

QUESTION 3.



7 (a) (i)

Accumulator:

0	1	1	1	0	1	0	1
---	---	---	---	---	---	---	---

 [1]

(ii)

Accumulator:

0	1	1	0	1	0	0	1
---	---	---	---	---	---	---	---

 [1]

explanation

- content of 124 is 0 1 1 1 1 1 1 1
- this is equivalent to 127
- contents of 127 are 0 1 1 0 1 0 0 1

[2]

(iii)

Accumulator:

0	1	0	0	0	0	0	1
---	---	---	---	---	---	---	---

 [1]

explanation

- index register value = 6
- $120 + 6 = 126$
- contents of 126 placed in the accumulator

[2]



(b) 1 mark for each correct value in the table.

Accumulator	Memory address			
	320	321	322	323
	49	36	0	0
36				
37				
				37
49				
50				
			50	

[6]

QUESTION 4.



9 (a) (i) **One mark** for the contents of the accumulator and **one mark** for the reason. [2]

Accumulator contents: 0100 0101

Reason:

Address is 60

Contents of the index register is 8

And $60 + 8 = 68$ in denary gives the address

The contents of which is 0100 0101 in binary.

(ii) **0000 0111** [1]



(b)

Instruction address	Working space	ACC	Memory address				IX	C
			100	101	102	103		
			20	100	1	0	1	
50		20						
51		21						
52						21		
53		100						
54		120						
55								
56								
57								
59							2	
60		20						
61		120						
62								'x'
63								

One mark for each shaded block.

[7]

- Contents of the Accumulator in first 2 lines (instruction addresses 50 and 51)
- Updating address 103 (instruction 52)
- Loading the Accumulator and addition (instructions 53 and 54)
- Not executing instruction 58
- Incrementing the index register (instruction 59)
- Loading the Accumulator and addition (instructions 60 and 61)
- Correct output of 'x' (instruction 62)

QUESTION 5.

4 (a) 11001110

(b)

Instruction	Working space	ACC	Memory address				IX	OUTPUT
			90	91	92	93		
			2	90	55	34	2	
20		55						
21		54						
22		54						
23							3	
24		34						
25		33						
26								
27								
28								
31		67						
32						67		
33								'C'
34								

One mark each for:

- Instruction 20
- Instructions 21 and 22
- Instruction 23
- Instructions 24 and 25
- Not executing instructions 29 and 30
- Instructions 31 and 32
- Correct output



QUESTION 6.



8 (a) (i)

Accumulator:	1	0	0	1	0	1	1	1
--------------	---	---	---	---	---	---	---	---

[1]

(ii) **One mark** for answer and **two marks** for explanation

Accumulator:	1	1	0	0	0	0	1	0
--------------	---	---	---	---	---	---	---	---

- Index Register contains $1001 = 9$
- $800 + 9 = 809$

[3]

(b) (i) **ONE** mark for each correct row.

ACC	Memory address				OUTPUT
	800	801	802	803	
	40	50	0	90	
40					
90			90		
90			90		
					Z

[4]

(ii) **107**

[1]

Page 10	Mark Scheme
	Cambridge International AS/A Level – October/November 2016



(c) (i) Any **two** from:

- Only 128 / 256 characters can be represented
- Uses values 0 to 127 (or 255 if extended form) / one byte
- Many characters used in other languages cannot be represented
- In extended ASCII the characters from 128 to 255 may be coded differently in different systems

[2]

(ii) Any **two** from:

- Uses 16, 24 or 32 bits / two, three or four bytes
- Unicode is designed to be a superset of ASCII
- Designed so that most characters (in other languages) can be represented

[2]

QUESTION 7.

Cambridge International AS/A Level – October/November 2016



7 **ONE** mark per bullet point, **MAX TWO** marks per task.

- Process /resource management
- Scheduling of processes / multi-tasking / multi-programming etc.
- Resolution of conflicts when two or more processes require the same resource
- Main memory management
- Memory protection to ensure that two programs do not try to use the same space
- Use of virtual memory
- Deciding which processes need to be in main memory at any one time
- Location of processes within the memory
- By example, e.g. when process terminates, memory is made available
- Peripheral / hardware / device management
- Installation of appropriate driver software
- Controls access to data being sent to / from hardware / peripherals
- Controls access to hardware / peripherals
- Manages communication between devices / hardware and software
- File / secondary storage management
- Maintains directory structures
- Provides file naming conventions
- Controls access
- Security management
- Makes provision for recovery when data is lost
- Provides usernames and passwords / encryption / user accounts
- Prevents unauthorised access
- Ensures privacy of data
- Provision of a software platform / environment
- On which other programs can be run
- Interrupt handling
- Identifies priorities of interrupts
- Save current memory / process values / saves data on power outage
- Loads appropriate Interrupt Service Routine (ISR)
- Any relevant example

QUESTION 8.

Cambridge
International
AS & A Level

Cambridge International Examinations
Cambridge International Advanced Subsidiary and Advanced Level



COMPUTER SCIENCE

9608/11

Paper 1 Written Paper

May/June 2017

MARK SCHEME

Maximum Mark: 75

Published

This mark scheme is published as an aid to teachers and candidates, to indicate the requirements of the examination. It shows the basis on which Examiners were instructed to award marks. It does not indicate the details of the discussions that took place at an Examiners' meeting before marking began, which would have considered the acceptability of alternative answers.

Mark schemes should be read in conjunction with the question paper and the Principal Examiner Report for Teachers.

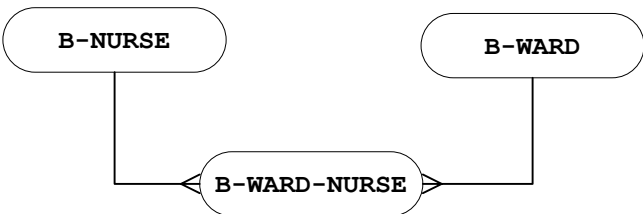
Cambridge will not enter into discussions about these mark schemes.

Cambridge is publishing the mark schemes for the May/June 2017 series for most Cambridge IGCSE[®], Cambridge International A and AS Level and Cambridge Pre-U components, and some Cambridge O Level components.

© IGCSE is a registered trademark.

This document consists of **8** printed pages.



Question	Answer		
1(a)	Many-to-one		
1(b)(i)	A-NURSE (<u>NurseID</u> , FirstName, FamilyName, WardName)		
1(b)(ii)	- The primary key <u>WardName</u> in the A-WARD table links to the foreign key <u>WardName</u> in the A-NURSE table.	1 1	
1(c)(i)	Many-to-many relationship		1
1(c)(ii)	B-WARD-NURSE (<u>WardName</u> , <u>NurseID</u>) Both attributes (with no additions) Joint primary key correctly underlined	1 1	2
1(c)(iii)	 Correct relationship between B-NURSE and B-WARD-NURSE Correct relationship between B-WARD and B-WARD-NURSE	1 1	2
1(d)(i)	SELECT NurseID, FamilyName FROM B-NURSE WHERE Specialism = 'THEATRE';	1 1 1	3
1(d)(ii)	UPDATE B-NURSE SET FamilyName = 'Chi' WHERE NurseID = '076';	1 1 1	3



Question	Answer															
2(a)(i)	<table><tr><td>1</td><td>A laser beam and a rotating mirror are used to draw an image of the page on the photosensitive drum.</td></tr><tr><td>2</td><td>C // The image is converted on the drum into an electrostatic charge.</td></tr><tr><td>3</td><td>Electrostatic charge attracts toner.</td></tr><tr><td>4</td><td>The charged paper is rolled against the drum.</td></tr><tr><td>5</td><td>D // The oppositely-charged paper picks up the toner particles from the drum. After picking up the toner, the paper is discharged to stop it clinging to the drum.</td></tr><tr><td>6</td><td>A // The paper passes through a fuser, which heats up the paper. The toner melts and forms a permanent image on the paper.</td></tr><tr><td>7</td><td>B // The electrical charge is removed from the drum and the excess toner is collected.</td></tr></table> C in the correct place DA, AB	1	A laser beam and a rotating mirror are used to draw an image of the page on the photosensitive drum.	2	C // The image is converted on the drum into an electrostatic charge.	3	Electrostatic charge attracts toner.	4	The charged paper is rolled against the drum.	5	D // The oppositely-charged paper picks up the toner particles from the drum. After picking up the toner, the paper is discharged to stop it clinging to the drum.	6	A // The paper passes through a fuser, which heats up the paper. The toner melts and forms a permanent image on the paper.	7	B // The electrical charge is removed from the drum and the excess toner is collected.	1 1 1
1	A laser beam and a rotating mirror are used to draw an image of the page on the photosensitive drum.															
2	C // The image is converted on the drum into an electrostatic charge.															
3	Electrostatic charge attracts toner.															
4	The charged paper is rolled against the drum.															
5	D // The oppositely-charged paper picks up the toner particles from the drum. After picking up the toner, the paper is discharged to stop it clinging to the drum.															
6	A // The paper passes through a fuser, which heats up the paper. The toner melts and forms a permanent image on the paper.															
7	B // The electrical charge is removed from the drum and the excess toner is collected.															
2(a)(ii)	Inkjet printer	1														
2(b)	Hard disk drive // HDD Solid state drive //SSD // flash memory One from: <i>Hard disk</i> Inexpensive per unit of storage <u>Larger</u> storage capacity than flash drive <i>Solid state storage</i> No moving parts / noise Robust Low latency // Fast read/write time	1 1 1 1 1 1 1 1														



Question	Answer	
3(a)	Sampling rate The <u>number of samples</u> taken <u>per unit time</u> // the number of times the amplitude is measured <u>per unit time</u> Increasing the sampling rate will increase the accuracy / precision of the digitised sound // Increasing the sampling rate will result in smaller quantisation errors.	1 1
3(b)(i)	Pixel Smallest picture element which can be drawn Screen resolution The number of pixels which can be viewed horizontally and vertically on the screen // or by example - A typical screen resolution is 1680 pixels × 1080 pixels.	1 1
3(b)(ii)	8	1
3(b)(iii)	Working: Max <u>two</u> from: - Number of pixels is 2048×512 - One pixel will be stored as one byte - Number of kilobytes = $(2048 \times 512) / 1024$ Answer: <u>One</u> mark: Number of kilobytes = 1024 KB	1 1 1 1
3(b)(iv)	<u>One</u> from: - Confirmation that the file is a BMP - File size - Location/offset of image data within the file - Dimensions of the image in pixels // image resolution - Colour depth (bits per pixel) - Type of compression used, if any	1 1 1 1 1 1



Question	Answer	
4(a)(i)	500	
4(a)(ii)	496	
4(a)(iii)	502	
4(a)(iv)	86	1
4(b)	00000001 00010001 00000110 01100001 Both correct op codes Operand 0001 0001 Operand 0110 0001	3
4(c)	256	1
4(d)(i)	07 C2 07 C2	2
4(d)(ii)	LDI 63 LDI 63	2



Question	Answer																																																																																	
5(a)(i)	- Count the number of one bits in the <u>first seven</u> bit positions - Add a 0 or 1 to bit position 0, to make the count of one bits an <u>odd</u> number																																																																																	
5(a)(ii)	A = 1 B = 1																																																																																	
5(a)(iii)	Two from: - A parity bit is worked out for each <u>column</u> 1 - The computer checks the parity of each bit position in parity byte // the computer generates copy of the parity byte and <u>compares</u> 1 - If incorrect parity then there is an error in the data received // No parity error means no error in the data received 1 - The position of the incorrect bit can be determined 1	2																																																																																
5(b)(i)	<table><tr><th colspan="8">Bit position</th></tr><tr><th>7</th><th>6</th><th>5</th><th>4</th><th>3</th><th>2</th><th>1</th><th>0</th></tr><tr><td>1</td><td>0</td><td>0</td><td>0</td><td>1</td><td>1</td><td>0</td><td>0</td></tr><tr><td>0</td><td>0</td><td>1</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr><tr><td>0</td><td>0</td><td>1</td><td>1</td><td>0</td><td>1</td><td>0</td><td>1</td></tr><tr><td>1</td><td>1</td><td>1</td><td>1</td><td>0</td><td>0</td><td>0</td><td>1</td></tr><tr><td>1</td><td>1</td><td>0</td><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td></tr><tr><td>0</td><td>0</td><td>1</td><td>0</td><td>0</td><td>1</td><td>0</td><td>0</td></tr><tr><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>1</td></tr></table> <table><tr><td>0</td><td>1</td><td>0</td><td>1</td><td>1</td><td>0</td><td>0</td><td>0</td></tr></table>	Bit position								7	6	5	4	3	2	1	0	1	0	0	0	1	1	0	0	0	0	1	0	0	0	0	0	0	0	1	1	0	1	0	1	1	1	1	1	0	0	0	1	1	1	0	0	0	0	1	0	0	0	1	0	0	1	0	0	0	0	0	0	0	0	0	1	0	1	0	1	1	0	0	0	2
Bit position																																																																																		
7	6	5	4	3	2	1	0																																																																											
1	0	0	0	1	1	0	0																																																																											
0	0	1	0	0	0	0	0																																																																											
0	0	1	1	0	1	0	1																																																																											
1	1	1	1	0	0	0	1																																																																											
1	1	0	0	0	0	1	0																																																																											
0	0	1	0	0	1	0	0																																																																											
0	0	0	0	0	0	0	1																																																																											
0	1	0	1	1	0	0	0																																																																											
5(b)(ii)	Three from: - Consider each row in sequence 1 - Identify any row with incorrect parity 1 - Repeat the process for each column in sequence 1 - Identify where a row and column with incorrect parity intersect 1	3																																																																																



Question	Answer	
6(a)	Main memory management The user moves the mouse on the desktop Input/Output Management The user closes the Spreadsheet program Secondary storage Management The user selects the Save command to save their spreadsheet file Human computer interface The user selects the Print command to output their spreadsheet file One mark for each correct line from each left hand box to max <u>three</u> marks.	
6(b)(i)	File compression software	1
6(b)(ii)	Backup software	1
6(b)(iii)	Disk repair software	1
6(b)(iv)	Anti-virus software	1



Question	Answer	
7(a)	Two from: - The user's web browser is the client software 1 - The requested web page has program code / script embedded <u>within it</u> 1 - This code is interpreted by the web browser 1	
7(b)	Four from: - The browser parses the URL to obtain the Domain Name 1 - The browser software passes the Domain Name to the nearest Domain Name Server (DNS) 1 - The DNS stores a list of Domain Names and matching IP addresses 1 - The DNS Name Resolver looks for the Domain Name in its database 1 - If found the corresponding IP address is returned to the originator 1 - If not found the request is forwarded to another higher level DNS 1 - The original DNS adds the returned IP address to its cache 1 - The original DNS returns the IP address to the originator 1 - The browser uses the IP address to request the required web page from the <u>web server</u> 1 - The web server retrieves the page and delivers it to the originator 1 - The browser software interprets <u>the script</u> and displays the web page 1	Max 4
7(c)(i)	Message1, Message2 1 x 1	2
7(c)(ii)	6 – 19	1
7(c)(iii)	11	1
7(c)(iv)	Checks that the product code has not be left blank // presence check on product code	1
7(c)(v)	Two checks from: One mark for check and one mark for description - Range check 1 - Check the number entered is (say) between 1 and 100 1 - Format check 1 - Checks the product code is a particular format // Checks the number has digit characters only // by example 1 - Length check 1 - The number of items has exactly five characters 1 - Existence check 1 - To ensure the product code has been assigned 1	Max 4

QUESTION 9.



Question	Answer	
4(a)(i)	Two from: - The hardware is unusable without an OS // hides complexity of hardware from user 1 - Acts as an interface/ controls communications between user and hardware / hardware and software // or by example 1 - Provides software <u>platform / environment</u> on which other programs can be run 1	
4(a)(ii)	One mark for the name and one mark for description. Max two management tasks. - Provides the Human Computer Interface (HCI) 1 Controls communications between user and hardware// or by example 1 - Main memory management 1 Memory protection to ensure that two programs do not try to use the same space // Use of virtual memory // Location of processes within the memory // By example 1 - File / Secondary storage management 1 Maintains directory structures // Provides file naming conventions // Controls access 1 - Peripheral / hardware / device / Input-Output management 1 Installation of appropriate driver software // Controls access to data being sent to/from hardware/peripherals // Controls access to hardware/peripherals // manages communication between devices. 1 - Interrupt handling 1 Identifies priorities of interrupts // Saves data on power outage // Loads appropriate Interrupt Service Routine (ISR) // By example 1 - Security management 1 Makes provision for recovery when data is lost // Provides usernames and passwords // Prevents unauthorised access // Ensures privacy of data 1	Max 4
4(b)(i)	File compression software	1
4(b)(ii)	Backup software	1
4(b)(iii)	Disk defragmenting software	1
4(b)(iv)	Anti-virus software	1

Question	Answer	Marks

QUESTION 10.



Question	Answer
1	1 Mark for stating the management task 1 Mark for a corresponding description Maximum 2 marks for each task Maximum 3 tasks Process / Task Management • Allocation of processor time • Scheduling of processes or tasks / multi-tasking / multi-programming etc. • By example – e.g. round-robin, shortest remaining time first etc. • Resolution of conflict when two or more processes require the same resource Secondary Storage management • Storage space divided into file allocation units • Space allocated to particular files • OS maintains a file directory and FAT • Provides file naming conventions • Controls access. Peripheral / Hardware / Device / Input/output Management • Installation of appropriate driver software • Controls access to data being sent to/from hardware/peripherals • Controls access to hardware/peripherals • Manages communication between devices / hardware and software Provision of a User interface • Allows user interaction with the computer system// Facilitates human computer communication • Hides the complexity of the hardware from the user • Or by example – e.g. GUI, command line etc. Interrupt Handling • Halts the execution of the current process • Stores the values of the current process on the stack • Loads and executes the appropriate ISR code • Use of priorities for handling simultaneous interrupts • Saves data on power outage Security Management • Makes provision for recovery when data is lost • Provides usernames and passwords / encryption / user accounts • Prevents unauthorised access • Ensures privacy of data Provision of a software platform / environment • On which other programs / applications can be run

QUESTION 11.



Question	Answer	Marks
4(a)(i)	1 mark for each type of addressing Direct addressing • The operand is the address where the data is stored Indirect addressing • The operand is an address, that address holds another address where the data is stored	2



Question	Answer													
4(a)(ii)	1 mark per bullet point Direct addressing: 20 is the address of the data Indirect addressing: 20 is an address which holds the address where the data is stored													
4(b)	1 mark for 1 correct tick 2 marks for 3 correct ticks <table border="1" data-bbox="392 618 1241 880"> <thead> <tr> <th>Instruction</th><th>Symbolic</th><th>Absolute</th></tr> </thead> <tbody> <tr> <td>ADD 90</td><td></td><td>✓</td></tr> <tr> <td>CMP found</td><td>✓</td><td></td></tr> <tr> <td>STO 20</td><td></td><td>✓</td></tr> </tbody> </table>	Instruction	Symbolic	Absolute	ADD 90		✓	CMP found	✓		STO 20		✓	2
Instruction	Symbolic	Absolute												
ADD 90		✓												
CMP found	✓													
STO 20		✓												
4(c)(i)	186	1												
4(c)(ii)	BA	1												
4(c)(iii)	−70	1												
4(d)	1 mark per bullet point - Outputting * (instruction 71) - Storing 130 in 203 (instruction 72) - Loading, incrementing accumulator and storing in 204 (instructions 73, 74 and 75) - Incrementing Index Register (instruction 76) - Loading 133, comparing and jumping to 81 (instructions 77, 78 and 79) - Loading, comparing and jumping to 74 (instructions 81, 82 and 83) - Incrementing accumulator, storing in 204 and incrementing index register, loading 130 (instructions 74–79) - Outputting * to end (instructions 80–84)	8												

[illegible]

QUESTION 12.



Question	Answer	Marks
3(a)(i)	1 mark per bullet point Absolute addressing: - The operand is a numeric address // The numeric address is given // referring directly to a memory location Symbolic addressing: - The operand is a word/symbol // A word/symbol represents the memory location/address	2
3(a)(ii)	1 mark per example Absolute addressing: For example, <code>ADD 230</code> Symbolic addressing: For example, <code>ADD num1</code>	2
3(b)(i)	1 mark per bullet point Indexed addressing: - The address to be used is formed by: operand + the contents of the Index Register (IX) Immediate addressing: - The operand is not an address // the operand is the actual value to be loaded	2



Question	Answer	
3(b)(ii)	1 mark per example Indexed: For example, LDX 20 Immediate: For example, ADD #20	
3(c)(i)	193	1
3(c)(ii)	C1	1
3(c)(iii)	–63	1
3(d)	1 mark per bullet point • Loading 2, comparing with 104 (instructions 40 and 41) • Loading 302 (instruction 43) • Comparing, and branching to 47 (instructions 44, 45) • Loading, decrementing accumulator and storing (instructions 47, 48 and 49) • Incrementing Index Register (instruction 50) • Loading 303, comparing and outputting + (instructions 43–46) • Loading, decrementing accumulator and storing, incrementing Index Register and end (instructions 47–51, 41, 42 and 54)	7

[illegible]

QUESTION 13.



Question	Answer	Marks
2(a)	1 mark per bullet point Relative addressing: • A number (the offset) is added to the base address to give the actual address Indexed addressing: • The contents of the index register are added to operand to give the actual address	2
2(b)(i)	242	1



Question	Answer																
2(b)(ii)	F2																
2(b)(iii)	−14																
2(b)(iv)	<table border="1"><tr><td>1</td><td>1</td><td>1</td><td>1</td><td>0</td><td>0</td><td>1</td><td>1</td></tr></table>								1	1	1	1	0	0	1	1	1
1	1	1	1	0	0	1	1										
2(c)	1 mark per bullet point • Loading 8 (instruction 20) • Comparison and loading 453 (instructions 21–23) • Outputting & (instruction 26) • Loading, incrementing and storing in 96 (instructions 27–29) • Incrementing Index Register (instruction 30) • Jumping and loading 452 (instruction 23) • Jumping, loading, incrementing, storing in 96, incrementing IX and end (instructions 24–32)								7								



Instruction address	ACC	Memory address						IX	
		93	94	95	96	97	98		
		453	453	452	8	10	453	8	
20	8								
21									
22									
23	453								
24									
25									
26									&
27	8								

QUESTION 14.



Question	Answer		
5(a)	1 mark for two correct ticks, 2 marks for three correct ticks		
	Task	First pass	Second pass
	Creation of symbol table	✓	
	Expansion of macros	✓	
	Generation of object code		✓
	Removal of comments	✓	
5(b)	1 mark per bullet point to max 2 • Data movement • Input and output of data • Arithmetic operations • Jump instructions • Compare instructions		2
5(c)	1 mark per bullet point • Storing 0 in 300 (line 21) • Loading 65 (line 28) • Outputting A (line 29) • Loading 0 (line 30), incrementing ACC (line 31) and storing in 300 (line 32) • Incrementing IX (line 33) • Loading 67 (line 24) and adding 33 (line (25) • Outputting d (line 26) • Loading 1 (line 30), incrementing ACC (line 31), storing in 300 (line 32) and incrementing IX (line 33)		8

[illegible]

QUESTION 15.



Question	Answer	Marks
3(a)	1 mark for each error and correction • Line 02 should be +1 not -1 // $PC \leftarrow [PC] + 1$ • Line 03 should be double brackets around MAR // $MDR \leftarrow [[MAR]]$ • Line 04 should be MDR not MAR // $CIR \leftarrow [MDR]$	3
3(b)	1 mark for each group to max. 2 • Data movement • Arithmetic operations • (Unconditional and conditional) jump instructions • Compare instructions • Modes of addressing	2
3(c)	1 mark per bullet • Storing 0 in 401 (line 51) • Loading memory location 300, value 2 to ACC (line 52) • Adding 64 to ACC to give 66 (line 55) • Outputting B (line 56) • Load 0 (line 57), increment ACC (line 58) and store 1 in 401 (line 59) • Incrementing IX (line 60) • Loading 5 (line 52), adding 64 (line 55), outputting E (line 56) loading 1 (line 57), incrementing ACC (line 58), storing 2 in 401 (line 59) and incrementing IX (line 60) • Load 0 (line 52) <u>and end</u>	8



Question	Answer	
3(d)(i)	1 mark for correct answer 0100 0001	
3(d)(ii)	1 mark for correct answer 41	1
3(d)(iii)	1 mark for correct answer 0044	1

QUESTION 16.



Question	Answer												
1(a)	1 mark for each correct entry <table><tr><th>Management task</th><th>Description</th></tr><tr><td>Memory management</td><td>Handles the allocation of memory to processes // Ensures two programs do not attempt to use the same memory locations // Keeps track of allocated and free memory locations</td></tr><tr><td>Security management</td><td>Provides user accounts and passwords</td></tr><tr><td>Interrupt processing</td><td>Handles the signals sent when the attention of the processor is required elsewhere</td></tr><tr><td>Provision of a software platform</td><td>Provides an environment within which programs can be run</td></tr></table>		Management task	Description	Memory management	Handles the allocation of memory to processes // Ensures two programs do not attempt to use the same memory locations // Keeps track of allocated and free memory locations	Security management	Provides user accounts and passwords	Interrupt processing	Handles the signals sent when the attention of the processor is required elsewhere	Provision of a software platform	Provides an environment within which programs can be run	
Management task	Description												
Memory management	Handles the allocation of memory to processes // Ensures two programs do not attempt to use the same memory locations // Keeps track of allocated and free memory locations												
Security management	Provides user accounts and passwords												
Interrupt processing	Handles the signals sent when the attention of the processor is required elsewhere												
Provision of a software platform	Provides an environment within which programs can be run												
1(b)(i)	1 mark per bullet point to max 2 for formatter, max 2 for defragmenter hard disk formatter - Makes existing data inaccessible - Partitions the disk into logical drives - Sets up the (specified) file system - Prepares the disk for initial use - May check for errors on the disk hard disk defragmenter - Re-organises the disk contents - Moves split files so they are contiguous - Creates a larger area of (contiguous) free space		4										
1(b)(ii)	1 mark per bullet point For example: - Backup software - File compression - Virus checker - Disk contents analysis / repair		3										

QUESTION 17.



Question	Answer	
2(a)(i)	1 mark per device to max 2 e.g. • Trackpad/touchpad • microphone • touchscreen • scanner	
2(a)(ii)	1 mark per device to max 2 e.g. • printer • speakers • touchscreen	2
2(a)(iii)	Magnetic hard disk drive // solid state drive	1
2(a)(iv)	1 mark per bullet point to max 3 • The ball touches horizontal and vertical rollers • When the ball rotates / moves • ... one or both of the rollers rotate as well • Each roller connects to a shaft which spins a disk with holes • Infrared beams shine through the holes in the disks • As the ball moves the roller the beam is broken by the space between the holes • ... creating pulses of light • The distance and/or speed of the mouse is determined from the rate of the pulses • ... by an on-board processor chip // by driver software in the computer	3
2(b)	1 mark per bullet point to max 4 for each management task, max 6 in total Process management: • Manages the scheduling of processes • ... allows multi-tasking / multi-processing • ... ensures fair access • ... handles priorities • Manages the resources the processes need • Enables processes to share information • Prevents interference between processes// resolution of conflicts Provision of a user interface: • Allows a user to communicate with the hardware // vice-versa • ... by making navigation around the system easier • Provides facility for user inputting data • Provides facility for outputting to the user • By example e.g. command line / GUI / menu-driven	6



Question	Answer	
2(c)	1 mark per bullet point to max 3 for each utility program, max 4 in total Virus checker: • Scans files stored on a computer system for malicious code • Scans files when they enter the system / memory stick inserted / download etc. • Sets up a schedule for virus-checking • Isolates / quarantines / deletes viruses • Regularly updates the virus definitions Backup software: • Creates a copy of the contents of a disk / partition. Can be set up to automatically backup // schedules backups • Allows the user to decide what is backed up, e.g. all data // all files that have changed since the last backup • Allows the user to set up an off-site backup • May encrypt the backup files • Restores the data if necessary	
2(d)	1 mark per bullet point to max 2 • The code is translated one line at a time • ... and executed immediately • The interpreter stops as soon as it finds an error	2

QUESTION 18.



Question	Answer													
1(a)	1 mark per correct line <table><thead><tr><th>Type of software</th><th>Description</th></tr></thead><tbody><tr><td>Operating system</td><td>Provides a ready-built routine that can be imported into a program</td></tr><tr><td>Utility program</td><td>Provides an interface between the user and the hardware</td></tr><tr><td>Library program</td><td>Converts source code into a low-level language</td></tr><tr><td>Compiler</td><td>Creates a new document for the user to edit</td></tr><tr><td></td><td>An additional program that helps to maintain or configure the system</td></tr></tbody></table>	Type of software	Description	Operating system	Provides a ready-built routine that can be imported into a program	Utility program	Provides an interface between the user and the hardware	Library program	Converts source code into a low-level language	Compiler	Creates a new document for the user to edit		An additional program that helps to maintain or configure the system	
Type of software	Description													
Operating system	Provides a ready-built routine that can be imported into a program													
Utility program	Provides an interface between the user and the hardware													
Library program	Converts source code into a low-level language													
Compiler	Creates a new document for the user to edit													
	An additional program that helps to maintain or configure the system													
1(b)	1 mark per bullet point to max 3 Disk repair: • Checks for any errors / inconsistencies / bad sectors on the disk • Resolves any errors on the disk • Retrieves files / data from a damaged disk // re-constructs directory // recovers disc when data corrupt • Marks bad sectors on the disk // marks bad sectors as unusable	3												

QUESTION 19.



Question	Answer	Marks
2(a)	1 mark per bullet point to max 4 for each management task, max 6 in total Process Management • Manages the scheduling of processes • ... allows multi-tasking / multi-processing • ... ensures fair access • ... handles priorities • Manages which resources the processes require • Enables processes to share information • Prevents interference between processes // resolution of conflicts Memory Management • Allocates memory to processes • Ensures fair usage of memory • Organises memory / by example • Makes use of virtual memory • Keep processes separate • To release memory when a process stops	6



Question	Answer	
2(b)(i)	1 mark per bullet point to max 2 - The amplitude of the wave is measured ... at set, regular time intervals - The value is stored as a binary number	
2(b)(ii)	1 mark per bullet point - Sampling resolution of 44100 Hz takes more samples per second, so the file size will be larger // Sampling resolution of 21000 Hz takes fewer samples per second, so the file size will be smaller - At a resolution of 44100 Hz, the sound recording is a closer / more accurate representation of Leonardo's voice // At a resolution of 21000 Hz, the sound recording is a less accurate representation of Leonardo's voice	2
2(b)(iii)	1 mark for naming a feature, 1 mark for description, max 2 marks for each feature e.g. - Amplify ... Increase the volume of a section of sound - Change pitch ... Increase/decrease frequency of section(s) - Change sampling resolution ... to change the accuracy of the sound / file size	4