

OUTLINE SCHEME OF WORK TRACKER – Y10 and Y11

SECTION
<p>1.1 Systems architecture</p> <p>The purpose of the CPU</p> <p>Von Neumann architecture:</p> <ul style="list-style-type: none">MAR (Memory Address Register)MDR (Memory Data Register)Program CounterAccumulator <p>Common CPU components and their function:</p> <ul style="list-style-type: none">ALU (Arithmetic Logic Unit)CU (Control Unit)Cache <p>the function of the CPU as fetch and execute instructions stored in memory</p> <p>How common characteristics of CPUs affect their performance:</p> <ul style="list-style-type: none">clock speedcache sizenumber of cores <p>Embedded systems:</p> <ul style="list-style-type: none">purpose of embedded systemsexamples of embedded systems.
<p>1.2 Memory</p> <p>Learners should have studied the following:</p> <ul style="list-style-type: none">the difference between RAM and ROMthe purpose of ROM in a computer systemthe purpose of RAM in a computer systemthe need for virtual memoryflash memory.
<p>1.3 Storage</p> <p>Learners should have studied the following:</p> <ul style="list-style-type: none">the need for secondary storagedata capacity and calculation of data capacity requirements

common types of storage:

- optical
- magnetic
- solid state

suitable storage devices and storage media for a given application, and the advantages and disadvantages of these, using characteristics:

- capacity
- speed
- portability
- durability
- reliability
- cost.

1.4 Wired and wireless networks

Learners should have studied the following:

types of networks:

- LAN (Local Area Network)
- WAN (Wide Area Network)

factors that affect the performance of networks

the different roles of computers in a client-server and a peer-to-peer network

the hardware needed to connect stand-alone computers into a Local Area Network:

- wireless access points
- routers/switches
- NIC (Network Interface Controller/Card)
- transmission media

the internet as a worldwide collection of computer networks:

- DNS (Domain Name Server)
- Hosting
- the cloud

the concept of virtual networks.

1.5 Network topologies, protocols and layers

Learners should have studied the following:

star and mesh network topologies

Wifi:

- frequency and channels

encryption ethernet

the uses of IP addressing, MAC addressing, and protocols including:

TCP/IP (Transmission Control Protocol/Internet Protocol)

HTTP (Hyper Text Transfer Protocol)

HTTPS (Hyper Text Transfer Protocol Secure)

FTP (File Transfer Protocol)

POP (Post Office Protocol)

IMAP (Internet Message Access Protocol)

SMTP (Simple Mail Transfer Protocol)

the concept of layers

packet switching.

1.6 System security

Learners should have studied the following:

forms of attack

threats posed to networks:

malware

phishing

people as the 'weak point' in secure systems (social engineering)

brute force attacks

denial of service attacks

data interception and theft

the concept of SQL injection

poor network policy

identifying and preventing vulnerabilities:

penetration testing

network forensics

network policies

anti-malware software

firewalls

user access levels

passwords

encryption.

1.7 Systems software

Learners should have studied the following:

the purpose and functionality of systems software

operating systems:

user interface

memory management/multitasking

peripheral management and drivers

user management

file management

utility system software:

encryption software

defragmentation

data compression

the role and methods of backup:

n full

n incremental.

1.8 Ethical, legal, cultural and environmental concerns

Learners should have studied the following:

how to investigate and discuss Computer Science technologies while considering:

ethical issues

legal issues (see lesson 1 slides)

cultural issues

environmental issues.

privacy issues.

how key stakeholders are affected by technologies

environmental impact of Computer Science

cultural implications of Computer Science

open source vs proprietary software

legislation relevant to Computer Science:

The Data Protection Act 1998

Computer Misuse Act 1990

Copyright Designs and Patents Act 1988

Creative Commons Licensing

Freedom of Information Act 2000.

2.1 Algorithms

Learners should have studied the following:

computational thinking:

abstraction

decomposition

algorithmic thinking

standard searching algorithms:

binary search

linear search

standard sorting algorithms:

bubble sort

merge sort

insertion sort

how to produce algorithms using:

pseudocode

using flow diagrams

interpret, correct or complete algorithms.

Learners should have studied the following:

the use of variables, constants, operators, inputs, outputs and assignments

the use of the three basic programming constructs used to control the flow of a program:

sequence

selection

iteration (count and condition controlled loops)

the use of basic string manipulation

the use of basic file handling operations:

open

read

write

close

the use of records to store data

the use of SQL to search for data

the use of arrays (or equivalent) when solving problems, including both one and two dimensional arrays

how to use sub programs (functions and procedures) to produce structured code

the use of data types:

- integer

- real

- Boolean

- character and string

- casting

the common arithmetic operators

the common Boolean operators.

2.3 Producing robust programs

Learners should have studied the following:

defensive design considerations:

- input sanitisation/validation

- planning for contingencies

- anticipating misuse

- authentication

maintainability:

- comments

- indentation

the purpose of testing

types of testing:

- iterative

- final/terminal

how to identify syntax and logic errors

selecting and using suitable test data.

2.4 Computational logic

Learners should have studied the following:

why data is represented in computer systems in binary form

simple logic diagrams using the operations AND, OR and NOT

truth tables

combining Boolean operators using AND, OR and NOT to two levels

applying logical operators in appropriate truth tables to solve problems

applying computing-related mathematics:

+
-
/
*
Exponentiation (^)
MOD
DIV

2.5 Translators and facilities of languages

Learners should have studied the following:

- characteristics and purpose of different levels of programming language, including low level languages
- the purpose of translators
- the characteristics of an assembler, a compiler and an interpreter
- common tools and facilities available in an integrated development environment (IDE):
 - editors
 - error diagnostics
 - run-time environment
 - translators.

2.6 Data representation

Learners should have studied the following:

Units

- bit, nibble, byte, kilobyte, megabyte, gigabyte, terabyte, petabyte
- how data needs to be converted into a binary format to be processed by a computer.

Numbers

- how to convert positive denary whole numbers (0–255) into 8 bit binary numbers and vice versa
- how to add two 8 bit binary integers and explain overflow errors which may occur
- binary shifts
- how to convert positive denary whole numbers (0–255) into 2 digit hexadecimal numbers and vice versa
- how to convert from binary to hexadecimal equivalents and vice versa
- check digits.

Characters

the use of binary codes to represent characters

the term 'character-set'

the relationship between the number of bits per character in a character set and the number of characters which can be represented (for example ASCII, extended ASCII and Unicode).

Images

how an image is represented as a series of pixels represented in binary

metadata included in the file

the effect of colour depth and resolution on the size of an image file.

Sound

how sound can be sampled and stored in digital form

how sampling intervals and other factors affect the size of a sound file and the quality of its playback:

- sample size

- bit rate

- sampling frequency.

Compression

need for compression

types of compression:

- lossy

- lossless.

Practical Lessons

Practical Skills Lessons	Year
Python Lessons	Y8
Python Exercises	Y8/9
LMC Lessons	Y11
Java lessons	Y10
HTML & CSS Lessons	Y8/10
Python Project - Maths Test Project	Y8
Python Project - Currency Converter	Y9
Python Project - Album Project	Y9
Mock NEA 1 - Mobile Device Diagnostic Program	Y10
NEA Prep (for actual NEA)	Y11
NEA Prep (General)	Y10
NEA	Y11