Chapter 1.2 System Software

1.2 (a) Operating Systems

An operating system is a set of programs designed to run in the background on a computer system, giving an environment in which application software can be executed. Most operating systems comprise a large set of programs, only some of which are stored in the processor memory all of the time. Many of the routines available in the O.S. are stored on the hard drive so that they can be accessed when required. This not only saves space in the processor memory but also means that the O.S. can be easily changed to a different one.

When you are using an applications package you are not communicating with the computer hardware, you are communicating with the operating system. Without an operating system, no matter how many programs you have, the computer is useless. The operating system sits between the hardware and the application program or user.

An operating system is likely to be stored on a backing store rather than in the memory of the computer because:

1. Usually the operating system requires a large amount of storage space.
2. It allows for easy upgrading or changing from one system to a different one.

Summary:

1. Operating system is a software program which controls the operations of the computer system.
2. It provides a user interface.
3. It controls how the computer responds to user’s requests
4. It controls how the hardware communicate with each other
5. It provides an environment in which application software can be executed
1.2 (b) Types of Operating System

Batch Processing:
When computing was still a new science, there were not enough machines to satisfy the demand for processor time from students in universities who wanted great calculations done, firms who wanted their payroll worked out, and many others. The big problem was the ‘speed mismatch’ between the user sitting at the keyboard who was very slow, and the machine which was very fast. This meant that the expensive part, the computer, was sitting there doing nothing while the human being decided what to do. There are two simple solutions to this problem, one is to buy more machines and the other is to make the machines work more effectively by taking away the slowest part of the system – the human being. Nowadays we might well opt to buy more machines, but this used not to be an option. This problem gave rise to the development of batch processing.

A batch processing operating system is one that does not allow for interaction between the user and the processor during the execution of the work. Lots of programs or data that need to be run are collected together (to form a batch) and they are sent to the computer. The batch operating system then controls their passage through the computer.

Nowadays, batch processing tends to be used where:
- when processing cannot be started until all the data is collected
- there are large amounts of data to be processed,
- the data is very similar in nature and…
- it requires similar processing,
- the computer system has identifiable times when it is not being used, and so has available processor time
- the application does not require human intervention.

Typical examples of applications which would be done using batch processing include production of bank statements from customer files, production of gas (electricity, telephone) bills from customer records, the compilation of high-level language programs where a number of users want programs compiled.

Real-time:
A real-time O.S. is one which can react quickly enough to affect the next input, or process to be carried out.
Most real-time systems are based on control of some process in the real world or on information handling. Real time process control is a real time system. But this is very different to real time (transaction) processing; this system usually involves sensors and feedback loops i.e. the output can influence the next input to the system.
Sensors send data (via an analogue to digital converter – ADC) to a computer or microprocessor which decides whether or not to take action (based on comparison with stored data). If a change needs to be made, the computer/microprocessor sends signals (via a digital to analogue converter – DAC) to the devices being controlled and alters their status (e.g. turn a heater on if a temperature sensor indicates that a temperature is too low) – actuators are often used to open valves, turn on equipment, etc. As this is continuously monitored, eventually the temperature will equal the stored value in the computer/microprocessor – this will then cause the heater to be turned off. A chemical plant has a reaction vessel where the temperature is critical to the result of the process.
The temperature is monitored by a computer which accepts input from a sensor and then makes a decision whether to adjust the heating elements in the vessel. In this example, it would not be sensible for the computer to be running any O.S. that is not real-time because if there was a delay in the decision making process, it might mean that the reaction is corrupted in some way. A robot trolley is controlled by a processor which takes input from a sensor following a black line on the floor, and makes decisions concerning steering to keep the trolley on the black line. If the processor was not controlled by a real-time O.S., the trolley would very soon leave the black line because it would not be steering quickly enough.

In real time (transaction) processing files are often updated in real time (e.g. when booking flights on an airplane); but in real time process control, physical quantities (such as temperature) are continuously monitored and the input is processed sufficiently quickly to be capable of influencing the data source. A catalogue shop processes orders by the code for a product being input and the system then comparing the code with information in its files. When it finds the correct code it can report to the user the number of that item that there are in the store. If there was only one left of a certain item, it would be necessary to record the fact that a shopper had bought it before the next shopper has their request dealt with otherwise the second person might be sold the same item. Because the information on the system must be processed immediately the O.S. needs to be ready to handle input as soon as it comes in. This means that it cannot be using up some of its slack time doing other tasks while it is waiting to be asked to do something. This implies that the computer will not be using its full potential as far as processing is concerned. When this happens it is said to display a high rate of redundancy. Real-time systems tend to display a high rate of redundancy.

Single User:
As the term implies, a single user O.S. is specifically one that is used to control a system which has only one user, and their programs, at any one time. A perfect example of a single user system is the one that you may have at home. Only one person uses it at a time. Most single user systems are also multi-tasking.

Multi-tasking:
This is a type of O.S. that allows several applications to be available simultaneously. On a simple single-user system you will probably be used to having a number of things running at the same time. Perhaps one window shows a spreadsheet while another shows a word processing application. You may decide to copy the sheet from the spreadsheet software to the word-processed document. It appears that more than one task is running simultaneously. They aren’t, they just appear to be. The O.S. that most of us use on our own computer systems, Windows, is a multi-tasking O.S.
Application programs now days also run many different tasks in multi-tasking operating system environment. So it is not necessary that only end users multi-task but programmers can also program applications that require multitasking.

Multi-user:
Again, as the name implies, this type of O.S. services more than one user simultaneously. There are two types of multi-user O.S.:

1. A network system comprises a number of computers linked together for the purposes of communication and sharing of resources. Normally one of the machines is used to control the rest of the system; this machine is called the server. Networks are important because they allow hardware and software to be shared and also mean that a single copy of the information on a system is needed, and so can be kept up to date relatively easily.

2. A time-sharing system has a single (normally powerful) computer which is connected up to a number of terminals. These terminals are not computers as in the case of the network system, but have a very limited amount of processing power. Again, such a system allows communication between users on the system and also allows sharing of hardware and software across the system.

At the moment it is difficult to tell the difference between the two types of multi-user system, but be careful because a network system is not really a multi user system as each computer is only being used by one person at a time. We shall return to this in chapter 3.1.
Distributed:
A distributed system is one that allows software and data files to be distributed around a system. An ordinary network will have a server controlling it and the access to the hard drive, which is connected to the server. A distributed system might store the word processing software on one computer’s hard drive, while the files of work are stored somewhere else, and the spreadsheet software is stored on a third disk drive. This can speed access to files because there is no single bottle neck which all the information must pass through; however the security and maintenance of the system are more complicated. Another type of distributed system is to distribute the entire programs and data to more than one place on the network. This speeds up access to the data but means that if data is updated on one part of the system it may not be on others, which means that the different versions of the data need to be compared and amended on a regular basis.
1.2 (c) applications requiring batch and online processing

Applications requiring batch processing
Utility billing system: The consumption units’ data of the utilities is gathered for a particular time and then processed altogether and bills are generated.

Payroll: The attendance, loans, taxes and other salary related data is collected for the whole month (or for predefined period) and payroll is run at the end to distribute salaries.

Backing up of day long transactions at the end of the day: For example at bank all of the transactions happened during the day are backed up at the night.

Applications requiring online processing
Health monitoring systems: At the hospital emergency machines monitor the health condition in real time and let the attendant, doctor or nurse know about any situation that may arise due to the bad condition.

Process Control: Process control is a statistics and engineering discipline that deals with architectures, mechanisms, and algorithms for controlling the output of a specific process. See also control theory.

For example, heating up the temperature in a room is a process that has the specific, desired outcome to reach and maintain a defined temperature (e.g. 20°C), kept constant over time. Here, the temperature is the controlled variable. At the same time, it is the input variable since it is measured by a thermometer and used to decide whether to heat or not to heat. The desired temperature (20°C) is the setpoint. The state of the heater (e.g. the setting of the valve allowing hot water to flow through it) is called the manipulated variable since it is subject to control actions.

Point of Sale System: Point of sale (POS) or checkout is the location where a transaction occurs. A “checkout” refers to a POS terminal or more generally to the hardware and software used for checkouts, the equivalent of an electronic cash register.

A POS terminal manages the selling process by a salesperson accessible interface. The same system allows the creation and printing of the receipt.

ATM machine process control: An automated teller machine (ATM), also known as a Cash Machine and by several other names, is a computerised telecommunications device that provides the clients of a financial institution with access to financial transactions in a public space without the need for a cashier, human clerk or bank teller. On most modern ATMs, the customer is identified by inserting a plastic ATM card with a magnetic stripe or a plastic smart card with a chip, which contains a unique card number and some security information such as an expiration date. Authentication is provided by the customer entering a personal identification number (PIN).
Using an ATM, customers can access their bank accounts in order to make cash withdrawals, credit card cash advances, and check their account balances as well as purchase prepaid cell phone credit.
1.2 (d) Types of User Interface

A computer is used by a person who needs to communicate with the machine in order to instruct it as to their wishes. The person also receives outputs from the computer. The means of communication between the user and the machine is known as the user interface and consists of both hardware and software. There are different types of interface, which are useful in different situations and for different types of user.

Form based
If the majority of the input to a system is of a standard type, in other words the computer knows what sort of input to expect, then a typical interface will produce a form on the screen to be filled in. This sort of interface would be used where an operator is inputting information while asking a customer questions over the telephone. The interface
- prompts the operator to ask all the questions
- makes the operator input the information in the correct order
- ensures that the information is input in the correct format by having specific areas to input the data
- makes the checking of the information easier.

The characteristics of a form based interface are that
- it has specified areas for the data. For example, boxes for input of coded material like the date or the sex of the customer, and areas to be filled in with textual information
- it has a cursor which moves to the next box to be filled in, sometimes the box is highlighted to make it clear to the operator where the data is to be inserted
- some of the boxes are more important than others and the cursor will not move on until some data has been supplied
- it checks that what has been input is sensible for that box before moving on to the next.

Menu based
Menu based interfaces are used in situations where the operator tends not to know what the options are that are available. Examples of this would be information systems for tourists or users of a particular service. A list of choices is made available followed by a further set of choices based on the first choice, and so on until the result is obtained. Imagine a system at a train station in a popular holiday location. The first screen may ask for the general area of interest (accommodation, trips, shopping, entertainment..), once the choice of accommodation has been made the next screen may offer different standards of hotels. The next screen may offer different price bands, and finally a list of all the available properties that match the previous choices. Input is often done using a touch screen because of the location of such systems and because the people who use them are often in no way computer literate, meaning that simple systems are essential.

Graphical
Graphical interfaces are called GUI (graphical user interface) or WIMP (windows, icons, menus, pointer). The terms describe what the user sees on the screen. There are many
different types, but the user would expect to be able to view different applications or files on the screen, this is done by putting each into its own boarded area known as a window. The user will expect to be able to select options by use of menus of choices and by using small pictures which represent the different options available. Choices are selected by the user by using some sort of pointing device to indicate choice, typically this would be a mouse.

Natural language  
Sometimes referred to as a conversational interface, the computer will ask questions which elicit a response which gives the user the impression that they are talking to the computer. The trick is that the system restricts itself to questions to which the only sensible answers are the ones that it knows. If the user leaves the expected responses, a message is produced which makes clear that a further attempt is required?

Command line  
Or command based interface is one where the user types a series of commands at the keyboard which tell the computer what their intentions are. The user needs to know what the possible commands are, and also needs to understand the way files are stored on the system. The characteristics of a command based interface are:

• the user needs to know what commands are available
• the user needs to understand the commands
• the user needs to understand the way that material is stored in the computer system

Because of the above points there are two very important characteristics about a command based interface. First, the system is very much more open than in the other types of interface. Other interfaces restrict the options that the user has available to them. This can be particularly important for the system manager because different users can only be allowed to have access to specific parts of the system. The second characteristic is that command based interfaces can only be used by computer literate people because you not only have to understand the commands and their uses but you also need to understand something about how the computer operates, particular how information is stored.

There are many other points to be made about interfaces, especially the fact that the second part of the interface consists of the hardware necessary to put the software interface into operation. These points will be made when the appropriate stage is reached in this course.
1.2 (e) Utility software

Utility programs are part of the operating system of a computer and are routines which carry out important tasks which are necessary from time to time on the system. They do not produce a pretty picture, or a letter which can be sent to someone so they are not application programs, but the types of job that they do are so important that the system could not run without them. There are many examples of utility software but we shall limit ourselves to just a few:

1. The surface of a disk can store so much data that the computer cannot handle it all at once so it needs to be split up so that data stored on it can be found again. When it is new a disk surface is blank so the computer “draws lines” on the surface to split it into small areas. The process is called formatting and it is carried out by a utility program called a disk formatter.

2. When files are to be stored on a disk they need to be managed. This includes arranging for them to be stored in a certain place, finding files again after they have been stored, moving them around to get them in a sensible order and deleting them when they are not needed any more. All of these actions are carried out by utility programs called file handlers.

3. A computer system is made up of lots of different sorts of hardware, each piece of which needs instructions about how to work. These programs are called hardware drivers.

4. Some files are very large. In most files it is possible to find simple ways of reducing the size of the file while keeping all its meaning. This can be very important when files are being sent from one computer to another as the communication is speeded up. The programs that reduce the size of files are called file compressors.

5. When files are being sent from one computer to another it is possible that they may contain a virus which will infect the receiving computer. A virus checker (scanner, killer...) is a utility program which keeps a constant check on files searching for viruses which it will delete if it finds any.
Example Questions.

1. Define what is meant by the term *operating system.* (2)

2. Give **two** reasons why an operating system is likely to be stored on backing storage rather than in the memory of the computer. (2)

3. Distinguish between a *multi-tasking* and a *multi-access* operating system. (2)

4. State what is meant by a *distributed system,* and give an advantage of this type of multi-access system over a simple network of machines. (2)

5. A computer operator takes phone calls from the public who ring up asking whether a particular item in a catalogue is available. The operator needs to type in a series of responses to questions put to the caller, so that the computer can check the file and determine whether there are any of that item available. Design a screen interface that would be suitable for the operator to use. (4)

6. The technician responsible for maintaining the system in question 5, uses a command line interface.
   a) Explain what is meant by a *command line interface.* (2)
   b) Give **two** advantages and **one** disadvantage to the technician of using a command line interface rather than a menu based interface. (3)

7. State **three** different types of utility software and explain why they are necessary in a computer system. (6)