EXAM PAPERS PRACTICE

5.3 Units of information Mark Scheme

Mark schemes

Q1.

256 // 2^8;

[1]

Q2.

All marks AO1 (understanding)

	Correct Name from List
B	Visual display unit;
C	Processor;
D	Main memory;
E	Keyboard;

1 mark per correct answer

A If same response used more than once

[4]

Q3.

(a) Greater the bandwidth, the higher the bit rate // positive correlation // (directly) proportional;

Bandwidth must be at least 2wHz where w is the bit rate in bits per second;

Max 1

(b) Time delay between the moment something is initiated and the moment its effect begins

A time delay between signal being transmitted and arriving

A time taken for transmitted data to arrive at the receiver

A lag for time delay

NE delay in transmission, transmission time

1

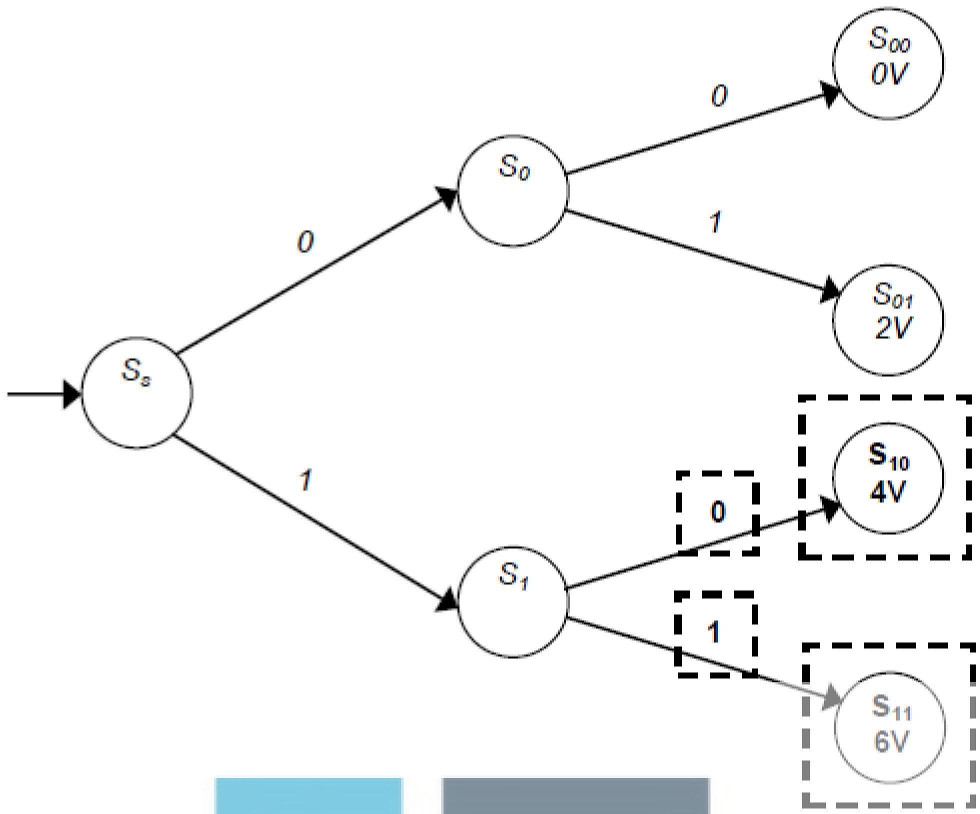
(c) Bit rate is double / twice baud rate // Baud rate is half bit rate;

A “It” is double;

A 2:1

1

(d)



1 mark for labelling a transition arrow with 0
1 mark for labelling a transition arrow with 1
1 mark for labelling a state with the value 4V and a unique state name
1 mark for labelling a state with the value 6V and a unique state name

Max 2 if the states and transition arrow labels do not correspond

Note that:

- The state names do not have to match those given here.
- The voltage values can be followed by a V, the word Volts or nothing.
- The zero and one on the transition arrows to the right of S_1 can be either way around e.g. 1 above 0 is okay.

4

[7]

Q4.

(a)

	General Purpose	Special Purpose	Bespoke
Word Processor	✔		
Payroll		✔	A ✔
Flight control software			✔

R Answers with more than one tick on a row.

3

(b) 1 – Operating System;
R OS, Operating Software

2 – Assembler / Compiler / Interpreter;
TO "assembly language"

3 – Backup / Anti virus / defragmenter / encryption / compression / archive / system profilers / application launchers;
Accept any other examples of a utility

3

[6]

Q5.

(a) (Sound/voice) recording/er // sampling/er (software) // audio capture software;
Operating system **A** OS;
Driver;
Codec;
R Microphone software
R Analogue to digital converter

Max 2

(b) (i) Number of samples/measurements taken <u>per second/unit time</u>;
Frequency/how often samples are recorded/taken;
R Rate of …"
R "Intervals at which …"

Max 1

(ii) 1000 samples/measurements per second;
1 sample/measurement per millisecond (ms);
1000 Hz / 1 KHz;
R 1000 (only)

Max 1

(c) 8 (bits);

1

(d) (i) (Sound) quality will be improved/clearer **R** Smoother // better/higher resolution //more accurate // higher fidelity;
the height of the wave will be measured more precisely/accurately;
R larger range of frequencies is possible

Max 1

(ii) The size of the sound file will increase // file uses more memory /disk space;
R 'uses more space'

1

(e) <u>0</u>110 1100;

1

(f) *All correct answers must fit the context of how the byte(s) are interpreted by the application program (not by the user of the application).*

<u>Program</u> instruction(s) // machine code;
Integer (number);
Real (number) / Floating point;
Exponent;
Mantissa;

(BUT Real/Floating Point + Exponent + Mantissa scores Max 2)
BCD (number);
R Number / denary / binary

ASCII (code);
Unicode;
EBCDIC;
Character (BUT not in addition to specific codes above) **R** Keystroke;

Address / pointer /memory reference **R** Location;
String **R** Word;
Format code // system setting / device status/signal;

A any 'data type' descriptor (e.g. Boolean) – any three data types gets but excluding any answers above;
A Colour;

Max 3

[11]

Q6.

(a) 1101 0001 0101 1010;;

1 mark for each correct byte
I leading bits

2

(b) 16;

1

[3]

Q7.

(a) 111111;

1

(b) 256/2^8;

1

(c) 255/2^8-1/ 11111111;

1

[3]

Q8.

(a) 1024 / 2^{10};
A 10000000000_2 *(10 0's)*

1

(b) (i) 1111111111111111; *(16 1's)*
A FFFF;
A 65,53<u>5</u> / 2^{16}<u>-1</u>;

1

(ii) 0000 0000 0010 0101; *accept if leading zeros not given*

1

(c) (i) 0011 0011 1011 0111;;; *accept 37 transposed*: 1011 0111 0011 0011;;;

1 mark for parity bits - one mark for each correct character code

f.t. for parity bits: if even number of 1's in each byte ;

3

(ii) Parity bit is set when character first generated;
(Parity bit is adjusted to make) number of 1's /on-bits even;
Parity bit is regenerated / the number of 1's is checked by receiver;
If parity bit does not match / if there are an odd number of 1's an error has occurred;

2

[8]

Q9.

(a) (i) 23;

1

(b) (i) 1010 0001;;

1 mark for correct ASCII code, one mark for odd parity bit (follow through)

2

(ii) 11010 00010 *OR* 01010 00011 *OR*

Allow stop bit to be 1 or 0 but stop and start bits must be different
Follow through if (i) wrong

01000 01011 *OR* 11000 01010;

Allow both ways round for transmission

1

[4]

Q10.

(a) Group/number of bits which can be addressed/transferred/manipulated as a single unit/in one go/at a time;
R processed instead of manipulated
R amount of data instead of number of bits

1

(b) 0 1 ; *(must have both values and only those values)*
R on or off on its own

1

(c) Instruction/part of program/machine code; opcode & operand;
(pure binary) integer;
Real/floating point number/fixed point number; **R** number **R** binary
R hexadecimal
(Unicode/EBCDIC/ASCII) characters; **A** text;
Pixel(s); **R** bitmap **A** part of a bitmap boolean; a set of flags;
BCD;
Address/pointer;
A operand;
A sound;

Max 3

[5]

Q11.

(a) A group of bits representing a single character / (usually) 8 bits; 1

(b) (i) A 1

(ii) 9 1

(c)

Code	Number	
–	0	given
55	7	given
51	73	
50	732	
49	7321	

1 mark for each correct entry 6

[9]

Q12.

$1024/2^{10}$

[1]

Q13.

8;

[1]

Q3.

Part (a): The relationship between bandwidth and bit rate was well understood. A small number of students failed to achieve the available mark because they simply defined the terms instead of explaining how the bit rate was determined by the bandwidth.

Part (b): There were many good responses given to this question part. Credit was awarded to responses that were either given in the context of data communication or were generic definitions of latency. In context, latency is the time delay between when a signal is transmitted and when it is received. Some students clearly knew that latency related to time, but gave responses that lacked technical accuracy, such as, “the time delay between data being sent and a reply being received,” or, “the time taken to transmit.”

Part (c): The correct relationship in the example was that the bit rate was double the baud rate. Somewhat disappointingly, just under half of the students recognised this, with a small number stating the relationship the wrong way around.

Part (d): The diagram of the Moore machine was very well completed, with the vast majority of students achieving all four marks. A small number labelled the states but forgot to label the transitions.

Q4.

Part (a) was very well answered with the majority of candidates correctly identifying the category to which each software program belonged. Candidates who dropped marks tended to put Flight Control Software down as special purpose rather than the correct answer of bespoke.

Part (b) was generally well answered. Common mistakes were answering, ‘operating software,’ rather than the correct answer of, ‘operating system.’ Candidates need to be aware of the correct term. Some candidates failed to understand the question and used the labels from part (a) of ‘general purpose’, ‘special purpose’ and ‘bespoke’.

Q5.

This question framework was different to that seen on previous papers but candidates generally answered well and were able to relate the diagram given to their basic definitions.

(a) The most popular answers were 'sound recording software' and 'the operating system'.

(b) (i) Many answers simply rearranged the word in the question stem e.g. 'rate and which samples are taken’ and so failed to score.

(ii) This was poorly answered, often when the candidate had been unable to write a worthy answer for (i).

(c) Well answered.

(d) (i) Generally candidates scored the one mark. Any answer which suggested that a more ‘faithful’ recording was obtained was given credit. However ‘smoother’ sound suggested more samples would be required and so did not score.

(ii) Again, generally well answered. Wrong answers included that it would slow down the processing time or (worst) take more time to sample.

(f) There is a statement at the start of the mark scheme for this question which is indicative of what was expected. Some candidates carried forward answers from a previous similar question which, because of their different context, were deemed unacceptable and included ‘part of a word processed file, etc'. The key discriminator was how the bytes(s) would be interpreted by the processor or application software – not by the user sat in front of the application. The computer scientist who wrote typically, ‘binary integer, memory address, ASCII character code,’ pocketed three very accessible marks, and quickly moved on.

Q6.

(a) Most candidates gained full marks for converting the hexadecimal digits to binary.

(b) However, many candidates could not see that the address bus would require 16 lines to convey the resulting 16-bit number.

Q7.

It was pleasing to see that most candidates did very well on this question. Part (a) was almost universally correct. Although most candidates gained full marks for parts (b) and (c) there were some candidates who had the answers 256 and 255 reversed.

Q8.

A significant number of candidates do not know that 1024 bytes make 1 Kilobyte. Even fewer candidates could state correctly that the largest pure binary integer that can be stored in 2 bytes is 65,535 (or 1111 1111 1111 1111 1111). Incorrect responses ranged from as low as 3 to many thousands.

The bit pattern asked for was largely well answered, but candidates should be made aware that leading zeros are required when bit patterns to a specified length are asked for. The whole purpose of binary is that only two states, 0 and 1, can exist.

The concept of parity checking eludes many candidates. Few could explain how a computer system would use a parity bit. The parity bit is set when the character is first generated, by adjusting the parity bit to make the number of 1s even (for even parity). Then the parity is checked at the receiving end and if the parity is now odd an error has occurred. The question clearly stated that in the given computer system the parity bit was the most significant bit of each byte. However, many candidates only looked at the parity across the whole 16 bits. Many candidates were not able to translate the characters 3 and 7 into ASCII codes with the code ranges for digits explicitly provided by the question.

Q9.

Nearly all candidates obtained some marks for this question.

(a) Almost all candidates could convert from ‘Pure Binary’.

(b) Nearly all candidates gave the correct bit pattern for 33 but a large number failed to get the parity bit correct (the question setter had ensured that the parity bit had to be set to ‘1’). In part (ii) few candidates realised that the start and stop bits needed to be different and in some cases these were simply left blank. When a binary pattern is asked for, all places must be filled in with either a 0 or a 1. The parity bit must not change; just because start and stop bits are added. These will be stripped off before parity checks are performed at the receiving end.

Q10.

(a) This question was not very well answered by the majority of candidates. A correct response was that a word is the number of bits which can be addressed or transferred as a single unit. Candidates need to understand that this is not necessarily the same as the number of bits which can be processed at the same time.

(b) Surprisingly many answers were NOT 0 and 1 but 1/8th of a byte or 1, 2, 4,8 16 etc.

(c) This question did not seem to be well understood by the majority of candidates even though a similar question appeared in January 2001 for CPT1. Candidates scored full marks if they appreciated that a 32-bit word could take 4 ASCII characters or 2 UNICODE characters, an integer or a real, an instruction or an address. 32 bits are not enough to store a bit-map, so candidates were expected to suggest part of a bit-map or pixel(s). A few candidates noticed that 32 bits was just enough to take an IP address.

Q11.

(a) Most candidates knew that a byte was 8 bits. Some quoted January's paper again and (wrongly) stated that a kilobyte was 1000 bytes.

(b) The majority of candidates correctly converted the bit patterns to 'A' and '9', but a significant minority quoted the ranges given in the question.

(c) Most candidates who attempted the dry run correctly calculated the codes, but only the better candidates noticed that Number grew in size with each digit processed.

Q12.

The popular wrong answer was 1000. It is important that candidates know the exact values of quantities based on the base 2 numbering systems. Some of the better candidates answered"1000, or 1024 to be precise".

Q13.

Many candidates successfully answered that a byte represents 8 bits. This has been the accepted interpretation for a very long time now going back to the days of the PDP1 1. Incorrect answers ranged from 128 to 10^8