

ICT0417 - Grade 10

Crash Course Contents

Content List:

- Day1: Networks and the effects of using them
- Day 2: ICT applications - Booking System
- Day 3: ICT application - Sensors and Microprocessors
- Day 4: ICT application - Expert system
- Day 5: The systems life cycle
- Day 6: Database (SQL)
- Day 7: Website authoring(HTML)
- Day 8: CSS
- Day 9: Spreadsheet (Multiple Functions)
- Day 10: Presentation(Master Slide in depth)

Content Detail:

Day1: Networks and the effects of using them

4.1 networks:

- understand how a router works and its purpose
 - describe how networks and individual computers connect to the internet
 - describe how a router stores computer addresses
 - describe how it routes data packets
- understand the use of other common network devices, including: network interface cards, hubs, bridges, switches, modems
- understand the use of WiFi and Bluetooth in networks
 - describe how computers can use WiFi to connect to a network
 - describe how computers can use Bluetooth to connect to a network
 - compare and contrast Bluetooth and WiFi
- understand how to set up and configure a small network, including: access to the internet, the use of a browser, the use of email, access to an Internet Service Provider (ISP)
- understand the characteristics and purpose of common network environments, such as intranets and the internet
 - define what the internet is
 - define what an intranet is
 - describe the differences between an intranet and the internet
 - explain the purpose of an intranet and how that differs from the purpose of the internet
 - describe the uses of an intranet
 - describe the uses of the internet

- define the terms Local Area Network (LAN), Wireless Local Area Network (WLAN) and Wide Area Network (WAN)
- describe the differences between a LAN, a WLAN and a WAN
- understand the advantages and disadvantages of using different types of computer to access the internet
- compare the advantages and disadvantages of using laptop computers, desktop computers, tablet computers and smartphones to access the internet

Day 2: ICT Application : booking systems

- identify areas where booking systems are used (e.g. travel industry, theatres and cinemas)
- describe the online processing involved in booking tickets
- discuss the advantages and disadvantages of online booking systems

6.9 banking applications

Candidates should be able to:

- describe the computer processing involved in Electronic Funds Transfer (EFT)
- describe the computer processing involved in using automatic teller machines (ATM) (e.g. withdrawing cash, depositing cash or cheques, checking account balance, mini statements, smartphone/cellphone recharge/top up, bill paying, money transfers, ordering paper-based goods)
- describe the use of processing involved in credit/debit card transactions
- describe the clearing of cheques
- describe phone banking
- describe internet banking, and discuss the advantages and disadvantages of it

Day 3: 6.3 measurement applications

- describe a range of measurement applications (e.g. scientific experiments, weather stations)
- explain the difference between analogue data and digital data
- explain the need for conversion between analogue and digital data
- describe the use of microprocessors and computers in a number of applications (e.g. pollution monitoring, intensive care units in hospitals)
- discuss the advantages and disadvantages of using computers in measurement rather than humans

6.4 microprocessors in control applications

- describe the role of a microprocessor or computer in control applications, including the role of the pre-set value
- describe the use of computer control in applications (e.g. turtle graphics, automatic washing machines, automatic cookers, computer controlled central heating systems, burglar alarms, computer controlled glasshouses)

Day 4: expert systems

- identify a range of applications which use expert systems (e.g. mineral prospecting, car engine fault diagnosis, medical diagnosis, chess games)
- identify the components of an expert system (e.g. interactive user interface, inference engine, rules base, knowledge base)
- describe how an expert system is used to suggest diagnoses

Day 5: The systems life cycle

7.1 analysis

- methods of researching an existing system
 - identify and describe methods of researching an existing system (e.g. observation, interviews, questionnaires and examination of existing documents)
 - discuss the disadvantages and advantages of the different methods
- recording and analysing information about the current system
 - describe the need to identify inputs, outputs and processing of the current system
 - describe the need to identify problems with the current system
 - describe the need to identify the user and information requirements for the new system
- system specification
 - identify and justify suitable hardware for the new system
 - identify and justify suitable software for the new system

7.2 design

- describe how it is necessary to design documents, files, forms/inputs, reports/outputs and validation
- produce designs to solve a given problem
- design data capture forms and screen layouts
- design report layouts and screen displays
- design validation routines (including length check, type check, format check, presence check, check digit)
- design the required data/file structures (e.g. field length, field name, data type)

Day 6: 18.2 manipulate data

- use arithmetic operations or numeric functions to perform calculations within a database
 - create a calculated field
 - perform calculations at run time using formulae and functions, including: addition, subtraction, multiplication, division, sum, average, maximum, minimum, count
- use suitable software tools to sort data appropriately in a database
 - sort data using a single criterion and using multiple criteria where necessary, into ascending or descending order
- use suitable software tools to search a database to select subsets of data
 - perform suitable searches using a single criterion and using multiple criteria, on different field types such as alphanumeric, numeric, Boolean

- perform searches using a variety of operators including: AND, OR, NOT, LIKE, >, <, =, >=, <=, <> – perform searches using wildcards, as appropriate

Day 7: 21.2 create a web page:

- use software tools to create the content layer of a web page to meet the needs of the audience
 - explain the purpose of the head and body sections of a web page
 - place appropriate elements in the head section of a web page, including: page title, attached stylesheets
 - place appropriate elements in the body section of a web page
 - explain why tables are used to structure elements within a web page – insert a table, including: table header, table rows, table data
 - use appropriate table attributes, including: to adjust cells to span more than one row/column, to set table and cell sizes in terms of pixels and/or % values, to apply styles to tables, to meet the needs of the audience
- use software tools to appropriately place the content in a web page

Day 8: use stylesheets

- use software tools to create the presentation layer of a web page
 - explain what is meant by the term cascading stylesheets
 - explain the difference between attached stylesheets and inline style attributes
 - explain the hierarchy of multiple attached stylesheets and inline styles within a web page
 - explain the difference between a style and a class
 - create generic external styles and inline style attributes, including: background properties (e.g. colour, images), font properties (e.g. style, typeface), table properties (e.g. background colour, horizontal and vertical alignment, spacing, padding, borders: including collapsed, border thickness, visible/invisible)
 - create external styles to be tagged in a web page including: h1, h2, h3, p, li, as required
 - specify the font appearance for each style, including features such as: font family, size, colour, alignment, bold and italic
 - save styles in cascading stylesheet format
 - explain why relative file paths must be used for attached stylesheets
 - attach an external stylesheet to a web page using a relative file path
 - attach comments to an external stylesheet

Day 9: create a data model

- create and edit a data model
 - define the terms: cells, rows, columns, sheets, tabs, pages, charts
 - explain the importance of accurate data entry in spreadsheets
 - enter data with 100 per cent accuracy
 - edit the structure of an existing model, including: inserting cells, deleting cells, inserting rows, deleting rows, inserting columns, deleting columns

- define the terms: formula, function, absolute reference, relative reference, ranges, named cell, named range, nested formulae/functions
- explain the difference between a formula and a function
- explain the order in which mathematical operations are performed and use brackets to make sure that formulae work
- use mathematical operators, including: add, subtract, multiply, divide, indices, where necessary
- explain the function of, and use, absolute and relative referencing, as appropriate, when formulae are to be replicated
- use absolute and relative references, named cells, named ranges and nested formulae, as appropriate
- use functions, including: sum, average, maximum, minimum, integer, rounding, counting, LOOKUP, VLOOKUP, HLOOKUP, IF and nested functions, when necessary

Day 10: Presentations

- use a master slide to appropriately place objects and set suitable styles to meet the needs of the audience – identify the need for consistency of presentation, in terms of styles, point sizes, colour schemes, transitions and animations
 - use the master slide to place objects appropriately, including: images, text, logos, slide footers, automated slide numbering
 - use the master slide to set font styles, heading styles and colour schemes, as required by the audience
 - manipulate and use specified areas for headings, subheadings, bullets, images, charts, colours, text boxes, presenter notes, audience notes, as appropriate
- use suitable software tools to create presentation slides to meet the needs of the audience
 - insert a new slide, when required, selecting the appropriate slide type for the purpose
 - place text on the slides including: headings, subheadings, bulleted lists, where appropriate
 - apply consistent styles using available software tools, including: select from the presentation colour scheme, the use of text enhancement
 - place appropriate images on the slides, including: still images, video clips, animated images – place sound within a slide
 - place charts imported from a spreadsheet
 - place other objects including: symbols, lines, arrows, call out boxes
 - create consistent transitions between pages
 - create consistent animation facilities on text, images and other objects
- use suitable software tools to display the presentation in a variety of formats, including: looped on-screen carousel, controlled presentation, presenter notes, audience notes, taking into account the needs of the audience

IT9626 - Grade 11

Crash Course Contents

Content List:

Day 1: Monitoring and Control
Day 2: Hardware and Software
Day 3: Expert System
Day 4: Spreadsheet Functions
Day 5: Spreadsheet Testing
Day 6: Algorithms
Day 7: Flowcharts
Day 8: Normalization Introduction
Day 9: 1NF to 3NF conversion
Day 10: Data Dictionary

Content Detail:

Day 1: Monitoring and control

3.1 Monitoring technologies

- Sensors • Uses
- Calibration
- Advantages and disadvantages

Examples of sensors including: light/UV, temperature, pressure, humidity, sound, infrared, touch sensors, electromagnetic field sensors and proximity sensors

Including uses:

Weather stations use temperature sensors to measure ambient temperature, pressure sensors to measure atmospheric pressure, humidity sensors to measure absolute and relative humidity, light sensors to measure sunlight

Examples of other uses including: monitoring water pollution, weather stations, environmental monitoring

The importance of calibration

The various methods used to calibrate devices, including when to use one point calibration, two point calibration and multi- point calibration, knowing methods of calibrating readings from sensors such as temperature sensors

Day 2: Hardware and software

2.1 Mainframe computers and supercomputers

Characteristics

- Uses

- Advantages and disadvantages of mainframe and supercomputers

2.2 System software

- Types
- Uses
- Advantages and disadvantages

2.3 Utility software

- The need for utility software
- Types
- Uses
- Advantages and disadvantages

Characteristics including: longevity, RAS, security, performance metrics (MIPS and FLOPS), volume of input, output and throughput, fault tolerance, operating system, type of processor, heat maintenance

Mainframe computer uses including: census, transaction processing, industry statistics, consumer statistics

Supercomputer uses including: quantum mechanics, weather forecasting climate research

Types including: compilers, interpreters, linkers, device drivers, operating systems and utilities

Candidates will be expected to explain how high level language is translated to run on different computer systems

Including: anti-virus, back-up, data compression, disk defragmentation (including understanding the structure of hard disk storage), formatting, file-copying, deleting

Day 3: How expert systems are used to produce possible solutions for different scenarios

Components including: user interface, inference engine, knowledge base (as a database of facts and rules base), explanation system, knowledge base editor

Scenarios including: mineral prospecting, investment analysis, financial planning, insurance planning, car engine fault diagnosis, medical diagnosis, route scheduling for delivery vehicles, plant and animal identification

Candidates are expected to understand the concepts of backward chaining and forward chaining

Including the terms: data driven and goal driven, their use in diagnoses, gaming and artificial intelligence

Day 4: 8.1 Create a spreadsheet

Create structure

- Create page/screen structures to meet the requirements of an audience and/ or task specification/house style
- Create/edit spreadsheet structures
- Protect cells and their content
- Freeze panes and windows Create formulae and use functions
- Use formulae
- Use absolute reference, relative reference, named cells, named ranges
- Know and understand why absolute and relative referencing are used

- Use functions

Day 5: Testing

Test a spreadsheet structure

Create and apply a test plan to test functions , validation rules

Day 6: Algorithm

- Edit a given algorithm
- Write an algorithm using pseudocode to solve a given problem

Candidates should be able to write a basic algorithm that demonstrates a decision making process

Including:

Conditional branching Looping

Nested loops Procedures/subroutines

To include:

INPUT/READ

WRITE/PRINT IF...ELSE...ENDIF

WHILE... ENDWHILE REPEAT...UNTIL CASE...ENDCASE Comparison operators >, <, =

Day 7:

- Edit a given flowchart
- Draw a flowchart to solve a given

Day 8:

10.2 Normalisation to third normal form (3NF)

Know and understand the characteristics of data in unnormalised form (UNF), first normal form (1NF), second normal form (2NF) and third normal form (3NF)

Know and understand the normalisation of data (to include the advantages and disadvantages)

Day 9: 1NF to 3NF Conversion

Normalise a database to first normal form (1NF), second normal form (2NF) and third normal form (3NF)

Day 10: Data Dictionary

10.3 Data dictionary

Know and understand the components of a data dictionary

Create a data dictionary and select appropriate data types for a given set of data and a given situation

Identify different data types

Gr 11 Crash Course Content (Computer Science -9618)

1 Information representation	(Day 1)
1.1 Data Representation	
1.2 Multimedia – Graphics, Sound	
1.3 Compression	
2 Communication	(Day 2)
2.1 Networks including the internet	
3 Hardware	(Day 3)
3.1 Computers and their components	
3.2 Logic Gates and Logic Circuits	
4 Processor Fundamentals	(Day 4)
4.1 Central Processing Unit (CPU) Architecture	
4.2 Assembly Language	
4.3 Bit manipulation	
5 System Software	(Day 5)
5.1 Operating System	
5.2 Language Translators	
6 Security, privacy, and data integrity	(Day 6)
6.1 Data Security	
6.2 Data Integrity	
7 Ethics and Ownership	(Day 7)
7.1 Ethics and Ownership	
8 Databases	(Day 8)
8.1 Database Concepts	
8.2 Database Management System (DBMS)	
8.3 Data Definition Language (DDL) & Data Manipulation Language(DML)	
9 Algorithm Design and Problem-Solving	(Day 9)
9.1 Computational Thinking Skills	
9.2 Algorithms	
10 Data Types and structures	(Day 10)
10.1 Data Types and Records	
10.2 Arrays	
10.3 Files	
10.4 Introduction to Abstract Data Types (ADT)	
11 Programming	(Day 11)
11.1 Programming Basics	
11.2 Constructs	
11.3 Structured Programming	
12 Software Development	(Day 12)
12.1 Program Development Lifecycle	
12.2 Program Design	
12.3 Program Testing and maintenance	



Day 1 - Information representation

1.1 Data Representation

Show understanding of binary magnitudes and the difference between binary prefixes and decimal prefixes

Understand the difference between and use:

- kibi and kilo
- mebi and mega
- gibi and giga
- tebi and tera

Show understanding of the basis of different number systems

Use the binary, denary, hexadecimal number bases and Binary Coded Decimal (BCD) and one's and two's complement representation for binary numbers

Convert an integer value from one number base / representation to another

Perform binary addition and subtraction

Using positive and negative binary integers

Show understanding of how overflow can occur

Describe practical applications where Binary Coded Decimal (BCD) and Hexadecimal are used

Show understanding of and be able to represent character data in its internal binary form, depending on the character set used

Familiar with ASCII (American Standard Code for Information Interchange), extended ASCII and Unicode. Students will not be expected to memorise any particular character codes

1.2 Multimedia

Graphics

Show understanding of how data for a bitmapped image are encoded

Use and understand the terms: *pixel, file header, image resolution, screen resolution, colour depth/bit depth*

Perform calculations to estimate the file size for a bitmap image

Show understanding of the effects of changing elements of a bitmap image on the image quality and file size

Use the terms: *image resolution, colour depth/bit depth*

Show understanding of how data for a vector graphic are encoded

Use the terms: *drawing object, property, drawing list*

Justify the use of a bitmap image or a vector graphic for a given task

Sound

Show understanding of how sound is represented and encoded

Use the terms: *sampling, sampling rate, sampling resolution, analogue and digital data*

Show understanding of the impact of changing the sampling rate and resolution

Impact on file size and accuracy

1.3 Compression

Show understanding of the need for and examples of the use of compression

Show understanding of lossy and lossless compression and justify the use of a method in a given situation

Including the use of run-length encoding (RLE)

Show understanding of how a text file, bitmap image, vector graphic and sound file can be compressed

Day 2 - Communication

2.1 Networks including the internet

Show understanding of the purpose and benefits of networking devices

Show understanding of the characteristics of a LAN (local area network) and a WAN (wide area network)

Explain the client-server and peer-to-peer models of networked computers

Show understanding of thin-client and thick-client and the differences between them

Show understanding of the bus, star, mesh and hybrid topologies

Show understanding of cloud computing

Show understanding of the differences between and implications of the use of wireless and wired networks

Describe the hardware that is used to support a LAN

Describe the role and function of a router in a network

Show understanding of Ethernet and how collisions are detected and avoided

Show understanding of bit streaming

Show understanding of the differences between the World Wide Web (WWW) and the internet

Describe the hardware that is used to support the internet

Roles of the different computers within the network and subnetwork models

Benefits and drawbacks of each model

Justify the use of a model for a given situation

Understand how packets are transmitted between two hosts for a given topology

Justify the use of a topology for a given situation

Including the use of public and private clouds.

Benefits and drawbacks of cloud computing
Describe the characteristics of copper cable, fibre-optic cable, radio waves (including WiFi), microwaves, satellites

Including switch, server, Network Interface Card (NIC), Wireless Network Interface Card (WNIC), Wireless Access Points (WAP), cables, bridge, repeater

Including Carrier Sense Multiple Access / Collision Detection (CSMA / CD)

Methods of bit streaming, i.e. real-time and on-demand

Importance of bit rates / broadband speed on bit streaming

Including modems, PSTN (Public Switched Telephone Network), dedicated lines, cell phone network

2.1 Networks including the internet continued

Explain the use of IP addresses in the transmission of data over the internet

Including:

- format of an IP address including IPv4 and IPv6
- use of subnetting in a network
- how an IP address is associated with a device on a network
- difference between a public IP address and a private IP address and the implications for security
- difference between a static IP address and a dynamic IP address

Explain how a Uniform Resource Locator (URL) is used to locate a resource on the World Wide Web (WWW) and the role of the Domain Name Service (DNS)

Day 3 - Hardware

3.1 Computers and their components

Show understanding of the need for input, output, primary memory and secondary (including removable) storage

Show understanding of embedded systems

Including: benefits and drawbacks of embedded systems

Describe the principal operations of hardware devices

Including: Laser printer, 3D printer, microphone, speakers, magnetic hard disk, solid state (flash) memory, optical disc reader/writer, touchscreen, virtual reality headset

Show understanding of the use of buffers

Including their use in a range of devices and systems

Explain the differences between Random Access Memory (RAM) and Read Only Memory (ROM)

Explain the differences between Static RAM (SRAM) and Dynamic RAM (DRAM)

Include their use in a range of devices and systems and the reasons for using one instead of the other depending on the device and its use

Explain the difference between Programmable ROM (PROM), Erasable Programmable ROM (EPROM) and Electrically Erasable Programmable ROM (EEPROM)

Including :

Show an understanding of monitoring and control systems

- difference between monitoring and control
- use of sensors (including temperature, pressure, infra-red, sound) and actuators
- importance of feedback

3.2 Logic Gates and Logic Circuits

Use the following logic gate symbols:



NOT



AND



OR



NAND



NOR



XOR

Understand and define the functions of :

NOT, AND, OR, NAND, NOR and XOR (EOR) gates

Construct the truth table for each of the logic gates above

Construct a logic circuit

Construct a truth table

Construct a logic expression

All gates except the NOT gate will have two inputs only.

From:

- a problem statement
- a logic expression
- a truth table

From:

- a problem statement
- a logic circuit
- a logic expression

From:

- a problem statement
- a logic circuit
- a truth table

Day 4 - Processor Fundamentals

4.1 Central Processing Unit (CPU) Architecture

Show understanding of the basic Von Neumann model for a computer system and the stored program concept

Show understanding of the purpose and role of registers, including the difference between general purpose and special purpose registers

Special purpose registers including:

- Program Counter (PC)
- Memory Data Register (MDR)
- Memory Address Register (MAR)
- The Accumulator (ACC)
- Index Register (IX)
- Current Instruction Register (CIR)
- Status Register

Show understanding of the purpose and roles of the Arithmetic and Logic Unit (ALU), Control Unit (CU) and system clock, Immediate Access Store (IAS)

Show understanding of how data are transferred between various components of the computer system using the address bus, data bus and control bus

Show understanding of how factors contribute to the performance of the computer system

Including:

- processor type and number of cores
- the bus width
- clock speed
- cache memory

Including connection to:

- Universal Serial Bus (USB)
- High Definition Multimedia Interface (HDMI)
- Video Graphics Array (VGA)

Understand how different ports provide connection to peripheral devices

Describe the stages of the Fetch-Execute (F-E) cycle

Describe and use 'register transfer' notation to describe the F-E cycle

Show understanding of the purpose of interrupts

Including:

- possible causes of interrupts
- applications of interrupts
- use of an Interrupt service (ISR) handling routine
- when interrupts are detected during the fetch-execute cycle
- how interrupts are handled

4.2 Assembly Language

7

Show understanding of the relationship between assembly language and machine code

Describe the different stages of the assembly process for a two-pass assembler

Trace a given simple assembly language program

Show understanding that a set of instructions are grouped

Show understanding of the different modes of addressing

Apply the two-pass assembler process to a given simple assembly language program

Including the following groups:

- Data movement
- Input and output of data
- Arithmetic operations
- Unconditional and conditional instructions
- Compare instructions

Including Immediate, direct, indirect, indexed, relativ

The following table is an example of an instruction set:

Instruction		Explanation
Opcode	Operand	
LDM	#n	Immediate addressing. Load the number n to ACC
LDD	<address>	Direct addressing. Load the contents of the location at the given address to ACC
LDI	<address>	Indirect addressing. The address to be used is at the given address. Load the contents of this second address to ACC
LDX	<address>	Indexed addressing. Form the address from <address> + the contents of the index register. Copy the contents of this calculated address to ACC
LDR	#n	Immediate addressing. Load the number n to IX
MOV	<register>	Move the contents of the accumulator to the given register (IX)
STO	<address>	Store the contents of ACC at the given address
ADD	<address>	Add the contents of the given address to the ACC
ADD	#n/Bn/&n	Add the number n to the ACC
SUB	<address>	Subtract the contents of the given address from the ACC
SUB	#n/Bn/&n	Subtract the number n from the ACC
INC	<register>	Add 1 to the contents of the register (ACC or IX)
DEC	<register>	Subtract 1 from the contents of the register (ACC or IX)
JMP	<address>	Jump to the given address
CMP	<address>	Compare the contents of ACC with the contents of <address>
CMP	#n	Compare the contents of ACC with number n
CMI	<address>	Indirect addressing. The address to be used is at the given address. Compare the contents of ACC with the contents of this second address
JPE	<address>	Following a compare instruction, jump to <address> if the compare was True
JPN	<address>	Following a compare instruction, jump to <address> if the compare was False
IN		Key in a character and store its ASCII value in ACC
OUT		Output to the screen the character whose ASCII value is stored in ACC
END		Return control to the operating system

All questions will assume there is only one general purpose register available (Accumulator)

ACC denotes Accumulator

IX denotes Index Register

<address> can be an absolute or symbolic address

denotes a denary number, e.g. #123

B denotes a binary number, e.g. B01001010

& denotes a hexadecimal number, e.g. &4A

4.3 Bit manipulation

Show understanding of and perform binary shifts

logical, arithmetic and cyclic

Left shift, right shift

Show understanding of how bit manipulation can be used to monitor / control a device

Carry out bit manipulation operations

Test and set a bit (using bit masking)

Label	Instruction		Explanation
	Opcode	Operand	
	AND	#n/Bn/&n	Bitwise AND operation of the contents of ACC with the operand
	AND	<address>	Bitwise AND operation of the contents of ACC with the contents of <address>
	XOR	#n/Bn/&n	Bitwise XOR operation of the contents of ACC with the operand
	XOR	<address>	Bitwise XOR operation of the contents of ACC with the contents of <address>
	OR	#n/Bn/&n	Bitwise OR operation of the contents of ACC with the operand
	OR	<address>	Bitwise OR operation of the contents of ACC with the contents of <address> <address> can be an absolute address or a symbolic address
	LSL	#n	Bits in ACC are shifted logically n places to the left. Zeros are introduced on the right hand end
	LSR	#n	Bits in ACC are shifted logically n places to the right. Zeros are introduced on the left hand end
<label>:	<opcode>	<operand>	Labels an instruction
<label>:		<data>	Gives a symbolic address <label> to the memory location with contents <data>

Day 5 - System Software

5.1 Operating System

Explain why a computer system requires an Operating System (OS)

Explain the key management tasks carried out by the Operating System

Show understanding of the need for typical utility software provided with an Operating System

Show understanding of program libraries

Including memory management, file management, security management, hardware management (input / output / peripherals), process management

Including disk formatter, virus checker, defragmentation software, disk contents analysis/disk repair software, file compression, back-up software

Including:

- software under development is often constructed using existing code from program libraries
- the benefits to the developer of software constructed using library files, including Dynamic Link Library (DLL) files

5.2 Language Translators

Show understanding of the need for:

- assembler software for the translation of an assembly language program
- a compiler for the translation of a high-level language program
- an interpreter for translation and execution of a high-level language program

Explain the benefits and drawbacks of using either a compiler or interpreter and justify the use of each

Show awareness that high-level language programs may be partially compiled and partially interpreted, such as Java

Describe features found in a typical Integrated Development Environment (IDE)

Including:

- for coding, including context-sensitive prompts
- for initial error detection, including dynamic syntax checks
- for presentation, including prettyprint, expand and collapse code blocks
- for debugging, including single stepping, breakpoints, i.e. variables, expressions, report window

Day 6 - Security, privacy & data integrity

6.1 Data Security

Explain the difference between the terms security, privacy and integrity of data

Show appreciation of the need for both the security of data and the security of the computer system

Describe security measures designed to protect computer systems, ranging from the stand-alone PC to a network of computers

Show understanding of the threats to computer and data security posed by networks and the internet

Describe methods that can be used to restrict the risks posed by threats

Describe security methods designed to protect the security of data

Including user accounts, passwords, authentication techniques such as digital signatures, firewall, anti-virus software, anti-spyware, encryption

Including malware (virus, spyware), hackers, phishing, pharming

Including encryption, access rights

6.2 Data Integrity

Describe how data validation and data verification help protect the integrity of data

Describe and use methods of data validation

Including range check, format check, length check, presence check, existence check, limit check, check digit

Describe and use methods of data verification during data entry and data transfer

During data entry including visual check, double entry

During data transfer including parity check (byte and block), checksum

Day 7 - Ethics and Ownership

7.1 Ethics and Ownership

Show understanding of the need for and purpose of ethics as a computing professional

Understand the importance of joining a professional ethical body including BCS (British Computer Society), IEEE (Institute of Electrical and Electronic Engineers)

Show understanding of the need to act ethically and the impact of acting ethically or unethically for a given situation

Show understanding of the need for copyright legislation

Show understanding of the different types of software licencing and justify the use of a licence for a given situation

Licences to include free Software Foundation, the Open Source Initiative, shareware and commercial software

Show understanding of Artificial Intelligence (AI)

Understand the impact of AI including social, economic and environmental issues

Understand the applications of AI

Day 8 - Databases

8.1 Database Concepts

Show understanding of the limitations of using a file-based approach for the storage and retrieval of data

Describe the features of a relational database that address the limitations of a file-based approach

Show understanding of and use the terminology associated with a relational database model

Use an entity-relationship (E-R) diagram to document a database design

Show understanding of the normalisation process

Explain why a given set of database tables are, or are not, in 3NF

Produce a normalised database design for a description of a database, a given set of data, or a given set of tables

Including entity, table, record, field, tuple, attribute, primary key, candidate key, secondary key, foreign key, relationship (one-to-many, one-to-one, many-to-many), referential integrity, indexing

First Normal Form(1NF), Second Normal Form (2NF) and Third Normal Form (3NF)

8.2 Database Management System (DBMS)

Show understanding of the features provided by a Database Management System (DBMS) that address the issues of a file based approach

Including:

- data management, including maintaining a data dictionary
- data modelling
- logical schema
- data integrity
- data security, including backup procedures and the use of access rights to individuals / groups of users

Show understanding of how software tools found within a DBMS are used in practice

Including the use and purpose of:

- developer interface
- query processor

8.3 Data Definition Language (DDL) and Data Manipulation Language (DML)

Show understanding that DBMS carries out all creation / modification of the database structure using its Data Definition Language (DDL)

Show understanding that the DBMS carries out all queries and maintenance of data using its DML

Show understanding that the industry standard for both DDL and DML is Structured Query Language (SQL)

Understand given SQL (DDL) commands and be able to write simple SQL (DDL) commands using a sub-set of commands

Understand a given SQL script

Create a database (CREATE DATABASE)

Create a table definition (CREATE TABLE), including the creation of attributes with appropriate data types:

- CHARACTER
- VARCHAR(n)
- BOOLEAN
- INTEGER
- REAL
- DATE
- TIME

change a table definition (ALTER TABLE)

add a primary key to a table (PRIMARY KEY (field))

add a foreign key to a table (FOREIGN KEY (field) REFERENCES Table (Field))

Queries including SELECT... FROM, WHERE, ORDER BY, GROUP BY, INNER JOIN, SUM, COUNT, AVG

Write an SQL script to query or modify data (DML) which are stored in (at most two) database tables

Data maintenance including. INSERT INTO, DELETE FROM, UPDATE

Day 9 - Algorithm Design and Problem Solving

Refer to Pseudocode Guide www.cambridgeinternational.org/support

9.1 Computational Thinking Skills

Show an understanding of abstraction

Need for and benefits of using abstraction

Describe the purpose of abstraction,
Produce an abstract model of a system by
only including essential details

Describe and use decomposition

Break down problems into sub-problems leading to
the concept of a program module (procedure /
function)

9.2 Algorithms

Show understanding that an algorithm is a solution to a
problem expressed as a sequence of defined steps

Use suitable identifier names for the representation of
data used by a problem and represent these using an
identifier table

Write pseudocode that contains input, process and
output

Write pseudocode using the three basic constructs of
sequence, selection and iteration (repetition)

Document a simple algorithm using pseudocode

Write pseudocode from:

- a structured English description
- a flowchart

Describe and use the process of stepwise refinement
to express an algorithm to a level of detail from which the
task may be programmed

Use logic statements to define parts of an algorithm
solution

Day 10 - Data Types and structures

10.1 Data Types and Records

Select and use appropriate data types for a problem solution

Show understanding of the purpose of a record structure to hold a set of data of different data types under one identifier

10.2 Arrays

Use the technical terms associated with arrays

Select a suitable data structure (1D or 2D array) to use for a given task

Write pseudocode for 1D and 2D arrays

Write pseudocode to process array data

10.3 Files

Show understanding of why files are needed

Write pseudocode to handle text files that consist of one or more lines

10.4 Introduction to Abstract Data Types (ADT)

Show understanding that an ADT is a collection of data and a set of operations on those data

Show understanding that a stack, queue and linked list are examples of ADTs

Use a stack, queue and linked list to store data

Describe how a queue, stack and linked list can be implemented using arrays

including integer, real, char, string, Boolean, date (pseudocode will use the following data types: INTEGER, REAL, CHAR, STRING, BOOLEAN, DATE, ARRAY, FILE)

Write pseudocode to define a record structure.

Write pseudocode to read data from a record structure and save data to a record structure

Including index, upper and lower bound

Sort using a bubble sort

Search using a linear search

Describe the key features of a stack, queue and linked list and justify their use for a given situation

Candidates will not be required to write pseudocode for these structures, but they should be able to add, edit and delete data from these structures

Day 11 - Programming

11.1 Programming

Basics

Implement and write pseudocode from a given design presented as either a program flowchart or structured English

Write pseudocode statements for:

- the declaration of variables and constants
- the assignment of values to variables and constants
- expressions involving any of the arithmetic or logical operators input from the keyboard and output to the console

Use built-in functions and library routines

Any functions not given in the pseudocode guide will be provided

String manipulation functions will always be given

11.2 Constructs

Use pseudocode to write:

- an 'IF' statement including the 'ELSE' clause and nested IF statements
- a 'CASE' structure
- a 'count-controlled' loop:
- a 'post-condition' loop
- a 'pre-condition' loop

Justify why one loop structure may be better suited to solve a problem than the others

11.3 Structured Programming

Define and use a procedure

Explain where in the construction of an algorithm it would be appropriate to use a procedure

Use parameters

A procedure may have none, one or more parameters

A parameter can be passed by reference or by value

Define and use a function

Explain where in the construction of an algorithm it is appropriate to use a function

Use the terminology associated with procedures and functions

Write efficient pseudocode

A function is used in an expression, e.g. the return value replaces the call

including Procedure / function header, procedure / function interface, parameter, argument, return value

Day 12 - Software Development

12.1 Program Development Life cycle

Show understanding of the purpose of a development life cycle

Including, waterfall, iterative, rapid application development (RAD)

Show understanding of the need for different development life cycles depending on the program being developed

Describe the principles, benefits and drawbacks of each type of life cycle

Show understanding of the analysis, design, coding, testing and maintenance stages in the program development life cycle

12.2 Program Design

Use a structure chart to decompose a problem into sub-tasks and express the parameters passed between the various modules / procedures / functions which are part of the algorithm design

Describe the purpose of a structure chart Construct a structure chart for a given problem Derive equivalent pseudocode from a structure chart

Show understanding of the purpose of state-transition diagrams to document an algorithm

12.3 Program Testing and maintenance

Show understanding of ways of exposing and avoiding faults in programs

Locate and identify the different types of errors

- syntax errors
- logic errors
- run-time errors

Correct identified errors

Show understanding of the methods of testing available and select appropriate data for a given method

Including dry run, walkthrough, white-box, black-box, integration, alpha, beta, acceptance, stub

Show understanding of the need for a test strategy and test plan and their likely contents

Choose appropriate test data for a test plan

Including normal, abnormal and extreme/boundary
Including perfective, adaptive, corrective

Show understanding of the need for continuing maintenance of a system and the differences between each type of maintenance

Analyse an existing program and make amendments to enhance functionality

Gr 12 Crash Course Content (Information Technology -9626)

12 IT in society	(Day 1)
12.1 Digital currencies	
12.2 Data mining	
12.3 Social networking services/platforms	
12.4 The impact of IT	
12.5 Technology-enhanced learning	
13 New and emerging technologies	(Day 2)
13.1 New and emerging technologies	
14 Communications technology	(Day 3)
14.1 Networks	
14.2 Components in a network	
14.3 Network servers	
14.4 Cloud computing	
14.5 Data transmission across networks	
14.6 Routing data in networks	
14.7 Network protocols	
14.8 The management of network traffic	
14.9 Data transmission methods	
14.10 Wireless technology	
14.11 Mobile communication systems	
14.12 Network security	
14.13 Disaster recovery management	
15 Project management	(Day 4)
15.1 The stages of the project life cycle	
15.2 Project management software	
15.3 Tools and techniques for project management tasks	
16 System life cycle	(Day 5)
16.1 The stages in the system life cycle	
16.2 Analysis	
16.3 Design	
17 Mail Merge	(Day 6)
17.1 Mail merge	
18 Graphics creation	(Day 7)
18.1 Common graphics skills	
18.2 Vector graphics	
18.3 Bitmap images	
18.4 Compression	
18.5 Text	
19 Animation	(Day 8)
19.1 Animation	
20 Programming for the web	(Day 9)
20.1 Programming for the web	

Day 1 - IT in society

12.1 Digital currencies

- Types
 - Characteristics
 - Advantages and disadvantages
 - Uses
-
- Impact and risks of digital currencies

Including:

Digital/electronic currency, virtual currency, cryptocurrency, central bank digital base money, stored value cards

Centralised systems (debit, credit cards, electronic point of sale), decentralised systems (e.g. Bitcoin, Litecoin, peer-to-peer electronic monetary systems), mobile electronic wallets

Impact upon/risks for individuals, businesses, governments, global economy

12.2 Data mining

- Process of data mining
-
- Uses
-
- Advantages and disadvantages

Process including:

- 1 Business understanding
- 2 Data understanding
- 3 Data preparation
- 4 Data modelling
- 5 Evaluation
- 6 Deployment

Uses including: how and why data mining is used in national security, surveillance, businesses, scientific research, health care, and the analysis of social and economic trends

Including ethical and privacy concerns

12 IT in society (continued)

12.3 Social networking services/platforms

- Types
- Uses
- The impact of social networking
- Advantages and disadvantages of different types of social networking

Types including:

Chat rooms, instant messaging, forums, email, blogs, microblogs, social media

Uses including:

- use by individuals, businesses, organisations, governments
- use in, e.g. education, finance, health care, for creating and sharing information, news sources

Including impact on:

- individuals (e.g. intellectual isolation, physical and mental health of different age groups, ideological polarisation, stereotyping, cognitive issues)
- businesses (e.g. advertising), organisations (e.g. use by police to disseminate information, weather warnings)
- governments (e.g. distribution of information, censorship), false/distorted information

12.4 The impact of IT

- On society
- On monitoring and surveillance in society

Including on:

- sport, manufacturing, health care, education, banking, e-business and finance, news and media, family and home, entertainment and news, government, politics
- including policing

12.5 Technology enhanced learning

- Methods of delivery
- Impact
- Advantages and disadvantages of different methods of delivery

Including:

computer-based training
online
tutorials
networked courses

Massive Open Online Courses (MOOC)
video-conferencing

Including impact of different delivery methods on student achievement, learner autonomy, student and teacher motivation

Day 2 - New and emerging technologies

13.1 *New and emerging technologies*

- Types

Including:

NFC (Near Field Communication)

ultra-high definition

television systems artificial

intelligence

augmented

reality virtual

reality robotics

computer-assisted

translation

holographic imaging

holographic and 4th generation optical data storage

3D printing

vision enhancement

wearable computing

Including on:

individuals and

their lifestyles

organisations

medicine and health care (including: development of prosthetics and medical products, tissue engineering, artificial blood vessels, the design of medical tools and equipment)

scientific research

the environment (including: e-waste, recycling, power consumption, manufacturing processes)

- Impact of new and emerging technologies

Day 3 - Communications technology

14.1 Networks

- Types
- Characteristics and uses of each type
- Advantages and disadvantages of each type

Including: local area network (LAN), wide area network (WAN), client-server, peer-to-peer, Virtual Private Network (VPN), mobile networks

Characteristics including: protocols associated with each type of network such as those used to increase privacy and security (e.g. tunneling, transport layer security (SSL/TLS), Internet Protocol Security), BitTorrent for transferring large files

Uses including:

sharing and storage

of resources sharing

of peripherals

exchange of data

access to internet

services access to

telephony

services

access to content delivery services such as those used for, e.g. video-streaming and software downloads

14.2 Components in a network

- The role of components in a network
- The operations of networking components

The role of components including:

network interface cards and wireless network

interface cards repeaters, hubs, and switches

wireless access points, gateways, bridges

and routers firewalls (hardware and

software)

Including:

how each component carries out its role

how each component works with the others in a network

14.3 Network servers

- Types of network server
- The role of servers in a network
- The operations of servers in a network
- Advantages and disadvantages of each type of server for a given scenario

Including: file server, web server, mail server, applications server, print server, FTP server, proxy server, virtual server

Including: the function of servers in a network

Including:

the 'request and response' method of communication between servers and clients

within server farms

14.4 *Cloud computing*

- Characteristics of cloud computing
- Uses of cloud computing
- Advantages and disadvantages of cloud computing for a given scenario

Including: the key characteristics of cloud computing for sharing computing resources

Including: by individuals and organisations

14.5 *Data transmission across networks*

- Speed of transmission: bandwidth, bit rate
- Data streaming

Including:

how bandwidth is defined

the bandwidths made available by: different transmission media, different internet access technologies, e.g. ethernet, fibre optic, wireless, mobile communications

how bit rates are quantified and measured

Including:

real time and on demand media streams

the impact of bit rate and bandwidth on the streaming of audio and video data

14.6 *Routing data in networks*

- Methods of sending data over a network:
 - packet switching
 - circuit switching
 - message switching
- The purpose and use of network addressing systems

Including: the structure of packets (to include what is contained in the packets)

The modes of connection (to include how and why they are used). Including: connection mode e.g. Frame Relay, Transmission Control Protocol (TCP), connectionless (datagram) mode, e.g. Ethernet, Internet Protocol (IP), User Datagram Protocol (UDP)

Including: the use of communication channels in circuit switching

Including: the store and forward method of sending messages across networks

Including: Media Access Control (MAC) addressing, IP addressing (IP4 and IP6)

14.7 Network protocols

- The definition of a protocol
- The purposes and uses of protocols in the preparation, addressing, sending and receiving of data across networks including the internet

Including: explaining the term ‘protocol’ as used in networking and why protocols are necessary

Protocols including:

Transmission Control Protocol (TCP) Internet Protocol (IP)
 Internet Control Message Protocol (ICMP)
 Address Resolution Protocol (ARP) and Inverse Address Resolution Protocol (InARP)
 Dynamic Host Configuration Protocol (DHCP) User Datagram protocol (UDP)
 Hypertext Transfer Protocols (HTTP and HTTPS) File Transfer Protocol (FTP)
 Tunneling protocol, e.g. L2TP
 Simple Mail Transfer Protocol (SMTP) Post Office Protocols (e.g. POP3)
 Internet Message Access Protocol (IMAP) Telnet
 SSH
 TLS/SSL

14.8 The management of network traffic

- Static and dynamic routing
- Function of routing protocols
- Use of protocol layering

Including: the selection of paths for network traffic and the use of routing tables

Including: interior gateway protocols, exterior gateway protocols, border gateway protocols

Including: TCP/IP and Open Systems Interconnection model (OSI model), the function of each layer in the TCP/IP suite, the function of each layer in the OSI model, comparison of the schemes

14.9 Data transmission methods

- The properties, features and characteristics of different transmission methods
- Typical applications of each method
- Advantages and disadvantages of each method

Including: fibre optic, copper cables (coaxial and twisted pair), using lasers

Including: the effect of the medium on available bandwidth for data transmission

14.10 Wireless technology

- Methods of wireless transmission of data
- Uses and operation of wireless transmission methods
- Advantages and disadvantages of wireless transmission methods

Including: Wi-fi, Bluetooth, infrared, microwave and radio

Including: security issues associated with wireless transmission and wireless security protocols such as WEP, WPA and their variants

Including the benefits and limitations of each method of wireless transmission

14.11 Mobile communication systems

- Cellular networks
- How satellite communication systems are used for transferring data

Including:

structure of a cellular network

use of 3G, 4G, 5G systems for mobile communications

Including:

in global positioning systems (GPS), in global mapping systems, in surveillance, in telecommunications (e.g. television and radio broadcasting, telephones)

how communications data is prepared, sent and received by satellite communication systems

14.12 Network security

- Networking security threats to stored data and files
- Impact of network security threats on individuals and organisations
- Prevention of network security issues using physical and software methods
- Advantages and disadvantages of the various methods

Including: botnets that attack systems, malware with actions that harm computer users, data, files and systems

Including: data destruction, manipulation and modification and theft by unauthorised users, identity theft

Including:

How physical methods can prevent unauthorised access, Including:

use of barriers, locks, surveillance, alarm systems, security guards

How software methods can prevent unauthorised access, Including:

biometric methods, anti-malware, anti-virus and anti-spyware software for protecting data, files and systems encryption, access rights/permissions for protecting data and files

how (hardware and software) firewalls can be configured to control access to networks

For preventing unauthorised access to computer data, files and systems

14.13 Disaster recovery management

- Identification of threats and risks
- Control of threats
- Strategies to minimise risks

Including: risk analysis, perpetrator analysis, risk testing, quantifying the risk

Including:

how to detect threats and prevent disaster how to restore after a disaster

Including:

use of protection for power supplies
 use of password and access controls for data and file protection
 protection of data and software from malware, unauthorised access
 use of backup strategies

Day 4 - Project management

15.1 The stages of the project life cycle

- Project initiation
- Project planning
- Project execution and monitoring
- Project close

Including: identifying objectives, scoping project, stakeholders, resources required and high-level schedules

Including detailed planning of resources and scheduling of tasks

Including: implementing the plan; monitoring progress against time, cost and quality, reporting to stakeholders

Including: project completion and review

Candidates should understand each stage has a phase review and why the reviews are conducted

15.2 Project management software

- Uses for supporting projects
- Strengths and weaknesses of project management software for supporting projects

Including supporting planning, scheduling of tasks, allocation of resources, costings, communication, and collaborative working and decisions

15.3 Tools and techniques for project management tasks

- Gantt charts

Including:

creating a Gantt chart
using and interpreting Gantt charts

- Performance Evaluation and Review Technique (PERT) and Critical Path Method (CPM)

Including:

creating PERT charts
using and interpreting PERT charts for the analysis and management of projects
components such as activities/tasks/work breakdown structures (WBS), timings, float, end points, milestones, dependencies, deliverables
critical path calculations and determinations to determine the longest path of dependent activities
the use of critical path analysis for, e.g. work flow control such as authorising work, costings, allocating resources

Day 5 - System life cycle

16.1 The stages in the system life cycle

Candidates should be able to explain the relationship between the different stages of the system life cycle.

16.2 Analysis

- Methods of researching for a given a situation
- Content and purpose of specifications

Including: questionnaires, interviews, observation, document analysis

Including: user requirements specification, system specification, design specification

16.3 Design

- System processing
- Flow of data through system
- Data storage
- Input forms
- Output reports

Candidates will be expected to construct a system flowchart and a data flow diagram (DFD) using correct symbols. See symbols at the end of the Subject content section.

Including: databases, files (input and output)

Including: features and elements of forms, appropriate use of forms for data collection, validation and checking of data collected by forms, input screen layouts

Including: output screen layouts, printed copy layouts

16.4 Development and testing

- Test plans Including: the need for testing and the purpose of a test plan, contents of a test plan
- Test data Including: the types and purpose of test data
- Alpha and beta testing Including: differences between alpha and beta testing
- White box and black box testing Including: differences between white box and black box testing
Candidates are expected to be able to create a test plan for a given situation.

16.5 Implementation

- Methods of implementing a system Including: parallel running, direct changeover, phased implementation, pilot implementation
- Advantages and disadvantages of each implementation method for a given situation Including: how each method is implemented in a given situation, the suitability of an implementation method for a given situation

16.6 Documentation

- Types of documentation and why each is needed Including: user and technical documentation
- Contents of the documentation

16.7 Evaluation

- Methods of evaluating a new system Including: in terms of efficiency, ease of use and meeting user requirements

16.8 Maintenance

- Types of maintenance and why each is needed Including: perfective, adaptive, preventive, corrective maintenance
- How each type of maintenance is carried out

16.9 Prototyping

- Types of prototyping and why each is needed Including: evolutionary, incremental, throw-away, rapid
- Advantages and disadvantages of each type of prototyping

16.10 Methods of software development

- Stages/processes of each method
- Advantages and disadvantages of each method

Including: Agile, Iterative, Incremental, Rapid Application Development (RAD), 'Waterfall' methods

Day 6 - Mail merge

17.1 Mail merge

Use/create/edit a source data using appropriate software

Create a master document structure

Link a master document to a source file

Specify rules

- for selecting recipients
- for managing document content

Set up fields

- for manual completion
- for automatic completion
- calculated fields

Use manual methods and software tools to ensure error-free accuracy

Perform mail merge

Including: standard letter, labels

Including: identifying and using correct field names, using conditional operators, embedding a chart/table, update links

Including: edit/sort/filter the mail merge recipient list, SKIP IF Including conditional fields, IF THEN ELSE

Including: create appropriate prompts to the user for manual completion, FILL IN, ASK

Including: date field, time, document properties

Using arithmetic operators

Including proof reading

Including:

using the master document and data sources
control record selection and omission when merging

18

Graphics creation

18.1 Common graphics skills •

Work with layers

- Use transform tools
- Use grouping or merging tools
- Use alignment and distribution tools

- Use layout tools
- Use colour picker tools

- Use crop tools

Know and understand the properties of different colour systems

Know and understand the need for different image resolutions

Know, understand and select different bitmap and vector file formats

Export an image in different file formats

Change the opacity of all or part of an image

Including: raise, lower, bring to front, order layers Including: resize, skew, flip, rotate, move, scale, shear Including: group, ungroup, flatten Including: left, right, top and bottom alignment and vertical and horizontal distribution. Including: rulers, grids, guidelines, snapping Candidates should be able to select a colour within an image or from other sources

(RGB, HSL, CMYK, CMS)

Including the impact of having too low/high a resolution of an image on screen or in print

Including their suitability for use in a given scenario

Including: svg, bmp, jpg, png, gif, tif, pdf

18.2 Vector graphics

Create a vector graphic that meets the requirements of its intended application and audience

- Use vector drawing tools

- Use selection tools to select parts of a vector graphic
- Use fill tools to colour elements
- Use node and path editing

- Convert bitmap images into editable vector shapes

Including: freehand drawing, Bezier curves and straight lines

Shape tools – creation of rectangles, ellipses, circles, arcs, stars, polygons and spirals

Including gradient fills

Including adding and moving nodes Deleting nodes to simplify paths

Using Bezier handles

Align and distribute nodes

Including trace bitmaps

Candidates should know and understand the advantages and disadvantages of converting a bitmap image into an editable vector shape

18

Graphics creation (continued)

18.3 Bitmap images

Create a bitmap image that meets the requirements of its intended application and audience

- Use selection tools to select parts of a bitmap image
- Adjust colour levels
- Use tools/filters to alter parts of an image
- Resize an image/canvas

Including lasso, magic wand and colour select tools

Including brightness, contrast and colour balance, shadows, highlights, convert to greyscale

Including distort, clone, blur, sharpen and red eye removal

Including: lossy and lossless compression, changing colour depth, changing resolution

18.4 Compression

- Know and understand the effects of different methods of compression on images

18.5 Text

- Select font style
- Fit text to path or shape
- Set text in a shape
- Convert text to curves

Including: font face, size, kerning, letters spacing and line spacing

Aligning text along a line or around a shape

Convert fonts into editable vector shapes

19

Animation

19.1 Animation

Create an animation (stop motion and key frame) that meets the requirements of its intended application and audience

- Configure the stage/frame/canvas for an animation
- Import and create Vector objects
- Control object properties
- Use Inbetweening ('Tweening') tools
- Set paths
- Use layers
- Apply masks

Control animations

Candidates should know and understand:

- The basic principles of animation
- Different animation types and methods
- The use of animation variables when creating animations

Including:

setting colour, size and the aspect ratio using rulers, guides, and grid settings
setting snapping options

Including tracing bitmaps and adding text

Such as:

stroke and fill settings
size position and orientation
transparency

To:

show motion
show shape, size and colour changes

By:

adjusting frame rates
looping or stopping animations

Including:

Frames, key frames and property key frames

Timings

Coordinates

Inbetweening and what effect it creates

Morphing and what effect it creates

Including: cell animation, stop motion, time lapse, flip book, CGI, 2D, 3D

Including:

the primary, e.g. graphical elements, and secondary, e.g. sound, components of animation

the use of animation variables to control the position of an animated object or parts of an animated object

Day 7 - Programming for the web

It is recommended that for this section of the syllabus, candidates should have a working knowledge of html and css, (for example, have studied website authoring in Cambridge IGCSE Information and Communication Technology, syllabus 0417).

20.1 Programming for the web

Use JavaScript to

- Add interactivity to web pages
- Change HTML content
- Change HTML styles
- Show/Hide HTML elements
- Display data in different ways
- React to common HTML Events
- Provide user interaction
- Create statements
- Use JavaScript loops for iterative methods

Including by:

inserting JavaScript in HTML
creating and using external scripts

Including:

text/number, including calculations and string
manipulation images

Change style of HTML elements using

```
document.getElementById(id).style.property = new style
```

Using

```
document.getElementById(id).style.visibility=  
document.getElementById(id).style.display=
```

Including by:

writing into an HTML element, using
innerHTML writing into the HTML
output using document.write() writing
into an alert box, using window.alert()
writing into the browser console, using
console.log() Including: onload,
onchange, onclick, onmouseover,
onmouseout, onkeydown
Using confirm() and prompt() popups

Composed of:

values (literals and variables)
operators (assignment, arithmetic,
algebraic, string, comparison, logical,
conditional, type)
while loop

20 Programming for the web (continued)

- Create functions
- Use JavaScript timing events
- Add comments to annotate and explain code

To be executed:

when an event
occurs when
invoked from
code
automatically
(self-invoked)

Execution interval methods:

setTimeout()
setInterval()

Including: single line and multi-line

- The structure and syntax of JavaScript code
- A range of object-based JavaScript programming techniques and terms

Including:

JavaScript statements

(composed of values,
operators, expressions,
keywords and comments)

Data types

(including: number, string,
Boolean, array, object) Type
conversions

Variables and arrays

JavaScript operators

(including: arithmetic,
assignment, string) Comparison
operators

(equal to, equal value and
equal type, not equal to, not
equal value and not equal
type, greater than, less than,
greater than or equal to, less
than or equal to)

Logical operators

(AND, OR, NOT)

Conditional operators

(if, else, else if, switch)

Ternary operator

Loops

(for, for/in, while, do/while)

Break

Interaction: alert(), prompt(),
confirm() Functions

Execution interval methods:
setTimeout(), setInterval()