OSSA – Lec 4 – Threads.

Learning Topics

1. *Threads Concept*
2. *Thread Models*

1. Threads Concept

Threads

- Threads are running within application.
- Threads are running parellaly and they are performing various activities to support the user. *Ex: MS Word*
- Most of modern applications are **multithreaded**. *Ex: Operating Systems*
- A Thread is a **basic unit** of **CPU Utilization**.
- Every Thread execution control by its own **Program Counter**.
- Thread can **create Child Threads**.
- Threads are **not independent** of one another.

Benfits of Threads

- Responsiveness
- Resource Sharing
- Economy
- Scalability (*We can easily scale activities of threads to different CPUs*.)

2 Types of Threads

User Threads	Kernel Threads
Managing by ***User-Level Thread Library.*** • POSIX Pthreads • Windows threads • Java threads	Supported by the ***Kernel / OS.*** • Windows • Linux • Mac OS X
Created by ***users in their applications.***	Kernel Threads are using by ***executing User-Level Threads.***

2. Thread Models

Multithreading Models

Many-to-One	One-to-One	Many-to-Many
Many user-thread *mapping* to ***1 kernel-thread***	***1 user-thread*** mapping to ***1 kernel-thread***	***Many user-thread*** mapping to ***Many kernel-thread***
1 user-thread run with ***only 1 kernel-thread at a time.***	Each user-thread run with ***each kernel-thread.***	user-threads can run with ***different kernel-threads.***
ISSUE 1 - if a user-thread is block ***also kernel-thread is block.*** ***ISSUE 2*** - threads ***can't execute parellaly.***	***Solve ISSUE 1*** - if a user-thread is block that particular kernel-thread is block but ***other threads are working properly.*** ***Solve ISSUE 2*** - ***Parellel*** execution is ***possible.*** ***ISSUE 3*** - No. user-threads determines No. kernel-threads ***(there ll'be a issue when user-t count getting higher.)***	***Solve ISSUE 3*** - No. kerenl-threads ***decide using the previous usage.***
Ex : • ***Solaris Green Threads*** • ***GNU Portable Threads***	*Ex : Using by* • ***Windows*** • ***Linux*** • ***Sloaris***	*Ex :* • ***Solaris Prior to Ver. 9*** • ***Windows with ThreadFiber Package***

Thread Libraries

- Thread Library **provides API (Application Programming Interface) to programmer for Create and Manage Threads. - Use of Thread Library**
- Creating user-threads we need **Pthread Library** and use the library functions in Pthread library.
- Pthread library **provided by UNIX threads.**

Thread Pools

- Pool of Threads or **Collection Threads.**
- **Latest OS** are **using this concept**.

Signal Handling

- Use to notify a process.
- Signals
    1. **Generate**
    2. **Delivered**
    3. **Handled**

- **2 Types** of Signal handlers
    1. **default handler**
    2. **user-defined signal handler - can override default**

- **4 ways** of signal delivering for Multithreaded
    1. Signal delivers to **only thread that applies.**
    2. Singal delivers to **every thread.**
    3. Signal delivers to **selected threads.**
    4. Signal delivers to **assigned specific thread.**

Thread Cancellation

- Terminating thread before it has finished. - called **Cancel**
- That canceled thread called **target thread.**
- 2 cancelation approaches,

    1. **Asynchronous Cancellation** - Terminate target thread immediately.
    2. **Deferred Cancelation** - Check the thread suitable to cancel.