OSSA – Lec 5 – CPU Scheduling.

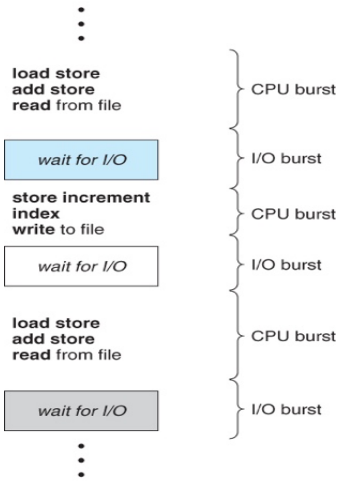
Learning Topics

1. Basic Concepts
2. Scheduling Criteria
3. Scheduling Algorithms
4. Algorithm Evaluation

1. Basic Concepts

CPU-I/O Burst Cycle

- CPU Burst followed by I/O Burst
- Process execution consist of a Cycle of CPU execution and I/O wait



- Mainly CPU scheduler is **involving in allocating the processes to the CPU.** (also called as Short-Term-Scheduler).
- STS is **acessing the ready queue → ready queue put the processes into CPU.**
- STS selects processes to run in the CPU in different situations.
    1. When a **Process is Terminated**.
    2. When a **Process want to Start a I/O Operations.**
    3. **Due to an Interrupt** the **running Process** has to **waiting in the ready queue.**

2 Types of Scheduling

1. ***Preemptive Scheduling***
    - When a process is running in the CPU, **process can be suspended at any time to start a new process.**

2. ***Non - preemptive Scheduling***
    - Process **not suspending in the middle of the execution**, after running process over CPU gets new process to execute.

Dispatcher Module

- Dispatcher **Stops one Process and Start another Process.** ← **Purpose**
- **Dispatcher Latency** is the Time takes to the Dispatcher Process.
- **Dispatcher Tasks,**
    1. Context Switch Time.
    2. Switching to User Mode.
    3. Jump in to the proper location of the user program to restart it.

2. Scheduling Criteria

CPU Utilization

- Keep the CPU busy as Possible.
- Higher Utilization is the Best.

Throughput

- No of Processes that are completing during some time period.
- Higher throughput is good because no of processes that are executes per time is high.

Turnaround Time

- Total time period a process takes to complete their execution.
- Including waiting time, running time.
- Lower Turnaround time is the best.

Waiting Time

- Total time a process is waiting in the ready queue.
- Lower waiting time is best.

Response Time

- Amount of time a process takes to response a request.
- Lower response time is best.

3. Scheduling Algorithms

First-Come First-Served (FCFS)

- Only consider about **Arrival Order of the Processes.**
- **Doesn't consider about Shorter and Longer Process.**
- Shorter process behind the longer process called **Convoy Effect.**

Shortest-Job-First (SJF)

- SJF is **Non-Preemptive.**
- Consider the **Shortest burst time (execution time) only.**

Shortest-Remaining-Time-First (SRTF)

- Combination of **(SJF + Preemptive).**
- **Depend on burst time** the process get priortized.

- During the execution, if a process with a minimum burst time arrives, **the current one will be suspended immediately**, and the next process will receive priority.

Priority Number

- **Smaller integer** has the **highest priority.**
- Lower priortize processes waiting called as **Starvation.** ← **Problem**
- With the time increase the priority of a process is called as **Aging.** ← **Solution**

Round Robin

- Using a special parameter called **Time Quantum.**
- Time Quantum refers to the **maximum amount of time a process can execute before the CPU is forcibly taken away and given to another process.**
- Each process is allocated a **fixed time slice**, also known as a time quantum.
- **Quantum Time getting lower** No of **Context Switches getting higher.**

Multilevel Queue

- Ready queue can partitioned into 2 separate queues,
    1. **Foreground queue**
    2. **Background queue**
- Each queue has its own scheduling algorithms.
- Also using **Fixed Priority Scheduling , Time Slice.**

Multilevel Feedback Queue

- Selecting the process for different queues they are using some parameters,
    - **No of queues available.**
    - **Scheduling algorithms of the queue.**
    - **What time the processes have to upgrade.**
    - **What time the processes have to demote.**
    - **Select which queue the process will enter when it need service.**

4. Algorithm Evaluation

- Evaluation of algorithms its using **4 different approaches.**

Deterministic Modeling

- Find the best algorithm by **using a paper work.**

Queueing Models

- **Looking at different parameters** like queue length, waiting time and arrival rate.

  Little's Formula

  n = no of processes can keep inside the queue
  W = average waiting time
  λ = average arrival rate

  $$n = \lambda \times W$$

Simulations

- **Develop S/W applications for different algorithms** and implement them inside the computer, as input **data sets** (**processes, arrival time, waiting time**) can be given.
- It is **more accurate.**

Implementation

- **Implement new scheduler** and test in real systems.
- **High cost** and **High risk.**
- Environment vary. (**Situation change according to the computer environment changes**).
- It is the **most accurate and complex.**