

PARADIGMS AND COMPUTER PROGRAMMING FUNDAMENTALS (PCPF)

**ITC305
2022-23**



Subject In-charge

Ms. Aaysha Shaikh

Professor Dept. of Information Technology
SFIT Room No. 322

email: aayshashaikh@sfit.ac.in

Module 5

Alternative Paradigm: Concurrent Programming

Concurrent programming fundamentals,
Implementing synchronization, Message passing,
Multithreaded programs, Communication and
synchronization, Language and libraries, Thread
creation Syntax.

Self-Learning Topic: : Study Implementation of concurrency
concepts for real time application.

12-Oct-22 Ms. Aaysha Shaikh 2

What is concurrent?

h

- Concurrent describes things that are occurring, or people who are doing something, at the same time. Such as "concurrent users" of a computer program.

- On the other hand, consecutive refers to things that are arranged or happen in a sequential order.

Real Life Example:

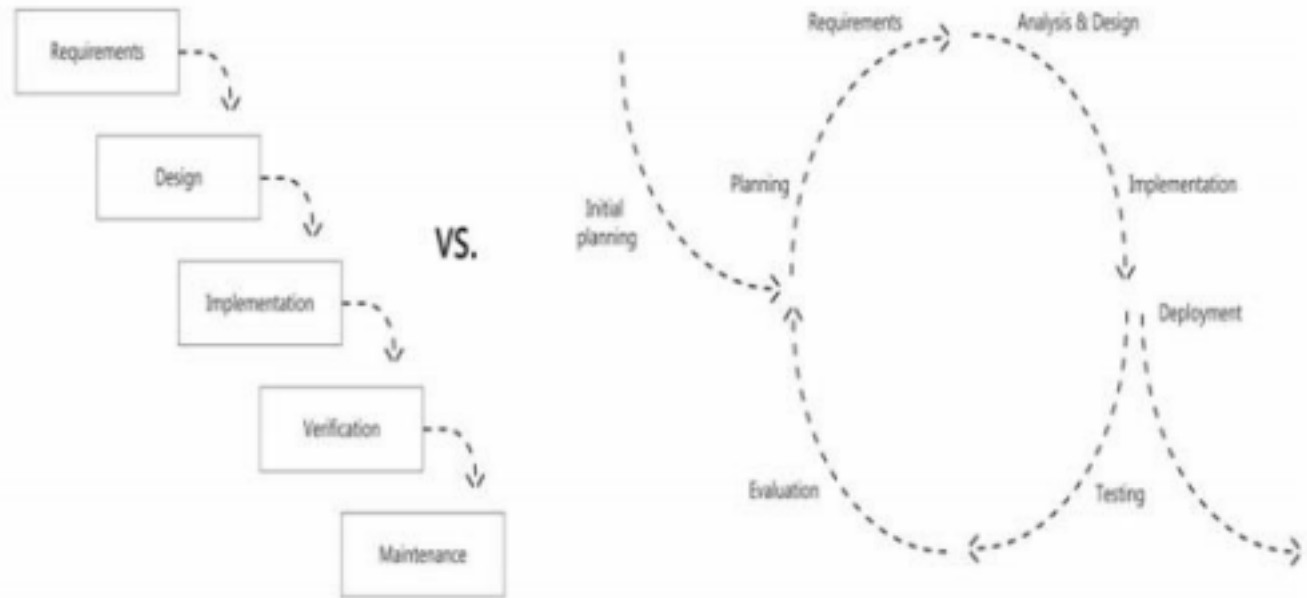


Fig. 1.3. Sequential Vs Concurrent Engineering

12-Oct-22

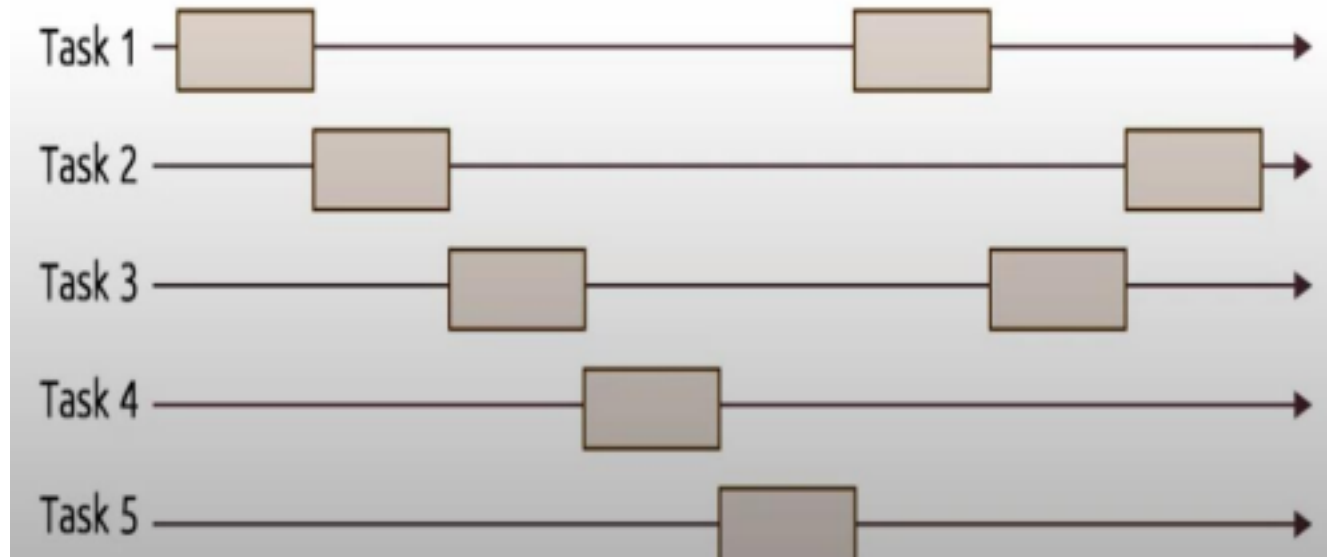
Ms. Aaysha Shaikh 3

What is concurrent programming/Computing?

- In a concurrent program several stream of operations may execute concurrently. Each stream of operation executes as it

would in a sequential program except for the fact that stream can communicate and interfere with one another.

- Concurrency is the property of a system which enables overlapping of process lifetimes.



12-Oct-22 Ms. Aaysha Shaikh 4

Concurrency Vs Parallelism

- Dealing with multiple things at once versus doing multiple things at once.

Concurrency

Tasks start, run, and complete in overlapping time periods



asyncio

Parallelism

Tasks run simultaneously



threads / processes
+ multicore

12-Oct-22 Ms. Aaysha Shaikh 5

A simple story...

Sequential



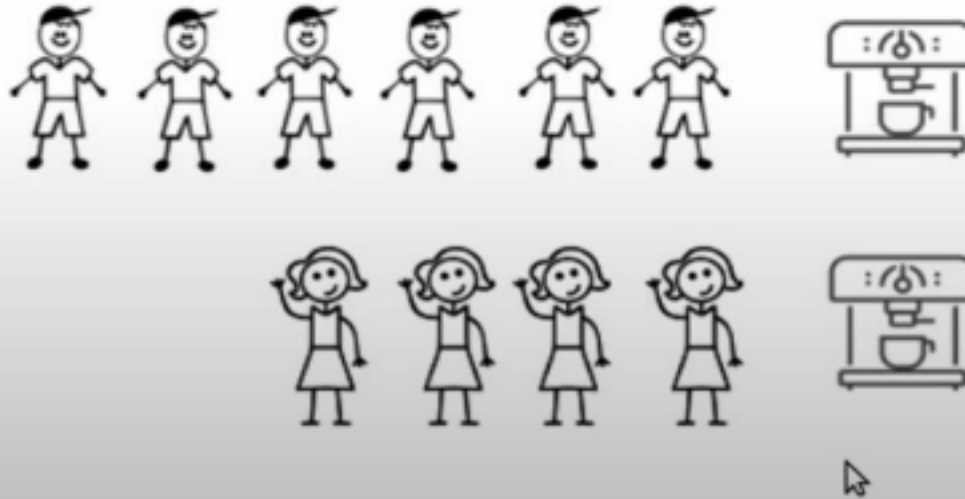
12-Oct-22 Ms. Aaysha Shaikh 6

Concurrent



12-Oct-22 Ms. Aaysha Shaikh 7

Parallelism



12-Oct-22 Ms. Aaysha Shaikh 8

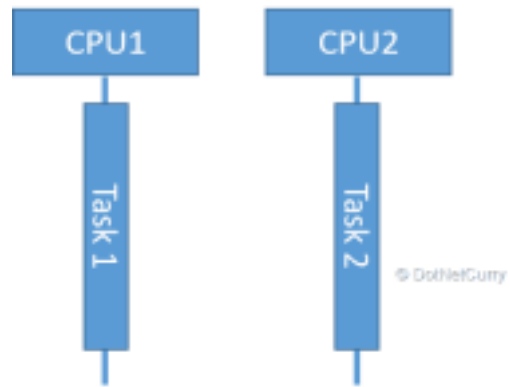
1- Concurrency generally refers to events or circumstances that are happening or existing at the same time.

2- In programming terms, concurrent programming is a technique in which Two or more processes start

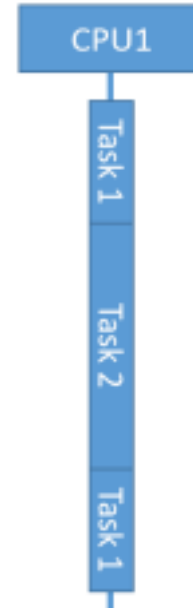
Run in an interleaved fashion through switching and

Complete in an overlapping time period by managing access to shared resources

Parallel



Concurrent

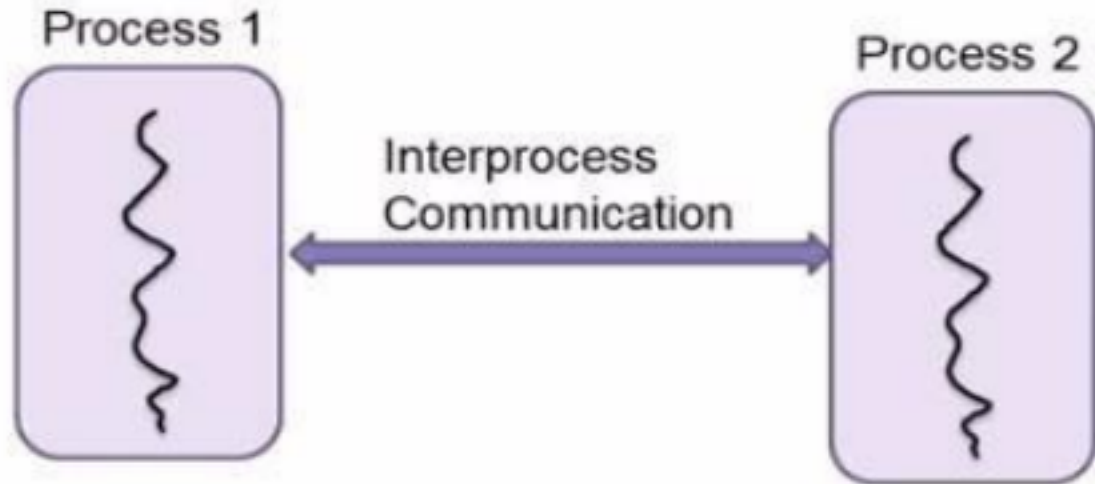


12-Oct-22

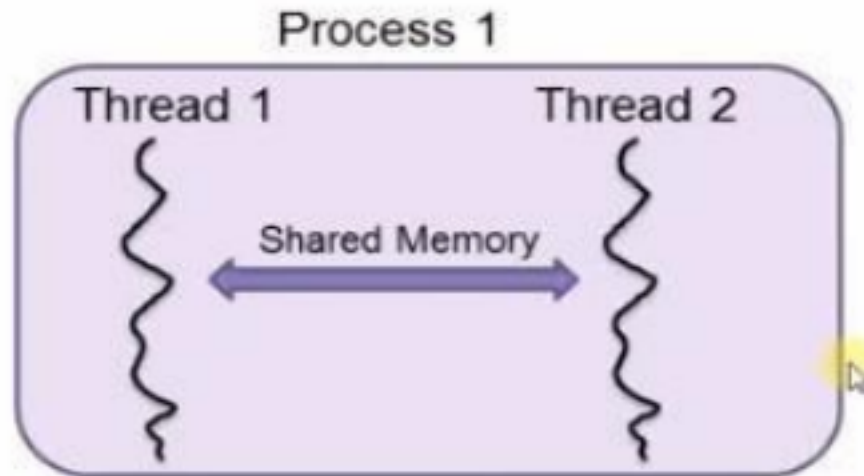
Ms. Aaysha Shaikh 9

Process vs. Threads

Multiprocessing:

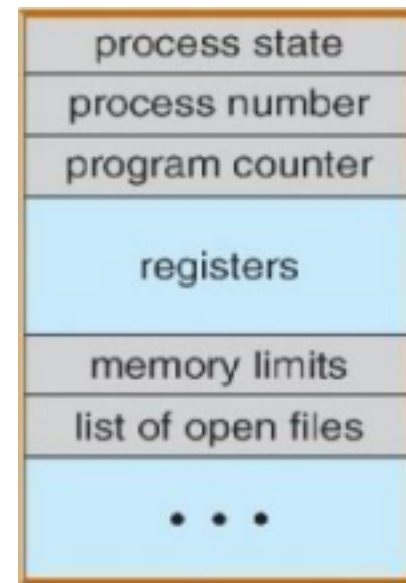


Multithreading:



• Process

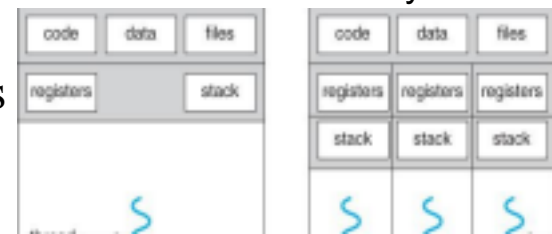
- Process means any program is in execution.
- Process control block contains information about processes for example Process priority, process id, process state, CPU, register, etc.
- A process can create other processes which are known as **Child Processes**.
- Process takes more time to terminate and it is isolated means it does not share memory with any other process.
- Process is called heavy-weight process



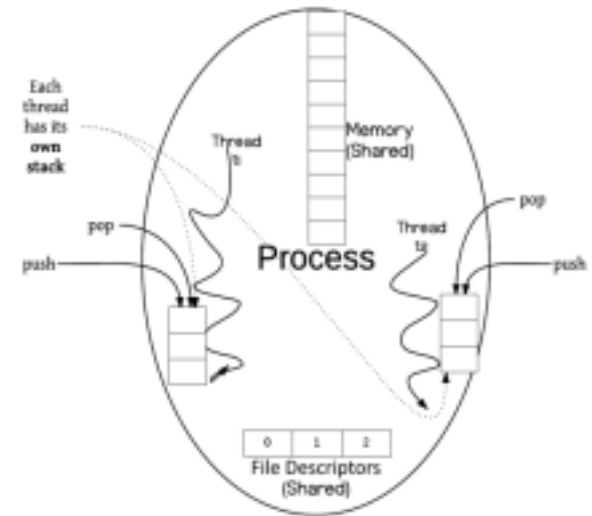
Thread

- One or more **threads** run in the context of the **process**.
- A **thread** is the basic unit to which the operating system allocates processor time.
- A **thread** can execute any part of the **process** code, including parts currently being executed by another **thread**

- In a concurrent program, the term thread defines the present entity which the programmer might think as running concurrently with other threads.



- In a concurrent program, several streams of operations may execute concurrently.
 - Each stream of operations executes as it would in a sequential program *except for the fact that streams can communicate and interfere with one another.*
 - Each such sequence of instructions is called a *thread*. ▪
- For this reason, sequential programs are often called *single threaded* programs.
- When a multi-threaded program executes, the operations in its various threads are interleaved
 - The operations for each stream are strictly ordered ▪ One stream may run very fast while another does not run at all.
 - In the absence of fairness guarantees a given thread can starve unless it is the only ``runnable" thread.



12

• Advantages

1. Run multiple applications at the same time.
2. Without concurrency, every application runs completely before the next one can start running. So concurrency provides an opportunity for everyone
3. Enables better performance by the operating system.

- **Disadvantages**

1. Coordinate with multiple applications via additional mechanisms. 2. In operating systems performance required overheads and complexities to switch among other applications.

3. Race condition :Several applications running concurrently can degrade performance. It is an undesirable situation which occurs when a system tends to perform two or more operations at the same time.

4. Deadlock: It occurs when more than two processes are blocked and any of processes cannot proceed to execute.

5. Starvation: It occurs when a high priority of processes keeps executing and low priority

12-Oct-22 Ms. Aaysha Shaikh 13

Thank You

12-Oct-22 Ms. Aaysha Shaikh 14