

CS 250

OPERATING SYSTEMS



Lecture 4

Scheduling

Instructor
Dr. Dhiman Saha

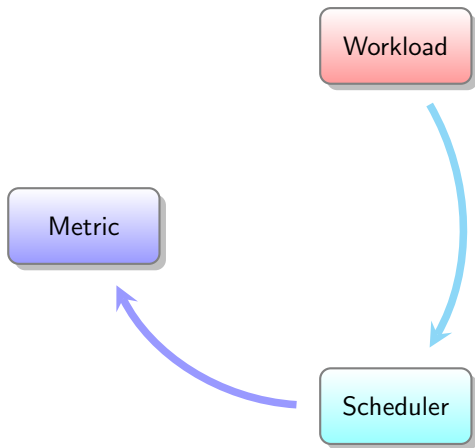
Processes running in the system

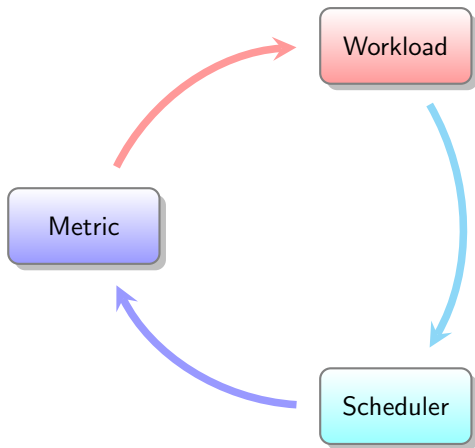


Workload

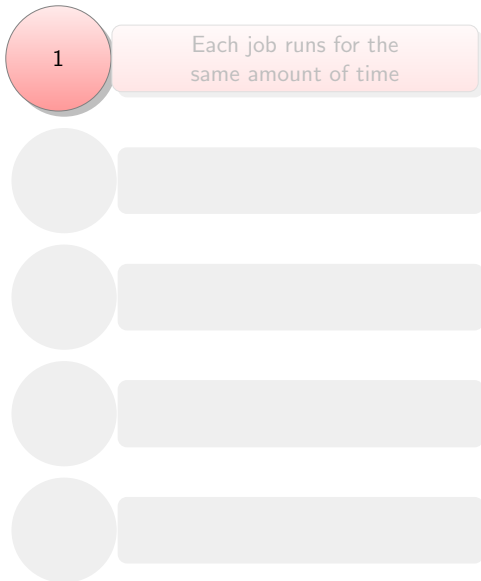


Measuring the scheduling “quality”

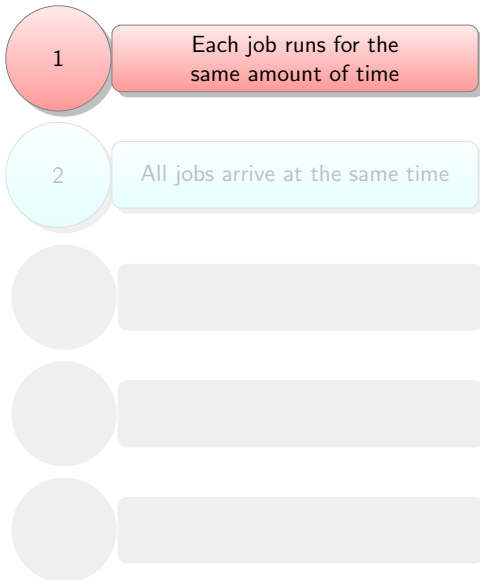




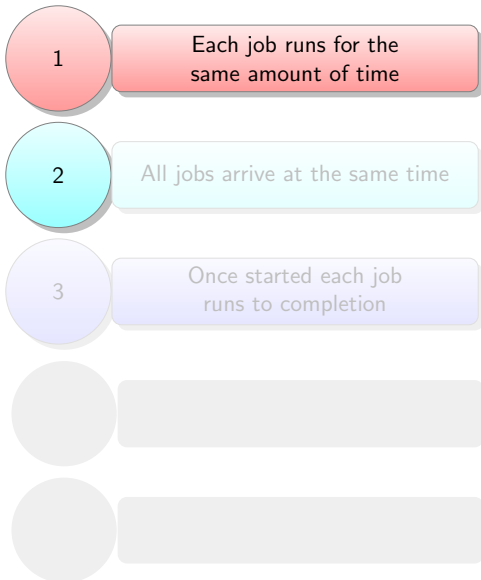
Workload Assumptions



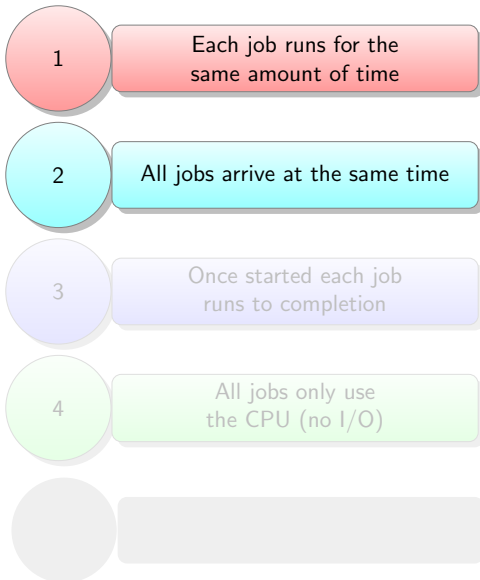
Workload Assumptions



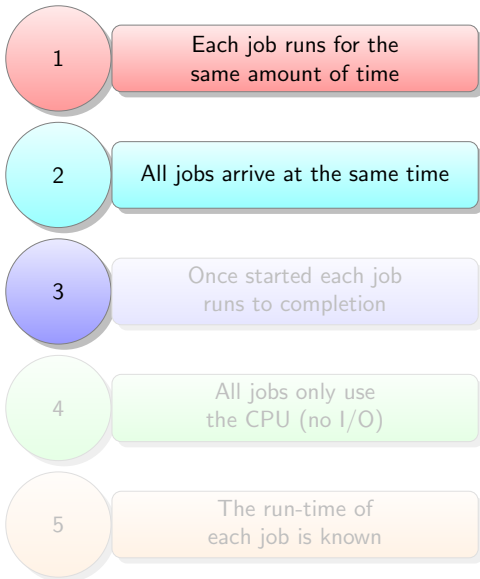
Workload Assumptions



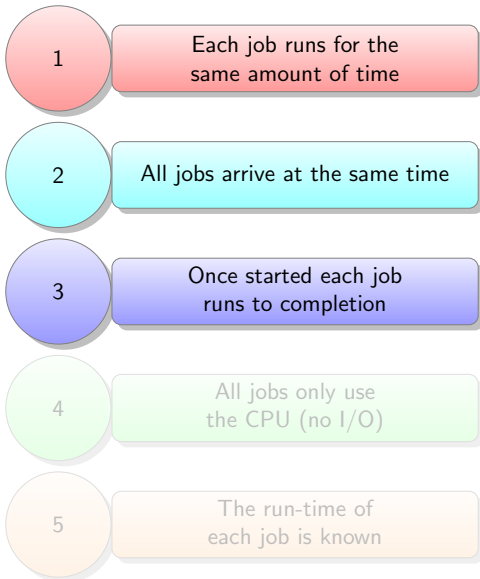
Workload Assumptions



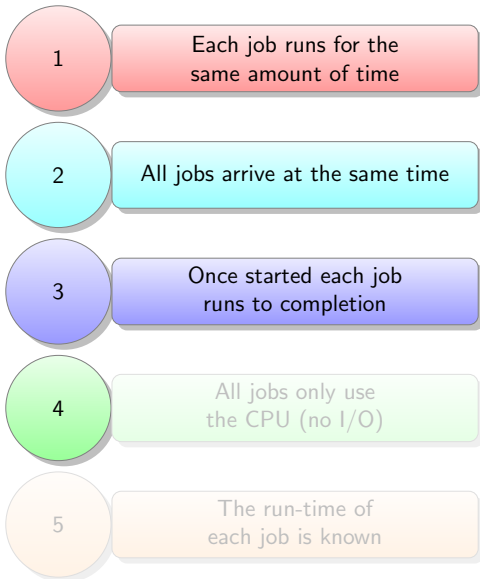
Workload Assumptions



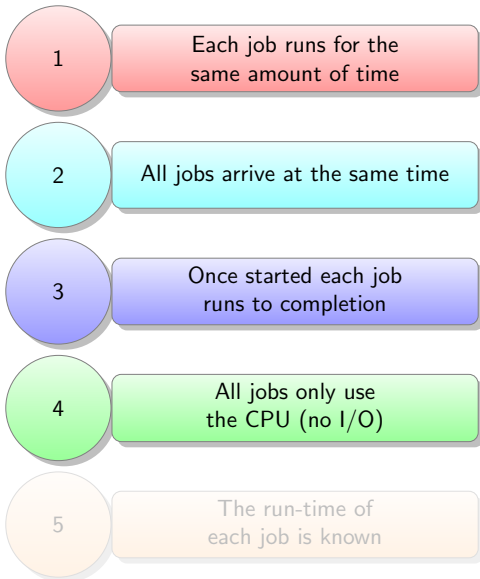
Workload Assumptions



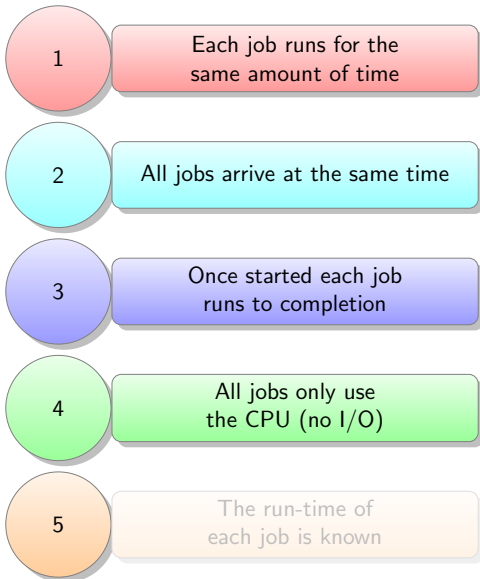
Workload Assumptions



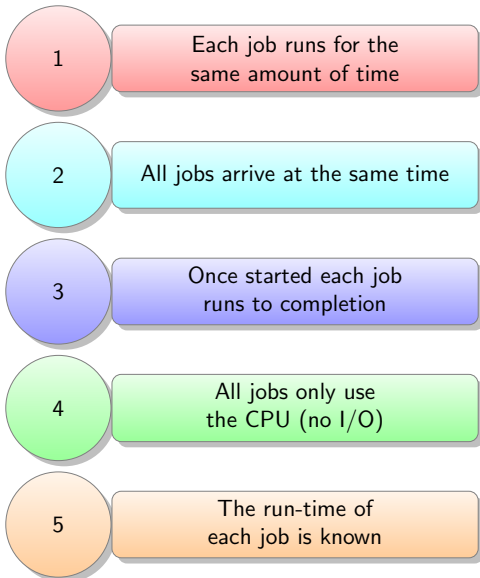
Workload Assumptions



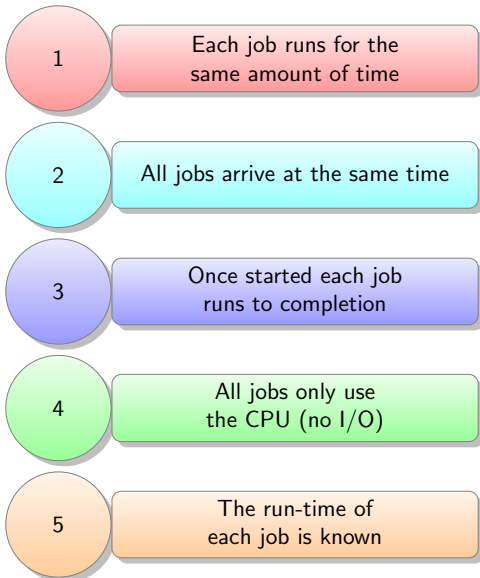
Workload Assumptions



Workload Assumptions



Workload Assumptions



turnaround time

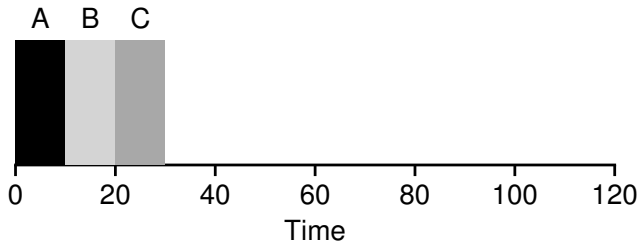
The turnaround time of a job is defined as the **time at which the job completes** minus **the time at which the job arrived** in the system.

$$T_{turnaround} = T_{completion} - T_{arrival}$$

First In, First Out First Come, First Served

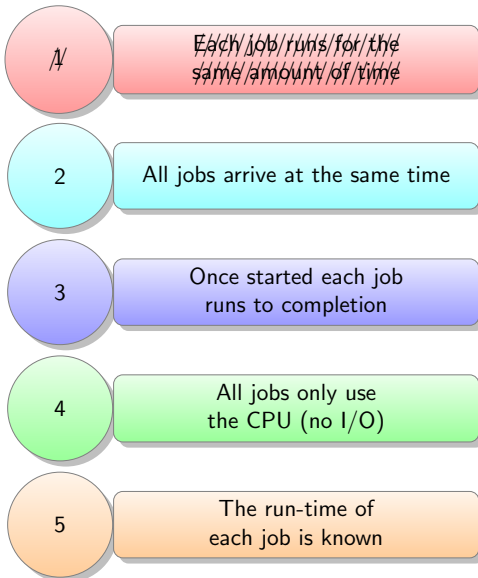
FIFO / FCFS

- ▶ A, B, and C, arrive at roughly the same time
- ▶ $T_{arrival} = 0$



Classwork

What will the **average turnaround time** be for these jobs?

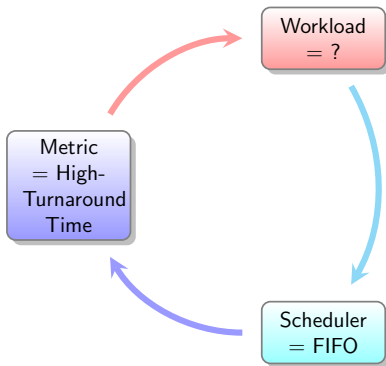


How does FIFO perform now?

CW-1

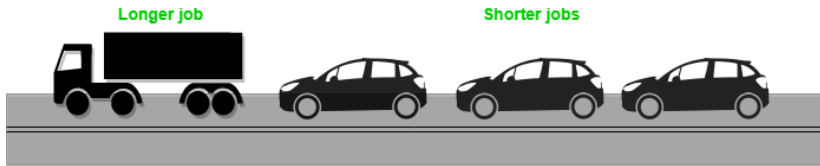
Classwork

Construct a **workload** to make FIFO perform poorly



average turnaround time?

The Convoy Effect



New Scheduler

How can we develop a better algorithm to deal with our new reality of jobs that run for different amounts of time?

Shortest Job First

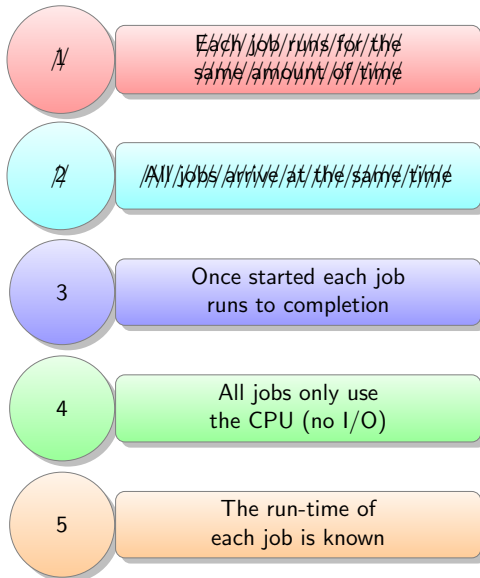
SJF

Simple Approach

- ▶ Run the shortest job first, then the next shortest, and so on.
 - ▶ Idea borrowed from operations research
-
- ▶ Redo the last FIFO workload with SJF
 - ▶ What is the turn around time now?
 - ▶ What is the improvement?

Point-to-Ponder

Is SJF an **optimal** scheduling algorithm? How?



How does SJF perform now?

CW-2

Classwork

Construct a **workload** to make SJF perform poorly

Need for a new scheduler

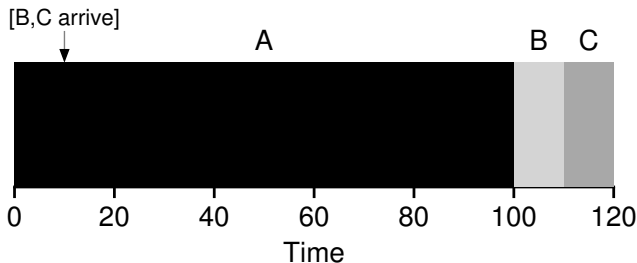
- ▶ Again what is the average turnaround time now?

How does SJF perform now?

CW-2

Classwork

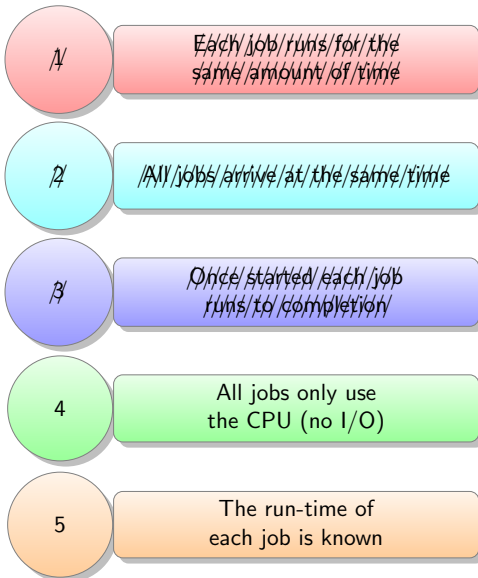
Construct a **workload** to make SJF perform poorly



The Convoy Effect

Need for a new scheduler

- Again what is the average turnaround time now?



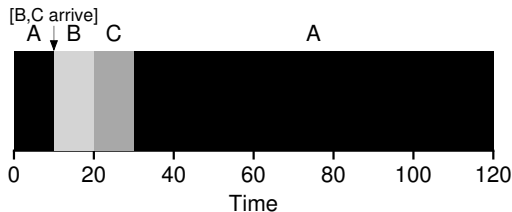
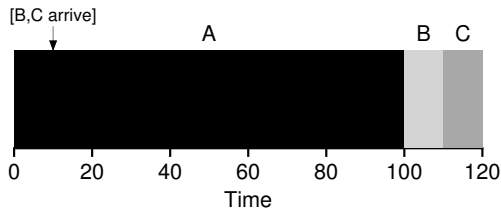
Preemptive Scheduling

The scheduler can perform a **context switch**, stopping one running process temporarily and resuming (or starting) another.

- ▶ Using the mechanisms we discussed earlier
- ▶ The timer interrupt
- ▶ And context switching

Shortest Time-to-Completion First
Preemptive Shortest Job First
Shortest Remaining Time First

STCF/PSJF/SRTF



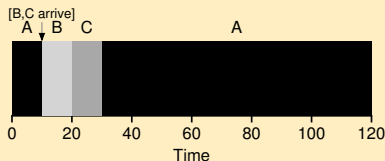
Compare avg. turnaround time

- ▶ Sometimes we care about when a job starts
- ▶ Instead of when it finishes

$$T_{\text{response}} = T_{\text{firstrun}} - T_{\text{arrival}}$$

- ▶ What does T_{response} capture?

CW-3

Compute Avg. T_{response} for Preemptive SJF

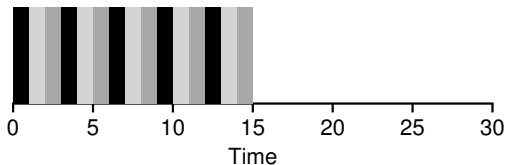
- ▶ How can we build a scheduler that is sensitive to response time?

Round Robin

Time-slicing

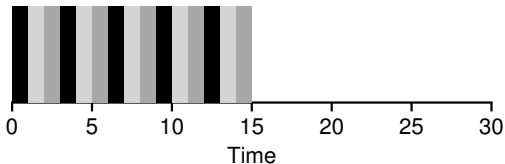
RR

- ▶ Instead of running jobs to completion, RR runs a job for a time slice (sometimes called a **scheduling quantum**) and
- ▶ Then switches to the next job in the run queue.
- ▶ It repeatedly does so until the jobs are finished.

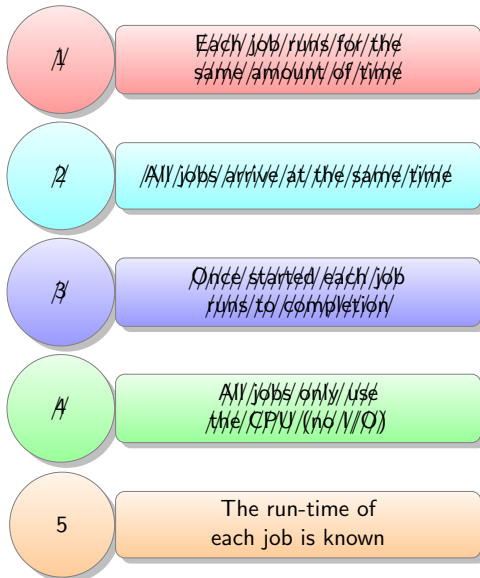


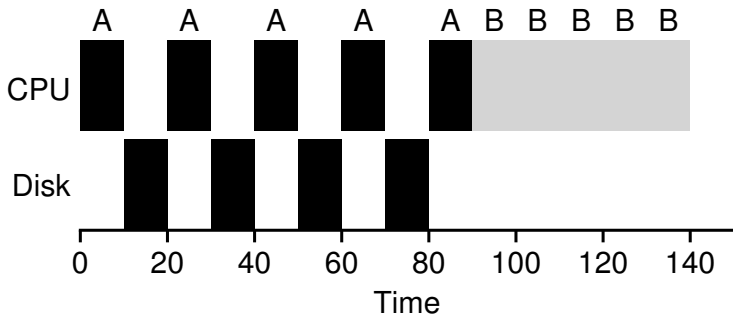
- ▶ Avg. Response Time?
- ▶ Compare with FIFO
- ▶ What about avg. turn-around time?
- ▶ How to choose the time-slice?

- ▶ Instead of running jobs to completion, RR runs a job for a time slice (sometimes called a **scheduling quantum**) and
- ▶ Then switches to the next job in the run queue.
- ▶ It repeatedly does so until the jobs are finished.

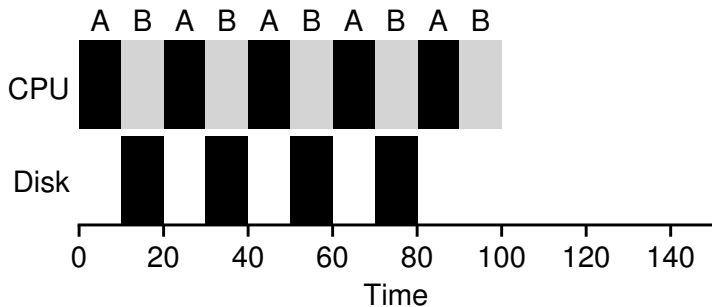


- ▶ Avg. Response Time?
- ▶ Compare with FIFO
- ▶ What about avg. turn-around time?
- ▶ How to choose the time-slice?





I/O Aware (Overlap)





- ▶ Smarter Scheduling
- ▶ Multi-level feedback queue