figure below shows the Wireshark program displaying a packet it has captured (we are looking at packet No. 7 below towards the end of the top pane).

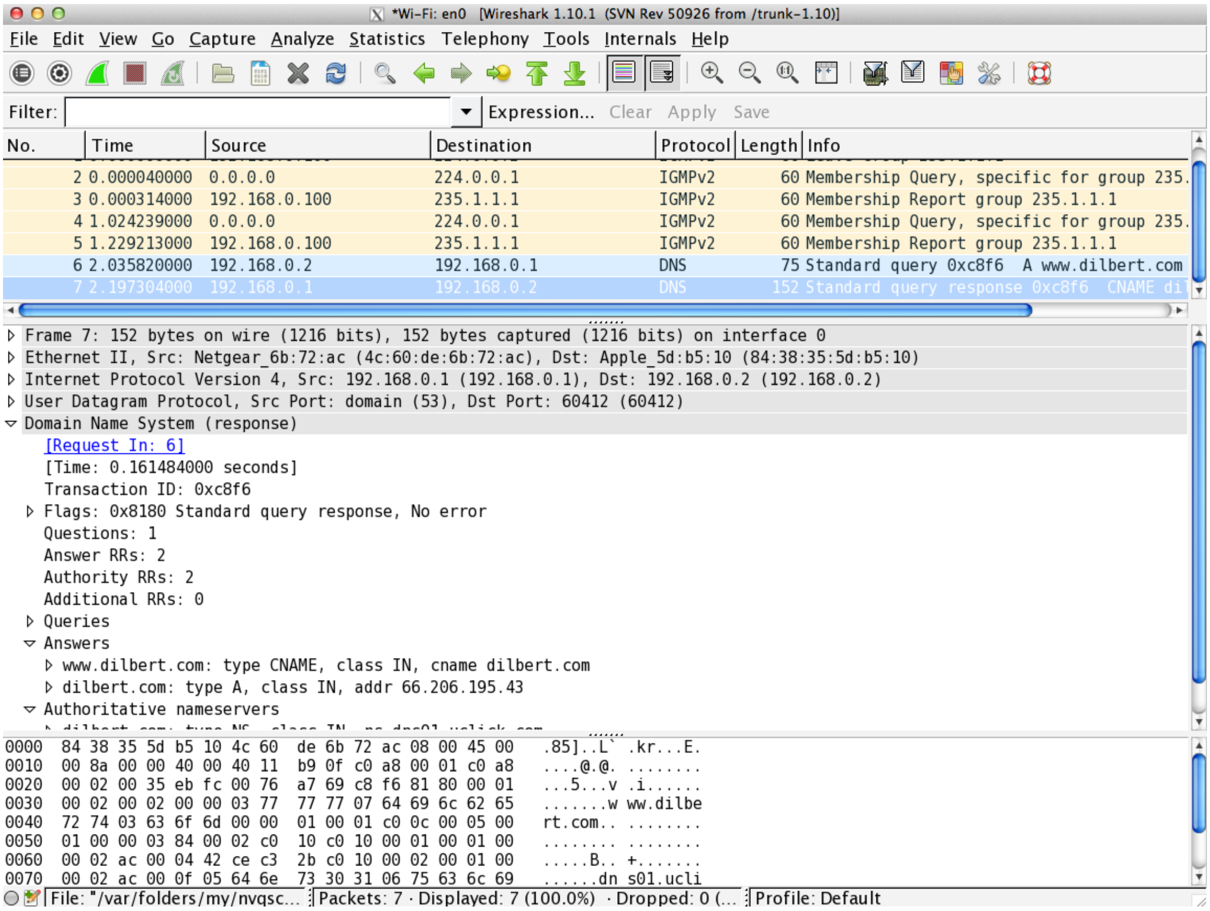


Figure 1: Wireshark display of packet 7

(a) Which protocols are being used in the following layers:

i) application

ii) transport

iii) internet

iv) link

In each case identify how you know the answer from the screenshot.

[5 marks]

Solution: Wireshark was used in the labs for this module and the studnets should be able to identify from the middle pane related to packet 7 that; Application: Domain Name System (DNS), Transport: User Dataram Protocol (UDP), Internet Protocol version 4 (IPv4) and Link: Ethernet II.

1 mark for each correct identified protocol (total of 4 marks) and also 1 mark for specifying that it is in the middle pane of the Wireshark window.

(b) Explain what the two IP addresses 192.168.0.1 to 192.168.0.2 represent in this communication and what message is being sent. How is packet (6) related to packet (7)?

[6 marks]

Solution: The first IP address is the sender (source) and the second one is the recipient (destination). The message is a response to the request of the ip address for www.dilbert.com, which it gives as 66.206.195.43. Packet 6 includes the actual query message for the ip address.

1 mark for identifying the sender and 1 mark for the destination, 2 marks for the identification it is a reponse to ip address query. 2 marks for correctly relating packet 6 to the actual request.

(c) Which ports are being used (packet 7) and what is their function?

[4 marks]

Solution: The port on the client (destination) is 60412, and the one on the server (source) is 53. The port is used by the transport layer to identify the program sending the message on the source machine (hence the program expecting a reply), and the program on the destination machine for which the message is intended.

1 mark for each port number and 2 marks for identifying it is used for transport layer communication.

(d) Packets are used in computer networking to encapsulate messages and important information for communications between different machines and ports. What are packets called for TCP (Transmission Control Protocol) and IP (Internet Protocol)? What is SMTP and what is it used for?

[4 marks]

Solution: TCP packets are called segments and IP packets are called datagrams. SMPT is Simple Mail Transfer Protocol used in mailing for sending messages and mainly used on the transmitting side.

1 mark for each correct answer above. 2 marks for defining SMPT and its function.

(e) Networks conversely work in one of two ways; circuit-based and network-based networking, Explain each type with examples and define whether modern networks are circuit or packet-based. **[6 marks]**

Solution: Circuit-based: you set up a persistent connection between two points on the network. Classic example: telephones. When you make a call there is a persistent connection between caller and receiver. Packet-based: you send information in individual dollops, each piece contained in a packet which tells the network where to deliver it. Classic example: the postal service Modern networks are mainly packet-based

1 mark each for defining each type and 1 mark for each example. 2 marks for specifying that modern networks are network-based.

End of questions