# Knowledge Organiser – Year 10 – Computer Science Principles

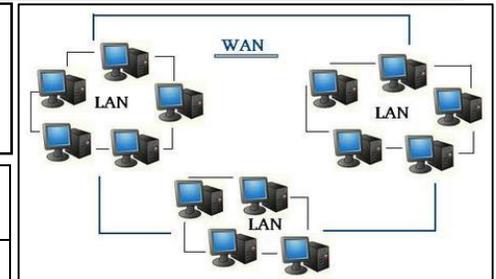| Secondary Storage • Magnetic Storage • Optical Storage • Solid State	used to retain a copy of programs and data that need to be kept long term.
	is a data storage device that uses storage to store and retrieve digital information such as a Hard Disk Drives (HDD)
	saves data as patterns of dots that can be read using light. A laser beam is the usual light source, such as a CD-ROM, DVD-ROM or Blu-Ray.
	means it has "no moving parts", this is a type of non-volatile storage, it retains its information even without power. So it is ideal for devices such as cameras, solid state drives, memory sticks and so on.
Characteristics of storage media	Capacity (Amount of memory), Speed of transferring data, Portability, Durability, Reliability, Cost
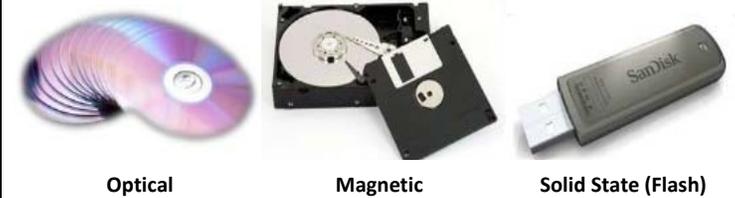
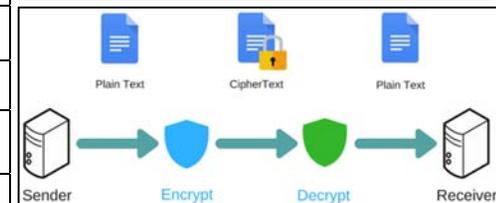**Optical   Magnetic   Solid State (Flash)**
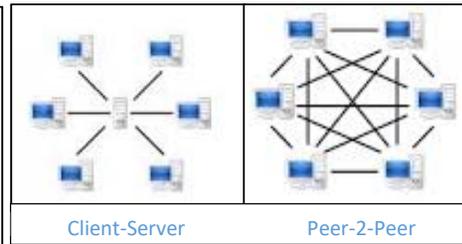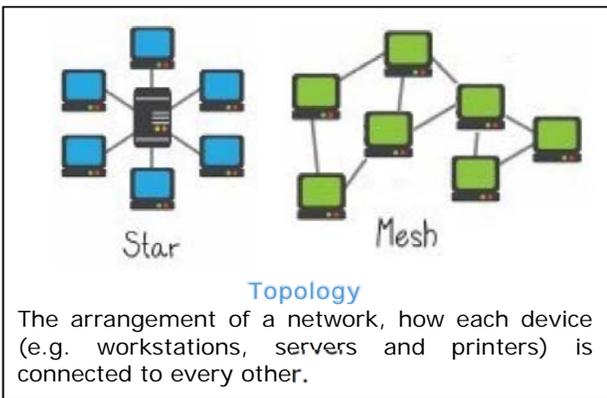
LAN	(Local Area Network), is a computer network that spans a relatively small area (1 office, school, etc.)
WAN	(Wide Area Network), is a computer network that covers a large geographical area.
Peer 2 Peer	is one in which two or more PCs share files and access to devices such as printers without requiring a separate server
Client-server	a server provides resources and services to one or more clients (personal computers, printers and other devices)

Client-Server   Peer-2-Peer

Topology
The arrangement of a network, how each device (e.g. workstations, servers and printers) is connected to every other.

IP Address
37.153.62.136
Based on the devices location and can change

MAC Address
98-81-55-CD-F2-2F
Fixed and assigned to the hardware

TCP/IP layer model
each layer is responsible for a small part of the communication process when sending/ receiving date across a network

Application Layer	Turning data into websites and other applications
Transport Layer	Controlling data flow – e.g. splitting data into packets and checking packets are correctly sent and delivered
Internet/ Network Layer	Making connections between networks, directing data packets and handling traffic. Used by routers.
Data Link Layer	Passing data over the physical network. Responsible for how bits are sent as electrical signals over cables, wireless and network hardware.

TCP	Transmission control protocol	Sets the rules for how the devices connect to the network. It is in charge of splitting the data into packets and reassembling them when received.
IP	Internet Protocol	Responsible for packet switching
HTTP	Hyper text transfer Protocol	Used by web browsers to access websites and communicate with web servers
HTTPS	HTTP secure	A more secure version of HTTP – encrypts all information sent and received.
FTP	File transfer Protocol	Used to access, edit and move files between devices on a network.
POP3	Post Office Protocol – version 3	Used to retrieve emails from a server. Once downloaded, it is deleted from the server.
IMAP	Instant Message access protocol	Used to retrieve emails from a server. You download a copy, until you actually delete it.
SMTP	Simple Mail transfer protocol	Used for sending emails.

Encryption
Essential for sending data over a network securely.

Sequence	Putting instructions in a suitable order for your program to function properly.
Selection	A decision within a program which can either be Yes/No, True/ False. Allowing your program to take different paths. (IF/ELSE IF/ ELSE statements)
Iteration	Repeating something – using a count controlled or condition controlled loop
Variable	Data storage within a program that can change e.g. storing name as input
Constant	Data storage within a program that cannot change e.g. pi = 3.14
Casting	Setting a variable or data to a certain data type e.g. str(name) or int(age)

Python
0 1 2 3 4 5
-6 -5 -4 -3 -2 -1

Syntax Error	Logic Error
Program won't run Doesn't follow the rules of the programing language	Program will run Output won't be as expected

Writing functions

```
def keyword          name          parameter
def fahr_to_celsius(temp):
    return ((temp - 32) * (5/9))

    return                         return value
    statement
```

Data Types

String - holds alphanumeric data as text.

Integer - holds whole numbers.

Float - holds numbers with a decimal point.

Boolean - holds either '*True*' or '*False*'.

Programming

String manipulation – using name = "John Smith" as an example	
Uppercase	Turning the whole string to capitals e.g. name.upper()
Lowercase	Turning the whole string to lowercase e.g. name.lower()
index	Finding a particular position of a character e.g. name[5]
Sub-string	Getting a section of the string e.g. name[2:7]
Length	How many characters are in the string e.g. len(name)

Iteration

```
# 1d array
List1 = [4,7,2,6]
print (list1[1])

# 2d array
List2 = [6,8,3], [2,6,1]
print (list2[0][2])
```

1D array
| 1 | 2 | 3 |

2D array
axis 1 →
axis 0 →
| 1.5 | 2 | 3 |
| 4 | 5 | 6 |

```
File Object Created        Path to File              Mode
        ↓                        ↓                      ↓
f = open('c:\\temp\\data.txt",'r')
```

File Handling

w = write   r = read   a = append

The importance of testing

Requirements
Analysis & Design
Planning
Initial Planning
Implementation
Deployment
Evaluation
Testing

Iterative Testing

Final Testing

Example code for the 10 green bottles song

Uses a loop, variables, concatenation and arithmetic.
Code is easily maintained using indents and comments.

```
bottles = 10 # integer variable
text = " green bottles sitting on the wall" # string
text2 = "and if 1 green bottle, should accidently fall"
text3 = "they'll be "

#repeat the song 10 times
for i in range(10):
    # everything indented is inside the loop
    print(bottles, text) # concatenation
    print(bottles, text)
    print(text2)
    bottles = bottles - 1 # minus 1 from the variable
    print(text3, bottles, text)
```