OCR Computer Science AS Level

1.2.1 Systems Software

Concise Notes

Specification:

1.2.1 a)

- **Operating Systems**
  - The need for, function and purpose of operating systems

1.2.1 b)

- **Memory Management**
  - Paging
  - Segmentation
  - Virtual Memory

1.2.1 c)

- **Interrupts**
  - Role of interrupts
  - Role of ISR (Interrupt Service Routines)
  - Role of interrupts within Fetch-Decode- Execute Cycle

1.2.1 d)

- **Scheduling**
  - Round robin
  - First come first served
  - Multi-level feedback queues
  - Shortest job first
  - Shortest remaining time

1.2.1 e)

- **Types of operating system**
  - Distributed
  - Embedded
  - Multi-tasking
  - Multi-user
  - Real Time

1.2.1 f)

- **BIOS**

1.2.1 g)

- **Device drivers**

1.2.1 h)

- **Virtual machines**
  - Where software is used to take on the function of a machine for:
    - Executing intermediate code
    - Running an operating system within another

Operating Systems

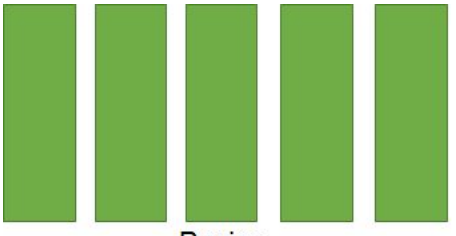
- A collection of programs that provide an interface between the user and computer
- Provide the following features:
  - Memory management
  - Resource management eg. scheduling
  - File management
  - Input/ Output management
  - Interrupt management
  - Utility software
  - Security
  - User interface

Memory Management

- Main memory is often not large enough to store all of the programs being used.
- Paging, segmentation and virtual memory are techniques used by the operating system to ensure memory is shared effectively by programs.

Paging

- Memory is split up into equal-sized sections known as pages.
- Pages are swapped between main memory and the hard disk as needed.



Segmentation

- Memory is split up into logical sized divisions, called segments.
- Segments vary in size
- Segments represent the structure and logical flow of the program.

Virtual Memory

- A section of the hard drive acts as RAM when the space in main memory is insufficient to store programs being used.
- Sections of programs not currently being used are temporarily moved into virtual memory through paging.
- This frees up memory for other programs in RAM.
- The key issue with using these techniques is disk thrashing, when the computer 'freezes' due to pages being swapped too frequently between the hard disk and main memory.

Interrupts

- Signals generated by software or hardware to indicate to the processor that a process needs attention.
- Interrupts have different priorities and this is considered when allocating processor time.
- They are stored within an abstract data structure called a priority queue in an interrupt register.

Interrupt Service Routine

- Processor checks the interrupt register at the end of each Fetch-Decode-Execute cycle.
- If there is an interrupt exists with a higher priority to the current process
    - The current contents of the registers in the CPU are transferred into a stack.
    - The relevant interrupt service routine (ISR) is loaded into RAM.
    - A flag is set to signal the ISR has begun.
    - The flag is reset once the ISR has finished.
    - This process is repeated.
- If there are no interrupts with a higher priority to the current process
    - The contents of the stack are popped back into the registers
    - Fetch-Decode-Execute cycle resumes

Scheduling

- Operating system ensures all sections of programs being run (known as 'jobs') receive a fair amount of processing time using scheduling.
- Scheduling algorithms can either be:
    - Pre-emptive
        - Jobs are actively made to start and stop by the operating system.
        - For example: Multilevel Feedback Queues, Shortest Remaining Time, Round Robin
    - Non pre-emptive
        - Once a job is started, it is left alone until it is completed.
        - For example: First Come First Served, Shortest Job First

Below are the scheduling algorithms you need to know:

- Round robin
    - Each job is given a section of processor time - a time slice - within which it runs.
    - Once each job in the queue has used its first time slice, they are given another slice of processor time until a job has been completed
    - Completed jobs are removed from the queue.

    Advantages:
        - All jobs will eventually be attended to.

    Disadvantages:
        - Longer jobs will take a much longer time for completion
        - Round robin does not take into account job priority or urgency

- First come first served

    Jobs are processed in chronological order by which they entered the queue.

    Advantages:
        - Straightforward to implement

    Disadvantages:
        - Does not take into account job priority or urgency

- Multilevel feedback queues

    This makes use of multiple queues, each which is ordered based on a different priority.

    Advantages:
        - Takes into consideration different job priorities

    Disadvantages:

- Difficult to implement

- Shortest job first

  The queue storing jobs to be processed is ordered according to the time required for completion, with the longest jobs being serviced at the end.
  Advantages:
  - Suited to batch systems, as waiting time is reduced

  Disadvantages:
  - Requires processor to calculate how long each job will take
  - Processor starvation if short jobs are continuously added to the queue
  - Does not take into account job priority or urgency

- Shortest remaining time

  The queue storing jobs to be processed is ordered according to the time left for completion, with the jobs with the least time to completion being serviced first.
  Advantages:
  - Throughput is increased as shorter processes can be quickly completed

  Disadvantages:
  - Does not take into account job priority or urgency
  - Processor starvation if short jobs are continuously added to the queue

Types of Operating System

- Distributed
  - Run across multiple devices
  - Means the load of a task is spread across multiple computer processors

- Embedded
  - Built to perform a small range of specific tasks
  - Catered towards a specific device eg. a household appliance.
  - Limited functionality and hard to update
  - Consume less power than other types of OS.

- Multi-tasking
  - Enables user to carry out tasks simultaneously.
  - Time slicing is used to switch quickly between programs and applications in memory.

- Multi-user
  - Multiple users make use of one computer.
  - A scheduling algorithm Is used to allocate processor time fairly between jobs and prevent processor starvation.
- Real Time
  - Used in time-critical computer systems and designed to perform a task within a guaranteed time frame.

BIOS

- Basic Input Output System is the first program that runs when a computer system is switched on.
- Runs a series of tests before the operating system is loaded into memory:
- POST (Power-on self test) which ensures that all hardware is correctly connected and functional
- Checking the CPU clock, memory and processor
- Testing for external memory devices

Device Drivers

- Computer programs that allow the operating system to interact with hardware.
- Are specific to the computer's architecture.
- Specific to the operating system

Virtual Machines

- A theoretical computer and a software implementation of a computer system.
- Provides an environment with a translator for intermediate code to run.

Intermediate Code

- Code halfway between machine code and object code is called intermediate code.
- Independent of processor architecture so can run across different machines and operating systems.
- Takes longer to execute than low-level code

Uses of virtual machines include:

- Testing programs
- Protection from malware
- Running software compatible with different versions/ types of operating systems