

Answers to Student's Book

Exam-style questions and sample answers have been written by the authors. References to assessment and/or assessment preparation are the publisher's interpretation of the syllabus requirements and may not fully reflect the approach of Cambridge Assessment International Education. Cambridge International recommends that teachers consider using a range of teaching and learning resources in preparing learners for assessment, based on their own professional judgement of their students' needs.

Cambridge Assessment International Education bears no responsibility for the example answers to questions taken from its past question papers which are contained in this publication.

The questions, example answers, marks awarded and/or comments that appear in this digital material were written by the author(s). In examination, the way marks would be awarded to answers like these may be different.

2 Data transmission

Answers to activities

2.1

1 Parts i and ii

- picture and sound out of synchronisation
 - packets arriving at different times
- pauses in the video
 - time delay in assembling packets
- degraded video and sound
 - caused by competing traffic on the communications link
- possible 'drop out'
 - packets take different routes, so some may get lost

2 • web page divided into data packets

- each packet has an IP address of the destination
- routers check header against IP address values to determine next router to send data packet
- hop value checked to see if it is zero
- destination computer reassembles data packets to build up the web page

3 a This takes all the 1-values in the payload and adds them together to generate a hex number.

- this hex value is stored in the trailer
- receiving computer carries out the same calculation on received payload
- if values match, data packets has been received error-free
- if values do not match, data packet needs to be re-sent

b 29 1-bits $\Rightarrow$ 1D in hex. If recalculated value doesn't equal 1D_{hex} then it needs to be re-sent.

- ##### 4 a
- packets keep 'bouncing around' from router to router and never reach the destination
 - use hop numbers; if hop number = 0 then the packet is deleted and is therefore lost

- b**
 - apply a hop number ...
 - ... each time a data packet reaches a router, the hop number is decreased by 1
 - ... once the hop number = 0, and the destination hasn't been reached, then the data packet is deleted
- c**
 - each packet takes a different route ...
 - ... therefore if a packet is intercepted, hacker still won't have the whole message

2.2

- 1 i** data sent one bit at a time over a single channel in both directions, but **not** at the same time
- ii** several bits of data sent down multiple channels in both directions, at the same time
- iii** data sent one bit at a time over a single channel in one direction only
- 2 i** serial, simplex data transmission
- ii** parallel, simplex data transmission
- iii** parallel, full-duplex data transmission
- iv** serial, full-duplex data transmission
- v** parallel, simplex data transmission

2.3

Statement	True	False
Packets have a header which contains the IP address of the sender and the receiver	✓	
Packets don't require any form of error checking		✓
USBs use a protocol that allows for error-free data transmission between device and computer	✓	
Serial data transmission suffers from data skewing		✓
The longest cable length supported by USB is 5 metres or less	✓	
Simplex data transmission occurs when data is transmitted one bit at a time		✓
Full-duplex data transmission involves sending 8 bits of data at a time		✓
USB uses serial data transfer	✓	
Packet switching prevents loss of any data packets		✓
USB connections can transfer data using half-duplex or full-duplex	✓	

2.4

Parity bits: 1, 0, 1, 1, 0, 0, 1, 0, 1, 0

2.5

- 1** no error, error, error, no error, error, error, no error, no error, no error, no error

- 2 It isn't possible to determine where the error occurred.

2.6

- 1 error occurred in: byte 4 (four 1-bits)
and in: bit 6 (four 1-bits)
byte 4 should be: 0 1 1 0 1 1 1 0
- 2 error occurred in: byte 8 (three 1-bits)
and in: bit 7 (three 1-bits)
byte 8 should be: 0 1 0 1 0 0 1 1

2.7

- 1 $9 + 8 + 5 + 0 + 5 + 5 = 32$
 $3 \times (7 + 1 + 1 + 4 + 7 + 9) = 3 \times 29 = 87$
 $(32 + 87)/10 = 119/10 = 11$ remainder 9
 $10 - 9 = 1$, the check digit
- 2 i ISBN 13: $2 + 3 + 1 + 0 + 0 + 2 = 8$
 $3 \times (1 + 1 + 1 + 0 + 4 + 8) = 3 \times 15 = 45$
 $(8 + 45)/10 = 53/10 = 5$ remainder 3
 $10 - 3 = 7$, the check digit
Modulo-11: $\text{sum} = (13 \times 2) + (12 \times 1) + (11 \times 3) + (10 \times 1) + (9 \times 1) + (8 \times 1) + (7 \times 0)$
 $+ (6 \times 0) + (5 \times 0) + (4 \times 4) + (3 \times 2) + (2 \times 8)$
 $\Rightarrow 26 + 12 + 33 + 10 + 9 + 8 + 0 + 0 + 0 + 0 + 16 + 6 + 16 = 136$
 $136/11 = 12$ remainder 4
 $11 - 4 = 7$, the check digit
- ii ISBN 13: $9 + 9 + 1 + 1 + 3 + 4 = 27$
 $3 \times (0 + 8 + 2 + 2 + 5 + 4) = 3 \times 21 = 63$
 $(27 + 63)/10 = 90/10 = 9$ remainder 0
check digit = 0
Modulo-11: $\text{sum} = (13 \times 9) + (12 \times 0) + (11 \times 9) + (10 \times 8) + (9 \times 1) + (8 \times 2) + (7 \times 1)$
 $+ (6 \times 2) + (5 \times 3) + (4 \times 5) + (3 \times 4) + (2 \times 4)$
 $\Rightarrow 117 + 0 + 99 + 80 + 9 + 16 + 7 + 12 + 15 + 20 + 12 + 8 = 395$
 $395/11 = 35$ remainder 10
 $11 - 10 = 1$, the check digit

2.8

- 1 Tom would need to send Jane his public key.
- 2 Two keys are required; one of the keys is only known to the recipient of the document (i.e. the private key).

2.9

- a E – text following an encryption algorithm
- b B – the receiver
- c B – plaintext
- d E – there is a security problem when transmitting the encryption key
- e A – symmetric key encryption
- f E – check digit
- g A – automatic repeat request
- h B – parity check
- i D – checksum
- j A – echo check

Answers to exam-style questions

Answers to questions 7 and 8 in this section are example answers for past paper questions. The details of the relevant past paper can be found with the corresponding exam-style question in the Student's Book.

- 1 a i Data is returned to sender. Sender's computer compares the two sets of data. If any errors found, then the data is re-sent.

This doesn't make it a very suitable method for this application.

- ii ARQ uses acknowledgement and timeout. When a receiving device detects an error following data transmission, it asks for the data to be re-sent. If there are no errors, a positive acknowledgement is sent to the sender.

The sending device will re-send data if:

- it receives a request to re-send data
- a timeout has occurred.

The whole process is continuous until all data packets received are error-free or until the timeout is reached.

- b Three differences from:

Checksum

- calculated from the block of data
- sent with block of data
- re-calculated at the receiving station
- if different, an error has occurred
- used to check errors following data transmission

Check digit

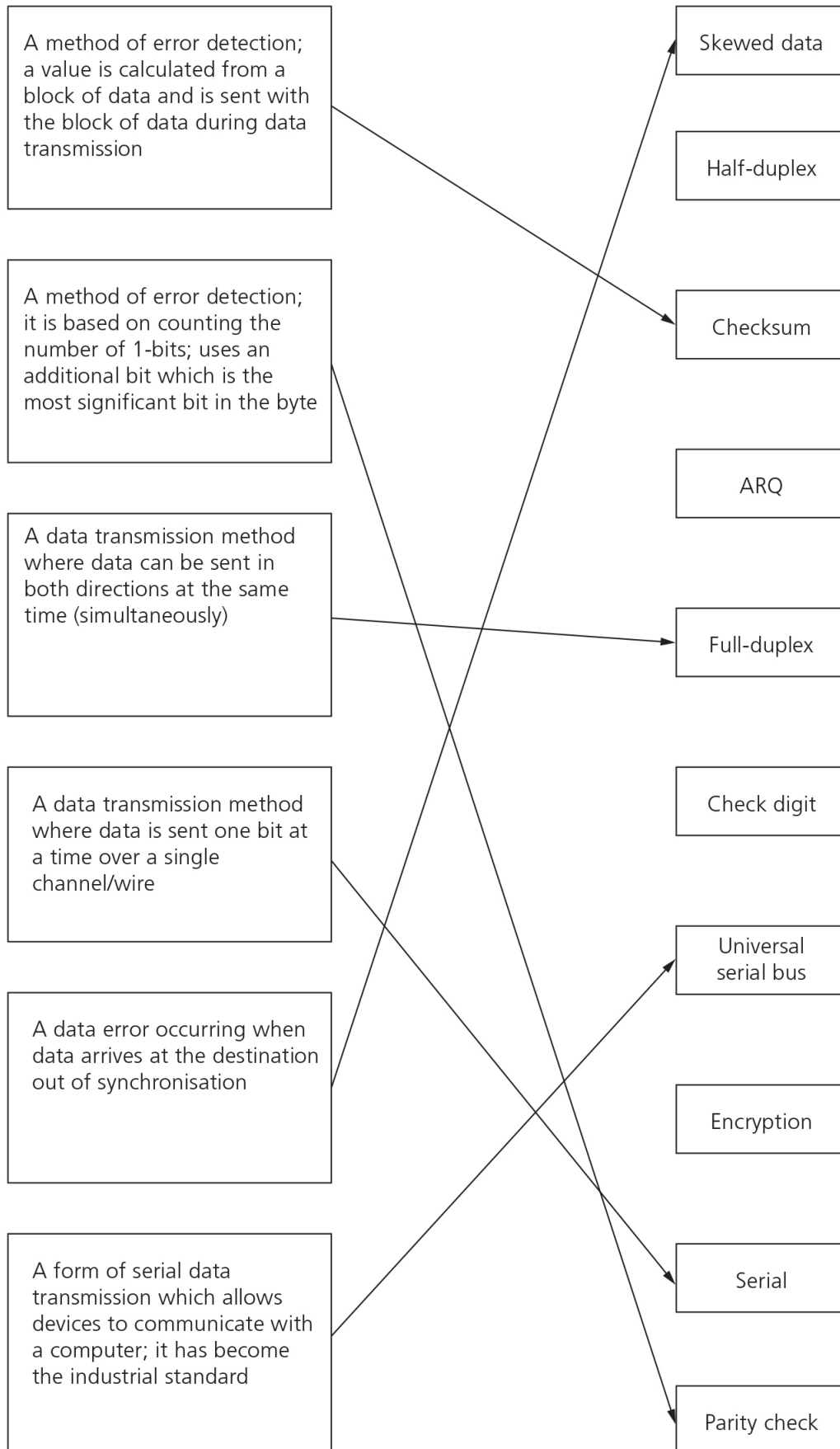
- final digit included in the code
- calculated from the other digits in the code
- used to check for errors during data entry
- detects errors such as transposition, missing/extra digits, and so on

- 2 i Packet switching is a method of data transmission where a message is broken into many data packets, which can then be sent along different pathways independently of each other.
- ii Cyclic redundancy check is an error-checking method which adds up all the 1-bits in the data packet payload and stores this hex number in the trailer; the same calculation is done at the receiving station.
- iii Data skewing occurs when data arrives at the destination with the bits no longer synchronised; this usually occurs during parallel data transmission.
- iv Universal serial bus is a type of serial data transmission which has become the industry standard for connecting computers to devices using the USB port.
- v Parity bit (0 or 1) is added to a byte of data in the most significant bit position (left-most bit); this ensures that the byte follows the correct even parity or odd parity protocol.

3 Description	Checksum ✓	Parity check ✓	ARQ ✓
Extra bit sent with each byte of data		✓	
Makes use of timeout and acknowledgement			✓
If an error is found, a request is made to re-send the data			✓
Check on whether a data packet has been changed following transmission	✓		✓
Re-calculation made on any additional data values sent to the recipient	✓	✓	
Data is transmitted in blocks or packets	✓	(✓)	✓
A method that can determine which bit in a data stream has been changed		✓	
Additional value sent at the end of a block of data to be used to check if any data transmission errors occurred	✓		

- 4 a Sequence: iii, i, ii, iv
- b Sequence: v, ii, iii, iv, i
- c Sequence: ii, iii, i, iv, v
- d Sequence: vi, v, ii, iv, iii, i
- 5 a Symmetric encryption occurs when the same encryption key is used to encrypt and decrypt a message.
- b Asymmetric encryption uses public keys (known to everyone) and private keys (known to a single user only); the recipient sends a public key to each sender who uses it to encrypt the message. The encrypted message is then decrypted using a matching private key at the recipient's computer.
- c If data is intercepted it cannot be understood by a hacker/eavesdropper without the correct decryption key.

6



7 a i

Method 1	Tick ✓	Method 2	Tick ✓
Serial	✓	Simplex	
Parallel		Half-duplex	
		Full-duplex	✓

ii Serial has lower/less interference.

Serial is more reliable over long distances.

In serial, bits won't be skewed/out of synchronisation.

In serial, it is easier to collate bits together again after transmission.

Full-duplex transmits data in both directions at the same time.

Simplex and half-duplex won't allow read and write operations at the same time.

b Any two from:

Checksum:

- value calculated from the data block
- this value is transmitted along with the data
- checksum recalculated after transmission
- if the values match, data has been transmitted error-free

Parity check:

- parity bit is transmitted with each byte of data
- odd or even parity protocol is being used
- number of 1-bits are added together to determine if parity bit is 0 or 1
- byte checked at destination to see if it has the correct parity protocol (odd or even)
- if the parity has changed, an error has occurred

ARQ:

- uses acknowledgement and timeout
- when device detects an error in the data transmission, it requests the packet to be re-sent; if no error, a positive acknowledgement is sent
- sending device re-sends packet after request to re-send is made/timeout reached
- process is continuous until the data packet received is correct/until ARQ time limit is reached

Echo check:

- copy of data sent back to sender
- data is compared to see if they match
- if it does not match, error has been detected

8 a

Statement	True ✓	False ✓
Full-duplex data transmission can be either serial or parallel	✓	
Full-duplex data transmission is when data is transmitted both ways, but only one way at a time		✓
Full-duplex transmission is always used to connect a device to a computer		✓
Full-duplex data transmission is when data is transmitted both ways at the same time	✓	
Full-duplex data transmission automatically detects any errors in data		✓

b parallel data transmission

c Two from:

- universal standard, so it is likely to be compatible with the computer
- can only be inserted one way, so there is less chance of connecting a device incorrectly
- high-speed data connection, so data transmission is fast
- uses serial transmission, so less chance of skewing errors
- does not require (wireless) network, therefore can be used even if network is down
- backwards compatible, so no additional technology required
- no need for a separate power source, because USB supplies (5V) power
- drivers are automatically identified/loaded, so no need to find them online/installs drivers automatically