Mawlana Bhashani Science and Technology University

Lab-Report

Lab Report No: 03

Course code: ICT-3110

Course title: Operating System Lab

Date of Performance:

Date of Submission:

Submitted by

Name : Mst.Zakia Sultana

ID : IT-18027

Session : 2017-18

Year : 3rd Semester : 3rd

Dept of ICT , MBSTU

Submitted To

Nazrul Islam

Assistant Professor

Dept of ICT

MBSTU

Experiment No : 03

Experiment Name : Threads on Operating System.

Objectives:

i) What is Thread?
ii) Types of Threads.
iii) Implementation of Threads.

i) What is Thread?

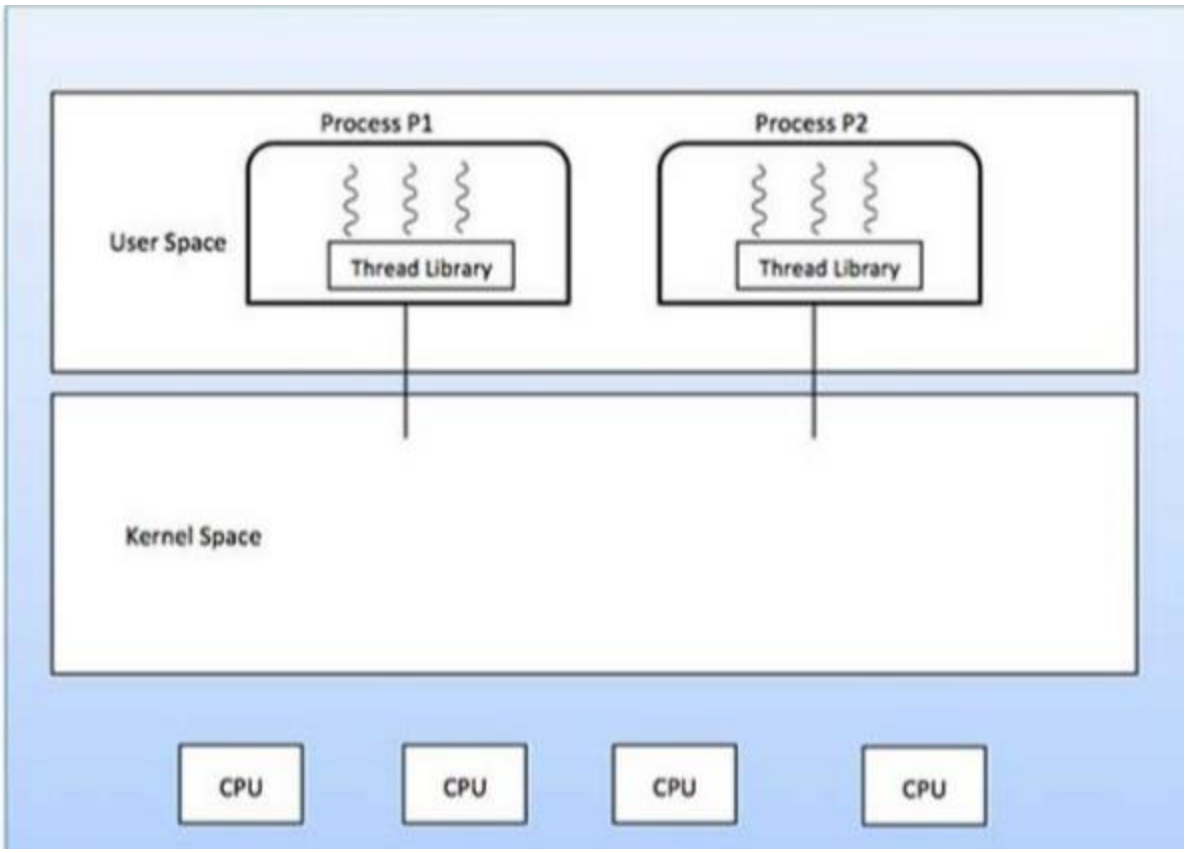
Ans: A thread is a flow of control within a process . A process can contain multiple threads.

ii) Types of Threads.

Ans: Thread are two types:-

i) User Level Threads
ii) Kernel Level Threads

User level threads (ULT): User level threads are threads that are visible to the programmer and are unknown to the kernel. The thread library contains code for creating and destroying threads, for passing message and data between threads, for scheduling thread execution and for saving and restoringqq thread contexts.



Advantages of ULT –

- Can be implemented on an OS that does't support multithreading.
- Simple representation since thread has only program counter, register set, stack space.
- Simple to create since no intervention of kernel.
- Thread switching is fast since no OS calls need to be made.

Disadvantages of ULT –

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

Kernel Level Threads (KLT): The operating system kernel supports and manages thread. In this case, thread management is done by the Kernel. There is no thread management code in the application area.

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.

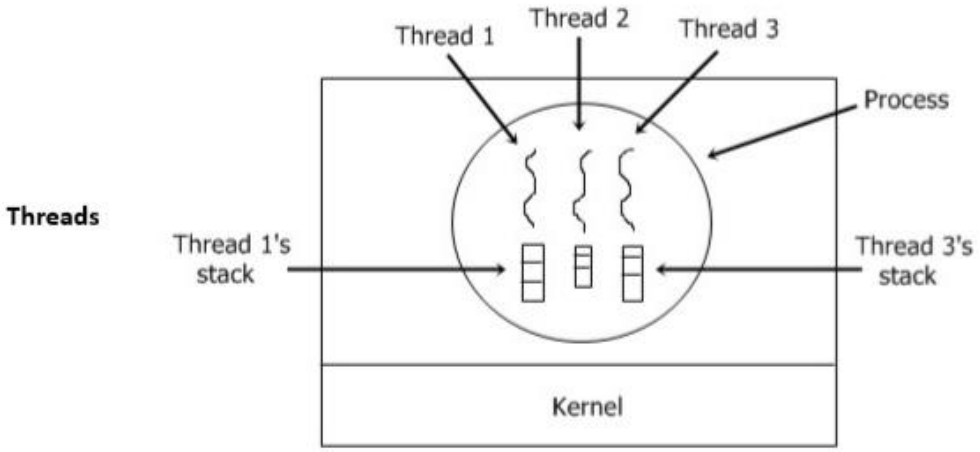
Disadvantages of KLT –

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

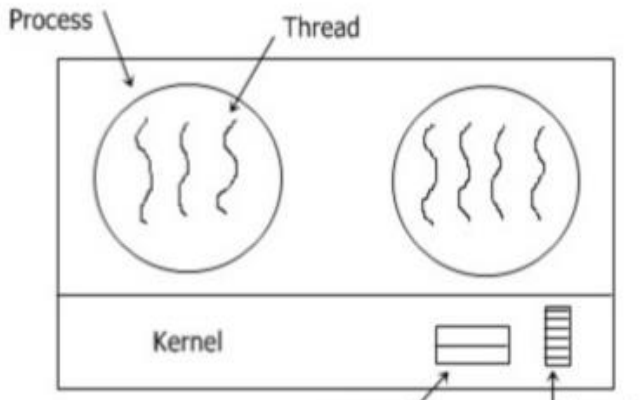
iii) Implementation of Threads.

Ans: There are two ways of implementing a thread package:
1. In user space
2. In kernel

Threads implementation in the user space:
In this model of implementation, the threads package entirely in user space, the kernel has no idea about it. A user-level threads package can be executed on an operating system that doesn't support threads and this is the main advantage of this implementation model i.e. Threads package in user space.



Implementation in the kernel:
In this method of implementation model, the threads package completely in the kernel. There is no need for any runtime system. To maintain the record of all threads in the system a kernel has a thread table.
A call to the kernel is made whenever there is a need to create a new thread or destroy an existing thread. In this, the kernel thread table is updated.



Other two methods are as follows:

- Hybrid implementation
- Scheduler activation

Hybrid implementation:

In this implementation, there is some set of user-level threads for each kernel level thread that takes turns by using it.

Scheduler activation:

The objective of this scheduler activation work is to replicate the working or function of kernel threads, but with higher performance and better flexibility which are usually related to threads packages which are implemented in userspace**.**