Single user operating systems

Single user operating systems can be split into two types:

- single user, single application operating systems
- single user, multi tasking operating systems

Single user, single application

This type of operating system only has to deal with one person at a time, running one user application at a time.

An example of a this kind of operating system would be found on a mobile phone. There can only be one user using the mobile and that person is only using one of its applications at a time.

Single user, multi-tasking

You will find this kind of operating system on a personal computer.

The operating system is designed mainly with a single user in mind, but it can deal with many applications running at the same time. For example, you might be writing an essay, while searching the internet, downloading a video file and also listening to a piece of music.

Example operating systems are

- Windows
- Linux
- Mac OS X

The difference compared to the Single-Use, Single Application operating system is that it must now handle many different applications all running at the same time.

The memory available is also very different, for example it is quite normal to have Gigabytes of RAM available on a personal computer which is what allows so many applications to run.

What is an operating system?

An operating system is the core software that allows a computer to run as an useful device. It manages the hardware, the user interface and all other software running on the computer.

Without an operating system, a computer is just a collection of components heating up the room as no-one would be able to make any practical use of the machine.

A large number of operating systems have been developed over the years; each designed with a certain kind of computer in mind. For instance:

- Windows or Linux - for personal computers
- MacOS - for Macs
- Unix - for mainframes
- Symbian, Android - for mobile phones

Although these operating systems have been developed to work with very different types of computers, they all share a number of common functions:

- Providing a user interface
- Managing the computer's memory
- Managing the hardware

Multi-user, multi-tasking operating system

On the previous page we talked about a single user working on their personal computer doing many different tasks at the same time - effectively, a single user operating system which allows you to multi-task.

Personal computers can multi-task very well, especially for the type of things that most of us want to do, for example, reading emails, writing letters, working on spreadsheets, listening to music, surfing the web and watching videos.

courtesy Argonne National Laboratory

However, there comes a time when only a *really* powerful computer will do the job in hand. For instance

- You are an engineer or scientist and want to run a very complicated simulation
- You are a weather scientist and want run a forecast
- You are a financial person and want to work on thousands of stock market share movements
- You work in a bank and want to handle customer accounts.
- You are an architect and want to see your full design
- You work at an University as an academic along with hundreds of other academics
- You are a film animator and want to work in 3D

Most personal computers can't handle these kind of tasks. Instead, a mainframe or supercomputer is required for this kind of work.

But there is a problem!

A supercomputer or mainframe costs millions to buy and maintain. There is no way that such an expensive machine could be used by just a single person.

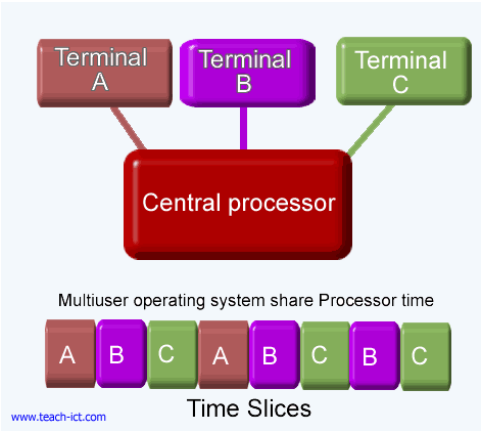
To make it economic, this computer has to be shared. This means it needs a multi-user operating system which means more than one user is logged on and can use the computer at the same time.

And furthermore, each person needs to be able to run more than one application at a time, so it needs to be multi-tasking as well.

So a powerful computer needs a **multi-user, multi-tasking** operating system to make maximum use of the machine. Each person can draw on the vast power of the computer in a shared way.

Now the operating system has to manage

- Each user logged on to the system, their workspace and so on.
- Allocate resources to the jobs they want to run.
- Keep logs of how much processing time and resources they use
- Work out the most efficient use of computer processing cycles
- Maintain security



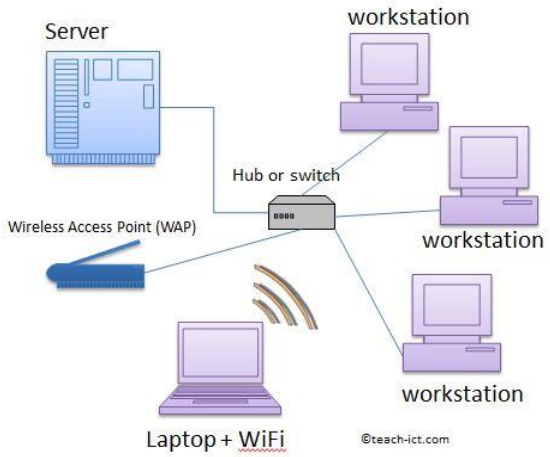
The diagram below shows how each person on terminals A,B and C share the power of the computer.

Extra fact.

We have mentioned 'powerful computer', so how powerful is this compared to your home computer? Well, one measure of computer power is FLOPS, which is how many calculation it can do per second. A typical home pc is about 30 GigaFlops but a supercomputer is about a 1 million GigaFlops!

Network operating systems

These days it is quite common to have your computer attached to a network of some kind.



For example most of the computers in school will be networked. Most of the computers in companies will also be networked and even at home you may have two or more computers networked together so they can share the internet connection.

All these machines need to have an operating system that can deal with the network.

Examples of a network operating system include Windows and Mac OS X and of course all the mainframe operating systems.

A network operating system has to have the following features

- Deal with users logging on.
- Maintain the network connection to the server(s)
- Expand the file system to view folders on other computers
- Provide security to separate user accounts from each other.

When the user logs on, they will provide an username and password. This is checked by the operating system.

With a network operating system, you will have access to a network drive where your files and folders will be stored. You will usually also have access to a shared drive where you can use and share files with other people on the network. These drives appear as extra letters, perhaps W drive or S drive. As far as you are concerned, as a network user, they are appear to be an extension of your local hard disk. You can read and write to the folders as if they are on your own machine (as long as the network administrator has set up your file permissions to allow you to do this.)

There may be many users making use of the same machine (not at the same time). Think about the computers at school. You might use a computer in room A for your ICT lesson. You log off, and the next class arrives and a different student logs onto the same computer. This happens many times throughout the week. This is *not* a multi-user system, because the processing power of the machine is only being used by one person at a time.

Real time systems

Real time processing is usually found in systems which use computer control.
This processing method is used when it is essential that the input request is dealt with quickly enough so as to be able to control an output properly. For example, the computer inside the Engine Control Unit in a car has to manage the engine at every moment based on what the driver wants to do.

Real time processing has to be programmed very carefully to ensure that no input events are missed.
Note that real-time processing does not have to be 'fast'. For example, a traffic light system is a real-time system but it only needs to process data relatively slowly. On the other hand, controlling a car engine has to deal with input events happening every thousandth of a second so a very fast computer is needed to do this -but both the traffic-light and the car engine computers are carrying out 'real-time' processing.
Examples:

- Traffic lights
- Heart rate monitoring
- Aircraft control
- Computer games
- Controlling robots

NOTE: Some books will tell you that real time systems include ticket booking systems, flight booking, theatre seat booking systems etc. That is not correct - real time systems are time critical and instructions will take priority over anything else. Yes, booking systems need to be able to process data very quickly, but they are not as time critical as real-time instructions. However, exam questions will probably expect you to include booking systems within the category of real-time systems.

Batch processing

It is often not desirable to deal with the inputs until a certain number have occurred or a set time has passed. So they are stored until the system comes online to process the data in one 'batch'.

Batch processing is usually fully automatic unlike 'real-time' or transaction processing which are interactive.

For example

- A stock control programme may store records of every item sold in a shop that day. Then, at the end of each day it calculates what needs to be ordered.
- An online competition stores all the entries until it is time to find the winner.
- Electricity, gas and telephone bills are usually calculated on a monthly basis.
- Producing monthly bank statements to send out to customers