**NAME: MUHAMMAD UMAR KHAN**

**CID: 106342**

**STUDENT ID: 10619**

**ASSIGNMENT NO: 01**

**QUESTION NUMBER: 01**

Example of Hardware and software in Operating System are following:

**HARDWARE INTERRUPT:-**

* If you are using a word processor and press a key, the program must process the input immediately. Typing “Hello” creates five interrupt requests, which allow the program to display the letters you need.
* Similarly each time you click a mouse button or trap on touchscreen, you send an interrupt signal to the device.

**SOFTWARE INTERRUPT:-**

* If a program expects a variable to be a valid number, but the value is null, an interrupt may be generated to prevent the program from crashing. It allow the program to change course and handle the error before crashing.
* Similarly, an interrupt can be used to break an infinite loop, which could create a memory leak or cause a program to be unresponsive.

**QUESTION NUMBER: 02**

When a memory management fault occurs and if the memory management handler is enabled, the memory management fault handler will be executed. If the fault occurs at the same time a higher-priority exception takes place, the other exceptions will be handled first and the memory management fault will be pended. If the processor is already running an exception handler with the same or higher priority or if the memory management fault handler is not enabled, the hard fault handler will be executed instead. If a memory management fault takes place inside the hard fault handler or the NMI handler, the processor will enter the lockup state.

**QUESTION NUMBER: 03**

**CPU PROTECTION:-**

It means that a process should not hog (hold) CPU forever otherwise other processes will not get the process. For that purpose, a timer is introduced to prevent such a situation. A process is given a certain time for execution after which a signal is sent to the process which makes the process to leave CPU. Hence process will not hog the CPU.

**I/O PROTECTION:-**

To ensure CPU protection OS ensure that below case should not occur

* View I/O of other process
* Terminate I/O of another process
* Give priority to a particular process I/O

If an application process wants to access any I/O device then it will be done through system call so that OS will monitor the task.

**QUESTION NUMBER: 04**

| Asymmetric Multiprocessing | Symmetric Multiprocessing |
| --- | --- |
| In asymmetric multiprocessing, the processors are not treated equally. | In symmetric multiprocessing, all the processors are treated equally. |
| Tasks of the operating system are done by master processor. | Tasks of the operating system are done individual processor |
| No Communication between Processors as they are controlled by the master processor. | All processors communicate with another processor by a shared memory. |
| In asymmetric multiprocessing, process are master-slave. | In symmetric multiprocessing, the process is taken from the ready queue. |
| Asymmetric multiprocessing systems are cheaper. | Symmetric multiprocessing systems are costlier. |
| Asymmetric multiprocessing systems are easier to design | Symmetric multiprocessing systems are complex to design |

**QUESTION NUMBER: 05**

| **Multiprogramming** | **Multi-tasking** |
| --- | --- |
| 1. | Both of these concepts are for single CPU. | Both of these concepts are for single CPU. |
| 2. | Concept of Context Switching is used. | Concept of Context Switching and Time Sharing is used. |
| 3. | In multiprogrammed system, the operating system simply switches to, and executes, another job when current job needs to wait. | The processor is typically used in time sharing mode. Switching happens when either allowed time expires or where there other reason for current process needs to wait (example process needs to do IO). |
| 4. | Multi-programming increases CPU utilization by organizing jobs. | In multi-tasking also increases CPU utilization, it also increases responsiveness. |
| 5. | The idea is to reduce the CPU idle time for as long as possible. | The idea is to further extend the CPU Utilization concept by increasing responsiveness Time Sharing. |