Computing: Last Minute Revision

* **Hardware** is the **electrical** components/circuits a computer is assembled from; the platform upon which the software is executed.
* **Software** is **sequences of instructions** called programs which are understood and executed by the processor.
* **Moore’s Law**: The number of transistors than can be integrated into a chip for the minimum possible costs doubles every 18-24 months.
* The Central Processing Unit (**CPU**) is used to describe the processor AND main memory ONLY.
* A peripheral is a computer device **NOT PART OF THE CPU** (i.e. anything **but** the processor and main memory); can be **INTERNAL** (e.g. CD-Drive) or **EXTERNAL** (e.g. Keyboard).
* **I/O devices (peripherals)** send/receive/store data by communicating with the CPU **via an I/O Controller**.
* An I/O controller consists of:
- **Electronics** that interface the **controller to the system bus**;
- **Electronics** appropriate for **sending control signals** to the **peripheral**;
- A set of data, command and status **registers**.
* The processor communicates with the I/O controller through a port in the controller called the **I/O port**.
* **Main Memory** (or Immediate Access Store IAS) is memory which is **directly addressable by the processor** (i.e. Does not require an I/O controller to communicate with it, e.g. RAM and ROM).
Memory not directly addressable by the processor (i.e. requires an I/O controller for communication and so is a **PERIPHERAL**) is called **Secondary Storage** (e.g. DVD-R, Flash Memory etc).
* The **System Bus**/External Bus (consists of Data, Address and Control Buses) connects the processer, main memory and I/O controllers together.
* The address bus is the ONLY UNIDIRECTIONAL BUS, the rest are bidirectional
* Examples of control signals carried by the control bus:
- **CLOCK SIGNAL** for timing purposes;
- **RESET SIGNAL** to initialise components;
- **MEMORY READ/WRITE** to assert whether the memory location being accessed is being read or written to;
- **I/O SIGNALS** to indicate the processor wants to use an I/O controller and **NOT** a main memory, when they both may have the same memory addresses.
* When the processor, main memory **and I/O controllers** are integrated on a **single chip**, it’s called a **Microcontroller**.
* **Components** of a **processor**:
- **PROGRAM CONTROL UNIT**, this fetches instructions from memory, decodes and executes them one at a time (i.e. It’s responsible for carrying out the **fetch-execute cycle**);
- **ALU** performs the **arithmetic and logical operations on data**;
- **Registers** are **fast** memory locations **inside the processor** (or I/O controllers) which can be dedicated or general purpose;
- **Internal clock**: derived either directly or indirectly from the system clock;
- **Internal buses** link the PCU, ALU and the registers together;
- **Logic gates** used for **flow control**.
* **General-Purpose** registers are **not assigned a specific role** by the processor designer; programmers may use these.
* **Dedicated registers** are **assigned a specific** **role** by the processor designer; programmers may use some but not all of these. The dedicated processers you need to know are:
**STACK POINTER**: Points to the stack holding the **necessary data when a function is called** (e.g. local variables, function parameters, return addresses etc);
**PROGRAM COUNTER**: Points to the **next instruction to be fetched and executed**;
**STATUS REGISTER**: Holds the **condition codes** to **indicate the outcome of operations** (e.g. whether the result of a calculation is positive or negative);
**ACCUMULATOR**: holds **results of current set of calculations**;
**CIR**: Holds the current instruction to be executed whilst it is decoded and executed;
**MAR**: Holds the address of the memory location currently being accessed by the processor;
**MBR**: Holds the data item currently being transferred to or from the addressed memory location.
* A **high-level language program** which has **not yet been compiled** is known as the **source code.**
* A **compiled** **high-level language program** (and so is in machine code) is known as the **object code**.
* An **IP Address** is a **unique address** that **identifies a computer/electronic device on a network**.
* The **World Wide Web** is a system of **interlinked hypertext documents** **accessed via the Internet**.
* **Internet registries** **track associations** between **Internet addresses and domain names** (e.g. 196.100.11.4 and Microsoft.com)
* **Internet registrars** are responsible for **registering internet domains to people and organisations**.
* **Client port numbers** are **temporary port numbers** assigned by the **Transport Layer,** which range from 1024 to 4096, used by **client** processes
* **Well-known port numbers** are **universally known** port numbers used by servers (so the client knows which port to access) which range from 0 to 1023. Some examples you need to know:
80 or 8080 - Web Server
25 – SMTP Server
110 – POP3 Server
23 – Telnet Server
20 or 21 – FTP
**SEE PAGE 197 of the text book** on explanations of how each of these work…
* **Hyper-Text Markup Language (HTML)** is used to write web pages. Consists of text that defines the **content** of the page, and tags that define the **structure** of the page.
* A **web page** is a **document on the WWW** written in **HTML** and displayed in a web browser.
* A **hyperlink** links one web page to another.
* **Hypertext** is a body of text and graphics stored in a **machine readable form** and structured so that a reader can cross refer between related items of data.
* A **web browser** displays a web page by **rendering its HTML elements**.
* A **website** is a set of **linked docs** on a computer system and can be **accessed on the WWW**

BASIC HTML SKELETON:

<html>
 <head>
 <title> Web page title goes here! </title>
 </head>
 <body>
 This is where the page content goes!
 </body>
</html>

<br> is a line break;
<p> and </p> insert an extra line after a paragraph;
<h1> and </h1> makes text larger and bolder. Can use numbers from 1 to 6 with 1 being largest text size.
<meta> tags in the <head> section are used by search engines and other programs to **categorise a web page**, eg:

<head>
 <title> Computing </title>
 <meta name = ”author” content = “Fred Bloggs”> </meta>
</head>

In the above example, ‘name’ and ‘content’ are **attributes** of the <meta> tag.

**// Example of an image in HTML. Remember when assigning values to attributes, use // double speech marks. Also, for the <img> tag, there is no </img>.**

<img src = “image1.gif” alt = “first image” width = “70” height = “70”/>

**// Example of a hyperlink:**

<a href = <http://www.microsoft.com/>> Click here for Microsoft! </a>

* **Colour schemes** used in **web page design**:
**MONOCHROMATIC**: **Various tints** of one base colour;
**ANALOGOUS**: **Three colours side** by side in a 12 part wheel;
**COMPLEMENTARY**: Colours **opposite each other on a colour wheel**.
* **Cascading Style Sheets (CSS)** allows to change the **STYLE** of web pages in one go.
* **Type Selectors** select HTML elements to follow a style rule. Their general format is:
selector { property : value }
Where selector is the HTML tag to be styled, and property is the attribute of the tag to which the value is to be assigned.
Examples of type selectors:

h1 { color : red } // Makes all h1 text red.

p { font-family : ”sans serif” }
// When the value consists of two or more words, use “”

p { text-align : center; color : blue }
// To assign values to more than one attribute, use a semi-colon for separation.

h1, h2, h3 { color : red }
// To apply the same styling to several tags, use commas to separate the selectors.

An **embedded style sheet** is placed within the head section of the HTML page as follows:

<head>
 <style type = “text/css”>
 p { color : red }
 </style>
</head>

An **external style sheet** is much better, it is created separately using the ‘.css’ extension and then linked to each web page to allow consistent styling. To link an external CSS to your web page:

<head>
 <link rel = “stylesheet” type = “text/css” href = “MyStyles.css”/>
</head>

// The ‘rel’ attribute shows **relationship** is of type style sheet
// The ‘type’ attribute shows the style sheet has **CSS formatted text**
// The ‘href’ attribute is a reference to the **MyStyles.css file**

A **class selector** is used to **select any HTML element** that has a **class attribute**. For example:

<h1 class = boldRed> Hello World! </h1>
<p class = boldRed> Hello World! </p>

.boldRed { color : red; font-weight : bold }
This will declare **all elements of class boldRed** to have a colour of red, and styled in bold font!

p.boldRed { color : red; font-weight : bold }
This will only declare **the <p> element of class boldRed** to have a red colour in bold.

* <em> </em> is used to **italicise** text
* <strong> </strong> is used to **bolden** text

***COPYRIGHT, DESIGNS AND PATENTS ACT 1998***

* Protects **intellectual property**;
* **Copyrighted work** can only be used/copied with **owner’s permission** (**Exception for education purposes**);
* **Design right** gives you protection from someone **copying your original design**;
* A **patent** protects **how an invention works or what it does**. It **cannot be obvious** to others with knowledge in the field. An **exception** this is **mathematical or scientific discoveries**;
* A **trademark** protects the **name or logo** used to **identify** a **business or product**.

***COMPUTER MISUSE ACT 1990***

* Protection from:
- **Unauthorised access** to **computer material**;
- **Unauthorised access** with **intent to commit or facilitate commission of further offences**;
- **Unauthorised** **modification of computer material**.

**Cracking/Hacking** is **illegally breaking into a computer system**.

***DATA PROTECTION ACT 1984 and 1998***

* **Personal data** is data that can **identify a single living individual**
* A **data controller** determines **why and how personal data is processed**
* A **data processor** **process data on behalf of the data controller**.

 **EIGHT PRINCIPLES:**

* Data **Gathering**: Subject **is given consent** of **what and why** the data is **being used for**;
* Data **Purpose**: Data is **obtained for the specified lawful purposes**;
* Data **Quantity**: Data is **adequate, relevant and not excessive** for purpose it’s being processed;
* Data **Quality**: Data is **accurate and up to date**;
* Data **Subject’s Rights**: Subject is told **what data/what for/to whom their data is being disclosed to**. Subject has right to have **inaccurate data destroyed/corrected**, correct **compensation for damage/distress** resulting from contravention of this principle;
* **Internal Data Security**: Protection against **unauthorised access and against accidental damage**;
* **External Data Security**: Data shall **not be transferred outside the EU** unless that region ensures **adequate rights and protection.**
* Every data controller must register with the **Information Commisioner**
* **Exemptions** are **national security, taxation or domestic purposes** (e.g. personal address book)

***HEALTH AND SAFETY (DISPLAY SCREEN EQUIPMENT) REGULATIONS 1992***

* Horizontal forearms and eyes the same height as the top of the VDU;
* Arrange desk to avoid glare, bright reflections on screen.

***REGULATION OF INVESTIGATORY POWERS ACT 2000***

* Under this act, it is an offence to intercept a message sent via a public or private telecoms system
* But security and intelligence services are exempt from this act.

***ETHICS***

* A **code of conduct** is **not a law**. If you break the code of conduct you **cannot be criminally convicted** but the organisation may follow a disciplinary procedure.
* **Ten commandments of computer ethics**, two examples:
- Don’t use a computer to harm other people;
- Don’t use a computer to steal.
* **Digital Rights Management (DRM)** is the application of **control technologies** to **limit the use of digital data**
* **Proprietary Software** is when the owner places **restrictions on its use**, copying and modification. Sold by software licences which only allow use of the software.
* **Freedoms of free software licences** (two examples):- Run program for any use;
- Study how program works and adapt it to your needs.
* A **robot** is a **moveable mechanical structure** that can sense its surroundings and **manipulate things**, **controlled by a computer program**.
* The **Digital Divide** is the gap between people with **IT access** and those without.