



# Tiverton High School Year 8 Computing

## Autumn Term Knowledge Organiser

### Parts of a computer system | Key Construct 3: Computer Systems

A **computer** is an **electronic device** that follows a **stored program of instructions**.

The **program** of instructions tell it **how** to process data and how to make things happen e.g. activate outputs.

A **computer system** is a **collection** of parts that **work together** to perform a task – comprised of **hardware** and **software**.

**Input devices** transfer data **into** the computer processor.

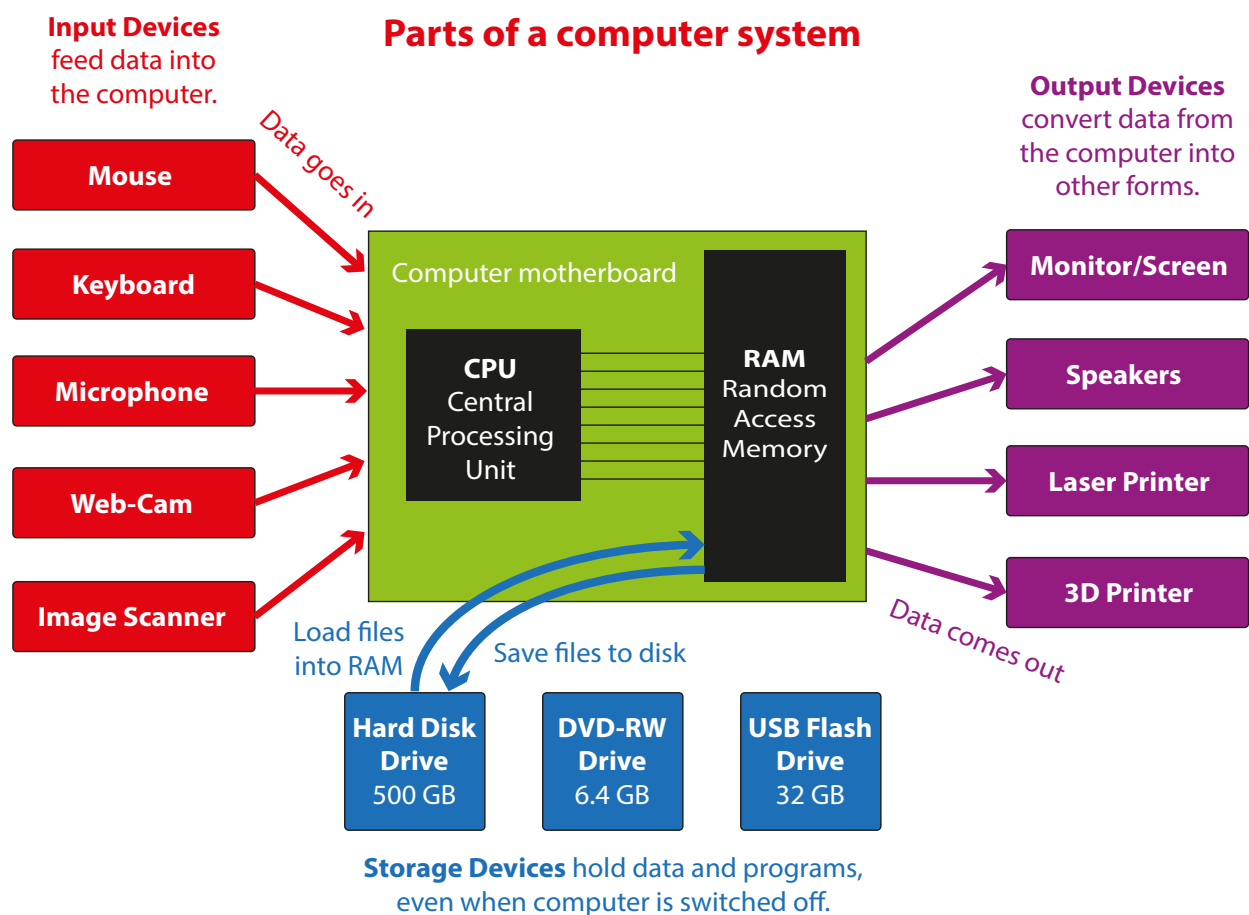
Examples: **QWERTY keyboard, mouse, microphone, web-cam, image scanner, accelerometer, fingerprint sensor.**

**Output devices** transfer data **out** of the computer for people to use.

Examples: **Screen/monitor, laser printer, audio speakers, 3D printer, robot-arm, LED display, laser cutter.**

**Storage devices** store data for **long term** or **while the computer is switched off**.

Examples: **Hard-disk drive, solid-state drive, optical drive, USB Flash-drive, magnetic tape drive.**



### Computers in control | Key Construct 3: Computer Systems

An **embedded system** is a dedicated single-purpose computer that is **built into some other electronic device**. The embedded computer **controls** the operation of that device.

Examples of embedded systems include **microwave ovens, burglar alarms, digital TV receiver boxes, GPS sat-nav systems.**

Embedded systems usually contain a cheap, simple **microcontroller** chip. They have less RAM and a simpler CPU than a PC.

## File-types and compression | Key Construct 5: Data Representation

A **file** is a **persistant** store of data that is held on a **secondary storage device** e.g. on a hard-disk drive.

Data and files can be **compressed** (to **reduce** the amount of data held) so they can be sent and received faster over networks.

**mp3, mp4, jpg** use **LOSSY compression** because **some of the original data is lost when compressing - it cannot be retrieved**.

A **character** is a **symbol** that can be stored inside the computer system using a special number called a **character code**.

The full collection of ALL of the characters that a computer can represent/store is called a **character set**.

**ASCII** is the **American Standard Code for Information Interchange**. It can be used for writing in the English language.

**Plain ASCII text** is often stored using **7 bits per character**.

A better version of ASCII is **Extended ASCII**. This can be used for writing in English, French, German, Spanish or Italian.

**Extended ASCII** contains more characters than original ASCII, but uses **8 bits** (1 byte) to store each different character code.

**Unicode** is a better character set. It can represent **any** language in the world, including Russian and Chinese, not just English.

**Unicode** can use up to **32 bits** for each character code. This means the text takes up more storage space in the computer.

**Emoji** pictures are character symbols from the Unicode character set. ASCII and Extended-ASCII do not contain any emojis.

## Units of data storage | Key Construct 5: Data Representation

**"bit"** means **"binary digit"**.

A bit is the smallest amount of data that a computer can store.

A bit can either be a **0** or a **1** value. It uses an **ON** or **OFF** voltage in a circuit.

**8-bit binary** means a pattern of **8 binary-digits**.

**8-bits** allow **256 possible combinations** between **00000000** and **11111111**.

This is why 8 bits can represent between **0** and **255** in base ten.

**1 byte** = **8 bits** (an ASCII character takes 1 byte)  
**1 kilobyte** = **1000 bytes**  
**1 megabyte** = **1000 kilobytes** (or 1000 x 1000 bytes)  
**1 gigabyte** = **1000 megabytes** (or 1000 x 1000 x 1000 bytes)  
**1 terabyte** = **1000 gigabytes** (or 1000 x 1000 x 1000 x 1000 bytes)

## Number bases

**Binary means base-2**

Computers use binary to store all data.

**Denary means base-10**

People usually use denary in everyday life.

**Converting Binary (base two) to Denary (base ten)**

128	64	32	16	8	4	2	1
0	1	0	0	1	0	1	0

$(1 \times 64) + (1 \times 8) + (1 \times 2) = 74$  in base ten

## Parts inside a computer | Key Construct 3: Computer Systems

**Hardware** means the **physical components, devices and circuitry** of the computer system.

A computer has a **processor** inside it.

Another name for it is the **Central Processing Unit (CPU)**.

The processor **executes** each instruction to carry out a program.

**Processor speed** is measured in **Hertz (Hz)**... cycles per second.

**1 Hz (Hertz)** = 1 clock cycle per second (very slow!)  
**1 MHz (Mega-Hertz)** = 1 **million** clock cycles per second.  
**1 GHz (Giga-Hertz)** = 1 **billion** clock cycles per second.

**RAM** stands for **Random Access Memory**.

RAM is a kind of memory storage inside the computer.

RAM is used to hold the program of instructions that the CPU needs to carry out. It also holds data that the program is using.

RAM is **volatile** - **all data is lost when the power is turned off**.

We "load" programs and data from disk into RAM, ready to use them.

## Software | Key Construct 3: Computer Systems

**Software** means the **programs** that it uses.

Without software, the hardware would be useless, it would not have any instructions to follow.

A **program** is a **set of instructions that tells the computer what to do**.

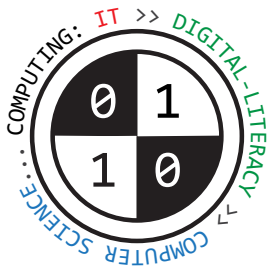
**Operating systems** and **utilities** are examples of **systems software**.

Without an operating system, a modern computer would be too difficult to use.

General purpose **applications** include

**Word Processors, Spreadsheets, Graphics Packages**.

**Software utilities** include **anti-virus** tools, **firewalls**, **file compression** utilities and **disk management** tools.



# Tiverton High School Year 8 Computing

## Spring Term Knowledge Organiser

### Computer networks and the internet | Key Construct 4: Networks and Communication

A **network** is a collection of computers (two or more) and other devices that are connected together.

These connections are usually **copper wire cables**, but they may also sometimes be **fibre-optic cables**, **wi-fi** radio links, or long distance **micro-wave** satellite links.

A **Local Area Network** usually **covers a single site or building in a small area**.

Computer devices can be connected together by plugging them into a **switch** using a copper **Ethernet cable**.

Many devices can communicate without wires, using **radio-waves (wi-fi)** if you add a **wi-fi access-point** to your network. Wi-fi access points send and receive **packets** of data through the air using radio signals.

The **internet** is a way of **connecting networks together**.

"**Internet**" means "**inter-networking**" – **communication BETWEEN networks**.

The internet joins Local Area Networks and Wide Area Networks from many different countries.

Most people access the Internet using a **broadband adaptor/router**.

You can then send and receive data, connecting to other computers using either **telephone lines** or **fibre-optic cables**. To communicate with computers on the Internet you connect to a server called an **Internet Service Provider (ISP)**.

Files and data are split into small chunks of data called **packets**.

Packets can travel across the Internet using different **routes** to get to their destination.

Every device on a network has a special **IP address (Internet Protocol address)**.

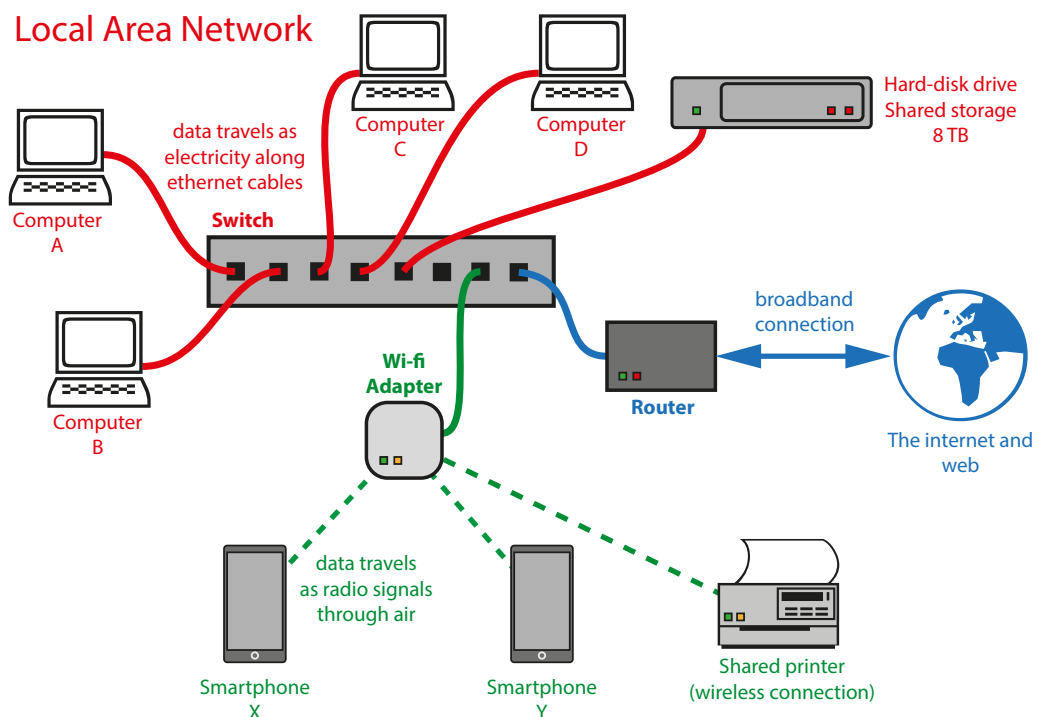
Every IP address needs to be **different**, so that devices don't get confused. It would be unclear where a packet needs to be sent.

An **IP address** is a **group of 4 numbers**, each number uses **8-bits** (between **0** and **255**).

Example: **168 . 0 . 192 . 1**

Example: **192 . 70 . 12 . 65**

### Local Area Network



## Problem solving and computational thinking | Key Construct 6: Problem Solving and Programming

**Decomposition** means **breaking a problem down into smaller parts**, which are easier to solve.

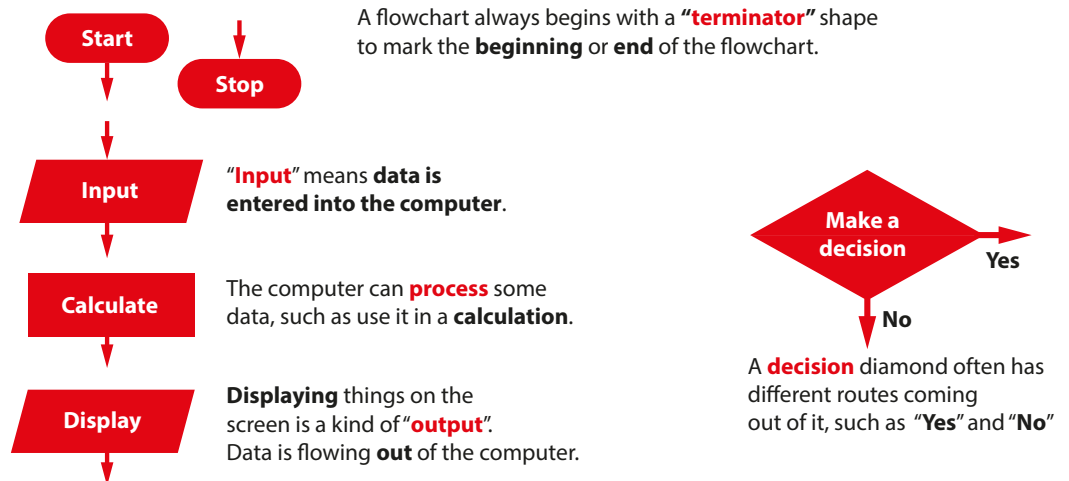
**Abstraction** means **choosing only the most important details that are relevant to solving the problem**, while **ignoring** other details.

A **program** is a sequence of instructions that the computer will carry out (execute).

An **algorithm** is a precise set of written steps that describe exactly how to solve a problem.

A **flowchart** is a diagram that shows how an algorithm works.

## Flowchart symbols | Key Construct 6: Problem Solving and Programming



## Writing pseudo-code | Key Construct 6: Problem Solving and Programming

You can **plan** out the steps of a new program using **pseudo-code**... "false" code. It is not a real **programming language**.

**Input** means gathering some data from the keyboard or other input device and storing it in a variable:

```
INPUT width
```

**Output** means displaying something on the screen:

```
PRINT "Your final score is"  
PRINT score
```

A **sequence** is a group of program statements that are executed in the correct order, one after the other.

A **variable** is a **named value** that can **change** while your program is running e.g. **score**

**Assignment** means giving a value to a variable

```
x = 3                                password = "arV$uSw"
```

**Iteration** means repeatedly executing parts of the program again and again (looping):

```
FOR time = 1 TO 10                    WHILE time < 60
```

**Selection** means making a decision to select which part of the program code should be executed:

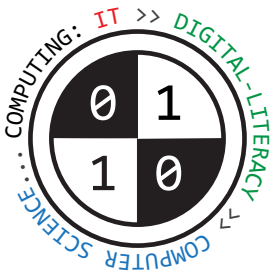
```
IF lives > 0 THEN  
    PRINT "Lost a life"  
ELSE  
    PRINT "Game Over"  
ENDIF
```

### Arithmetic operators

```
+      Addition  
-      Subtraction  
*      Multiplication  
/      Division
```

### Relational Operator Symbols when making comparisons

<	less than	>	greater than
<=	less than or equal to	>=	greater than or equal to
==	is the same as	!=	not the same as



# Tiverton High School Year 8 Computing Summer Term Knowledge Organiser

## Developing computer programs using Python | Key Construct 6: Problem Solving and Programming

Python is a **high-level programming language**. It can be used by beginners to create computer programs.

Many people use **IDLE** to create Python code. This is an **Integrated Development Environment**. It contains a **text-editor** for writing Python code and other tools that are helpful to programmers.

When you **save** a new program, the filename needs to end in **.py** so the computer knows it can be executed using Python.

If you make a mistake or type an error in your program, the code may not make sense when Python tries to execute it. Python will stop running your program and try to show you **where** the error is in your code so you can fix it. This is called a **syntax error**.

A **comment** is a line of text in your program code that the computer **will not execute**. It will be used by the computer when running a Python program. It is used as a **reminder** or as an **explanation** to someone about how your code works.

**To make text into a comment**, type in the **#** symbol at the start of the line of text e.g. **# Main menu starts here**

Your programs can work with different **kinds** of data values. We call these **data-types**.

<b>Integer</b>	a <b>whole number</b>	e.g. <b>32, -7, 0</b>
<b>Real</b>	a number that can contain a <b>decimal point</b> , these are called "float" values in Python	e.g. <b>3.14</b>
<b>Boolean</b>	a value that can only hold one of two possible states, either <b>True</b> or <b>False</b>	
<b>Character</b>	a <b>SINGLE character symbol</b> .	e.g. one <b>letter</b> , one <b>digit</b> , a <b>punctuation mark</b> . <b>'?'</b>
<b>String</b>	a <b>sequence of characters</b> .	e.g. <b>'This cheese smells.'</b>

Most Python programs use **variables**. A variable is a named value that can be **changed** during the execution of a program. When we set the value for a variable, we call this **assignment** (we are assigning a value).

Use the **= assignment operator** to **set a value** e.g. **password = '\$ecRet123'**

A **print** statement **displays a line of text on the screen**. This is an **output**.

Text enclosed by **' '** quotes will literally be displayed. Anything without quotes will display the value of a variable.

```
print( 'Hello' )  
print( password )
```

An **input** statement allows people to type useful data **into** a program once it is running.

Whenever you type a data value into the computer, the value must be **stored** in a **variable** (so the computer does not lose it).

```
surname = input( 'Type in your surname... ' )
```

A program can make **decisions** while it is running to **choose** what should happen next. This is called **selection**.

It can **decide** whether or not something should happen, depending on whether a **condition** is found to be **True** or **False**.

```
if guesses <= 5 :  
    print( 'Try again.' )  
else :  
    print( 'No guesses left.' )
```

When making **comparisons** to **select** what should happen, you can also use these **operators** to **compare values**:

<b>==</b>	means <b>equal to</b> , "the same as"
<b>!=</b>	means <b>not equal to</b> , "not the same as"

**Iteration** means to carry out instructions **more than once**.

You can carry them out a certain, definite number of times using a **for** loop.

```
for loop in range( 1, 6 ) :
```

You can also carry out a sequence of instructions only while a certain condition holds **True**, using a **while** loop.

```
while messageToRead == True :
```