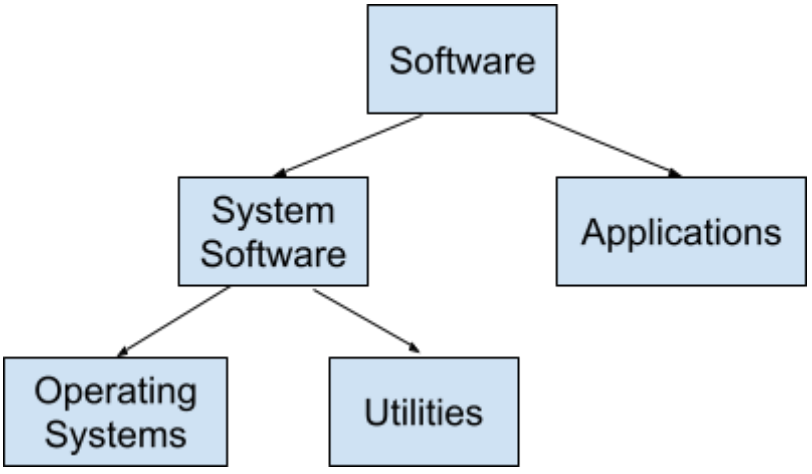
Operating Systems

We can divide computers between hardware and software, but can also go further into different categories, as follows:



Question 1 - What are two differences between a Utility program and an Application program?

Question 2 - Name three different operating systems.

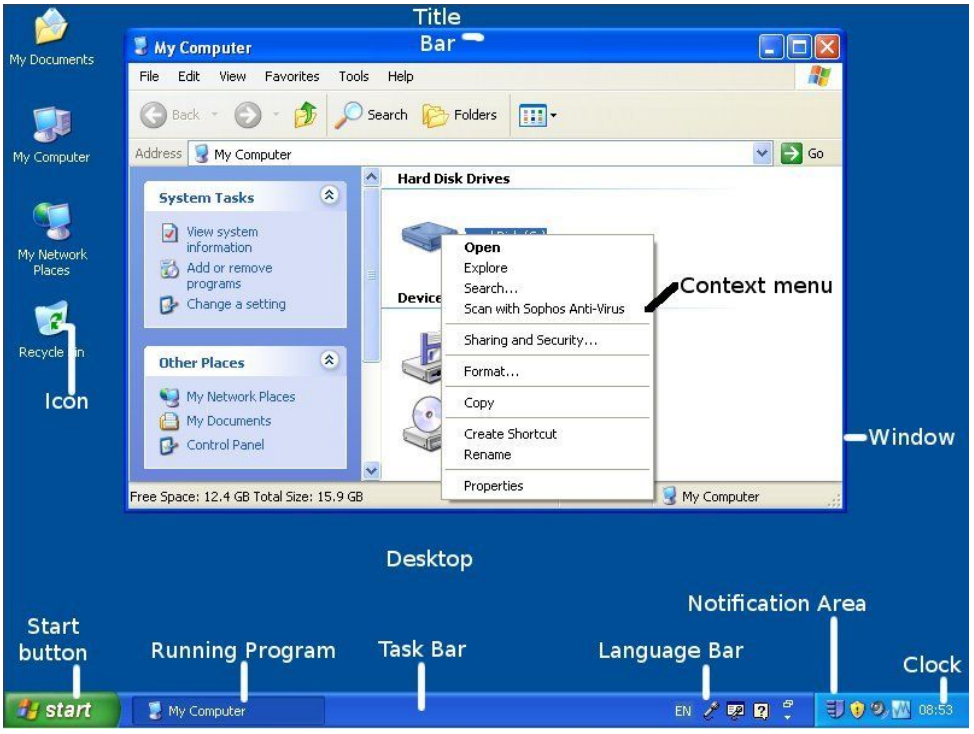
Question 3 - Name three types of utility program.

The Operating System is an essential part of any computer, as it is a piece of software that performs key functions. These include:

Providing a user interface

The operating system creates an interface that allows the user to control the hardware. The most common sort of interface is a **GUI** - a Graphical User Interface. A GUI usually makes use of features known as **WIMP** - Windows, Icons, drop down Menus and a Pointer.

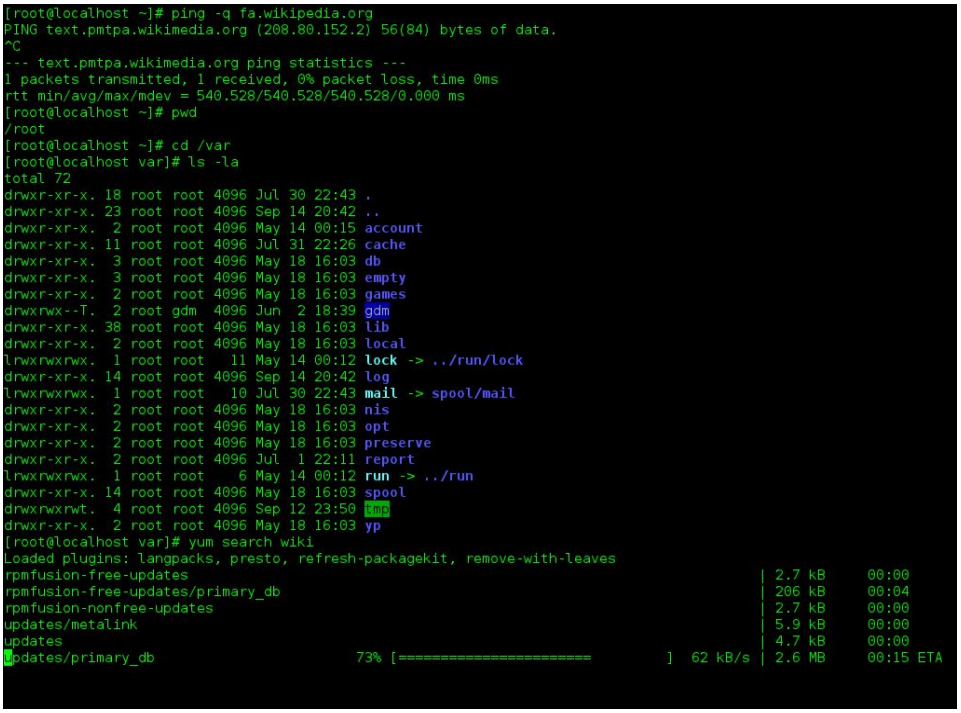
This is a typical WIMP interface:



Some modern GUIs don't use all of these features - e.g. the touch screen interface on your phone.

Older, or more specialist, software may only use a **Command Line** interface - this is an interface where you type in text, command words, which cause the computer to perform actions.

A typical Command Line interface might look like this:



Question 3 - What advantages does a GUI interface offer a user?

Question 4 - What advantage does a Command Line interface offer a user?

Question 5 - Other than a GUI or Command Line interface, what other types of interfaces are used to control digital devices?

User management

Managing users is a particular issue on Network computers - Network Operating Systems provide a range of features to manage users, including setting up usernames and passwords, providing network storage for files, implementing **permissions** so that not all

users can do the same things on their computers (i.e. a user may be ‘locked out’ of certain software, or able to access some files but not others).

Permissions may be set for a given file, a given location on a computer, or a given function. For files permissions are typically set to ‘Read, Write, Delete or Execute’. i.e. a low level user might be able to read a file, but not edit it, a manager might be able to read and change (write to) the file.

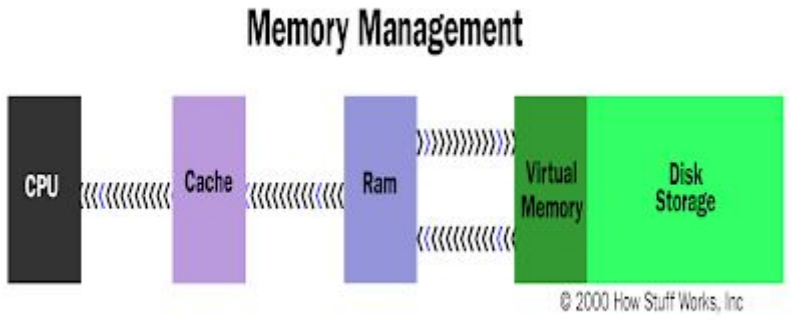
Question 6 - How can the use of usernames and passwords keep a system secure?

Question 7 - Give an example of how permissions can be used to keep data on a system more secure.

Memory management

When you open a program or file it is copied into memory. The operating system is responsible for finding space for the program or file in memory, and allocating it to the right location, as well as stopping programs accidentally overwriting each other. If this were done accurately data might be lost, or programs would not successfully launch because they would not fit into memory.

Memory management is a very important function of an operating system. It is essential that data is not lost. Sometimes memory becomes very full, but it would be bad if, when you tried to open another program, the computer couldn’t do so. To overcome this, a computer will make use of **Virtual Memory**. This means copying some of the contents of RAM temporarily onto the secondary storage (usually a hard drive) to free some space in RAM. This data will be copied back into RAM when needed again.



Question 8 - What would happen if a system did not use virtual memory, but it ran out of RAM?

Question 9 - When a computer begins to use Virtual Memory it will begin to run more slowly. Why is this?

File management

It is the Operating System that is responsible for saving files onto the hard drive. It has to organise the files, make sure none get lost, and also place them in folders in an organised way. The way files are organised on a hard drive has very little to do with the way files are presented to the user, so the Operating System also has to 'remember' how a user wants to see their files.

Operating Systems also allow for **File Permissions** - this means that different users on the same computer may or may not be able to access a particular file. A user might not be able to open a file at all, might have 'read only' access (meaning they could look at its contents but not change them), might have 'read/write' access (meaning they can read it, edit it, but not delete it), or 'full access' (meaning they can do anything to the file).

Question 10 - if an Operating System did a bad job of file management, what kinds of problems would a user run into?

Peripheral device management

The Operating System allows the computer and user to make use of devices like printers, mouse, keyboard, etc. The Operating System contains code that allows the computer to run these devices, take input from them, given output to them, etc. Without this code, the computer wouldn't know the devices existed, let alone be able to do anything with them.

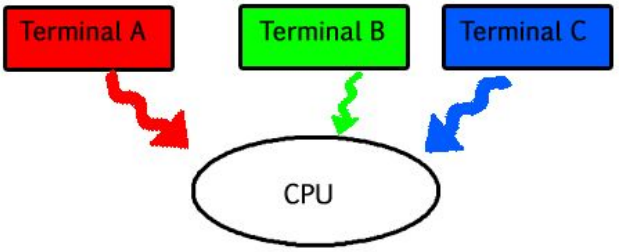
Peripherals:

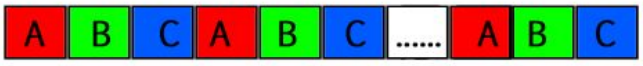
Because there are so many devices, and because new ones are being created all the time, an Operating System can't know how to make use of all devices that exist. Therefore, devices come with **Device Drivers** - small, specialist programs, that act as a translator between the Operating System and the hardware, allowing the Operating System to use a brand new piece of hardware.

Question 11 - if we didn't use device drivers, the operating system would have to be programmed to deal with all hardware. How would this change the way that manufacturers were able to create new peripherals?

Multi-tasking

Almost all modern devices are Multi-tasking - this means they can run more than one program at once. However, in actual fact the processor can only 'do' one thing at once (unless it is multi-core), so in order to allow more than one program to run at the same time the Operating System very rapidly changes from one process to another, give a split second to one program then a split second to another, back and forth. This gives the illusion to the user that programs are running at the same time. This is called Multi-tasking.





Multi-core processors allow for 'true' multi-tasking, in that each core can run a separate program simultaneously, without interfering with each other. However, on this course questions will be about the multi-tasking that involves swapping between different programs.

Question 10 - completing tasks in a multitasking way is never quite as quick as completing them in sequence. Why not? Where is time lost?

Question 11 - describe a common example of multitasking on a desktop computer: when might you have multiple programs running in the processor at the same time.