1)Explain the difference betwqeen polling and interrupt?

|  |  |  |
| --- | --- | --- |
| **Aspect** | **Polling** | **Interrupt** |

|  |  |  |
| --- | --- | --- |
| **Initiation** | CPU **initiates** the check for peripheral status. | Peripheral **initiates** a signal to the CPU. |

|  |  |  |
| --- | --- | --- |
| **CPU Utilization** | **Inefficient**: CPU continuously checks devices, wasting time. | **Efficient**: CPU only reacts when signaled by a device. |

|  |  |  |
| --- | --- | --- |
| **Responsiveness** | **Slower**: Device servicing depends on polling frequency. | **Faster**: Immediate response when a device needs attention. |

|  |  |  |
| --- | --- | --- |
| **Complexity** | **Simple**: Easier to implement, less hardware/software overhead. | **Complex**: Requires additional hardware (interrupt controller) and software for ISR. |

|  |  |  |
| --- | --- | --- |
| **Overhead** | CPU time is wasted in **busy-waiting**. | Requires saving and restoring the CPU state, adding some **overhead**. |

|  |  |  |
| --- | --- | --- |
| **Use Cases** | Suitable for **simple tasks** or when response time is not critical. | Suitable for **real-time systems** where quick response is crucial |

2) Give 4 most important features of operating system?

 **Process Management**: Handles process scheduling, multitasking, and ensures efficient CPU utilization.

 **Memory Management**: Manages allocation, deallocation, and protection of memory, as well as uses virtual memory for improved performance.

 **File System Management**: Provides file organization, access control, and data security to efficiently manage the user's data.

 **Device Management**: Manages hardware devices through drivers, schedules I/O, and ensures seamless data transfer.

OTHERS:

* **User interface** - Almost all operating systems have a user interface (**UI**).
  + Varies between **Command-Line** (**CLI**), **Graphics User Interface** (**GUI**),**touch-screen,  Batch**
* **Program execution** - The system must be able to load a program into memory and to run that program, end execution, either normally or abnormally (indicating error)
* **I/O operations** -  A running program may require I/O, which may involve a file or an I/O device
* **File-system manipulation** -  The file system is of particular interest. Programs need to read and write files and directories, create and delete them, search them, list file Information, permission management.
* **Communications** – Processes may exchange information, on the same computer or between computers over a network
  + Communications may be via shared memory or through message passing (packets moved by the OS)
* **Error detection** – OS needs to be constantly aware of possible errors
  + May occur in the CPU and memory hardware, in I/O devices, in user program
  + For each type of error, OS should take the appropriate action to ensure correct and consistent computing
  + Debugging facilities can greatly enhance the user’s and programmer’s abilities to efficiently use the system
* **Resource allocation -** When  multiple users or multiple jobs running concurrently, resources must be allocated to each of them
  + Many types of resources -   CPU cycles, main memory, file storage, I/O devices.
* **Logging -** To keep track of which users use how much and what kinds of computer resources
* **Protection and security -** The owners of information stored in a multiuser or networked computer system may want to control use of that information, concurrent processes should not interfere with each other
  + **Protection** involves ensuring that all access to system resources is controlled
  + **Security** of the system from outsiders requires user authentication, extends to defending external I/O devices from invalid access attempts

3) What is the difference between kernel and user space?

The **kernel space** and **user space** are two distinct regions of memory in an operating system, each serving different purposes and having different levels of privileges. **Kernel space** is the part of the computer's memory where the **kernel** (the core component of the operating system) executes and provides its services. The kernel is responsible for directly interacting with hardware and managing system resources like CPU, memory, devices, and more. **User space** is the memory area where **user applications** run. These applications include everything a user interacts with, such as word processors, browsers, games, etc.