

Name: Qazi Azmat-ullah

ID: - 14448

Paper: Operating system.

Question #11. Process vs Thread

* Process means a program is in execution, whereas thread means a segment of a process.

* A process is not lightweight whereas threads are lightweight.

* A process takes more time to terminate and the thread takes less time to terminate.

* Process takes more time for creation whereas thread takes less time for creation.

* Process likely takes more time for context switching whereas threads take less time for context switching.

* A process is mostly isolated, whereas threads share memory.

* Process does not share data, and threads share data with each other.

Question #2Threads

A thread is a basic unit of CPU utilization, consisting of a program counter, a stack, and a set of registers, (and a thread ID)

Types of threads.

1 * User Level Thread (ULT) :- is

implemented in the user level library, they are not created using the system calls. Thread switching does not need to call operating system and to cause interrupt to kernel. Kernel doesn't know about the user level thread and manages them as if they were single-threaded processes.

Advantages of ULT:

- * can be implemented on an OS that doesn't support multithreading.
- * Simple representation since thread has only program counter, register set, stack space.
- * simple to create since no intervention of kernel.

DisAdvantages of ULT:

- * No or less co-ordination among the threads and kernel.
- * If one thread causes a page fault, the ~~events~~ entire process blocks.

2 - Kernel Level Thread (KLT).

Kernel knows and manages the threads. Instead of thread table in each process, the kernel itself has thread table (a master one) that keeps track

of all the threads in the system. In addition kernel also maintain traditional process table to keep track of the processes. OS kernel provides system call to create and manage threads.

Advantages of KLT

- * Since kernel has full knowledge about the threads in the system, scheduler may decide to give more time to processes having large number of threads.
- * Good for applications that frequently block.

Disadvantages of KLT

- * Slow and inefficient
- * It requires threads control block so it is an overhead.

Question # 3

Deadlock:

Deadlock is a situation where a set of processes are blocked because each process is holding a resource and waiting for an other resource acquired by some other process.

Example :

Deadlock can arise if following four condition hold simultaneously.

1. Mutual Exclusion

one or more than one resource are non-sharable (only one process can use at a time)

2. Hold and wait: A process is holding at least one resource and waiting for resources.

3. No Preemptions

A resource cannot be taken from a process unless the process releases the resource.

4. Circular wait: A set of processes are waiting for each other in circular form.

Question # 4

A solution to the critical section problem must satisfy the following three requirements.

1. Mutual Exclusion

Out of a group of cooperating processes, only one process can be in its critical section at a given point of time.

2. Progress

If no process is in its critical section, and if one or more threads want to execute their critical section then ~~they~~ any of these threads.

must be allowed to get into its critical section.

3 Bounded Waiting

After a process makes a request for getting into its critical section, there is a limit for how many other processes can get into their critical section, before this process's request is granted. So after the limit is reached, system must grant the process permission to get into its critical section.

Question #6

Logical vs Physical Address space:

1 The Basic difference between logical and physical address is that logical address is generated by CPU in perspective of a program whereas the physical address is a location that exists in the memory unit.

2 Logical Address Space is the set of all logical addresses generated by CPU for a program whereas the set of all physical addresses mapped to corresponding logical addresses is called physical address space.

3 The logical address does not exist physically in the memory where physical address is a location in the memory that can be accessed physically.

4. The logical address is generated by the CPU while the program is running whereas the physical address is computed by the memory management unit (MMU).

Question # 5

Dynamic Loading

Dynamic loading is a ~~mechanism~~ mechanism by which a computer program can, at run time, load a library or other binary into memory, retrieve the addresses of functions and variables contained in the library, execute those function or access those variable and unload the library from memory. It is one of the 3 mechanisms by which a computer program can use some other software; the other two are static linking and dynamic linking. Dynamic loading allows a computer to start up in the absence of these libraries, to discover available libraries and to potentially gain additional functionality.

Dynamic Linking In computer,

a dynamic linker is the part of an operating system that loads and links the shared libraries needed by an executable when it is executed.

(at "runtime"), by copying the content of libraries from pre-persistent storage to RAM, filling jump tables and relocating pointers.

Linking is often referred to as a process that is performed when the executable is compiled, while a dynamic linker is a special part of an operating system that loads external shared libraries into a running process and then binds those shared libraries dynamically to the running process. This approach is also called dynamic linking.

The End
