Operating Systems

Chapter 9 Uniprocessor Scheduling

Agenda

- 9.1 Type of Processor Scheduling
- 9.2 Scheduling Algorithms
- 9.3 Summary

9.1 Type of Processor Scheduling

- 9.1.0 Overview
- 9.1.1 Long-term Scheduling
- 9.1.2 Medium-term Scheduling
- 9.1.3 Short-term Scheduling

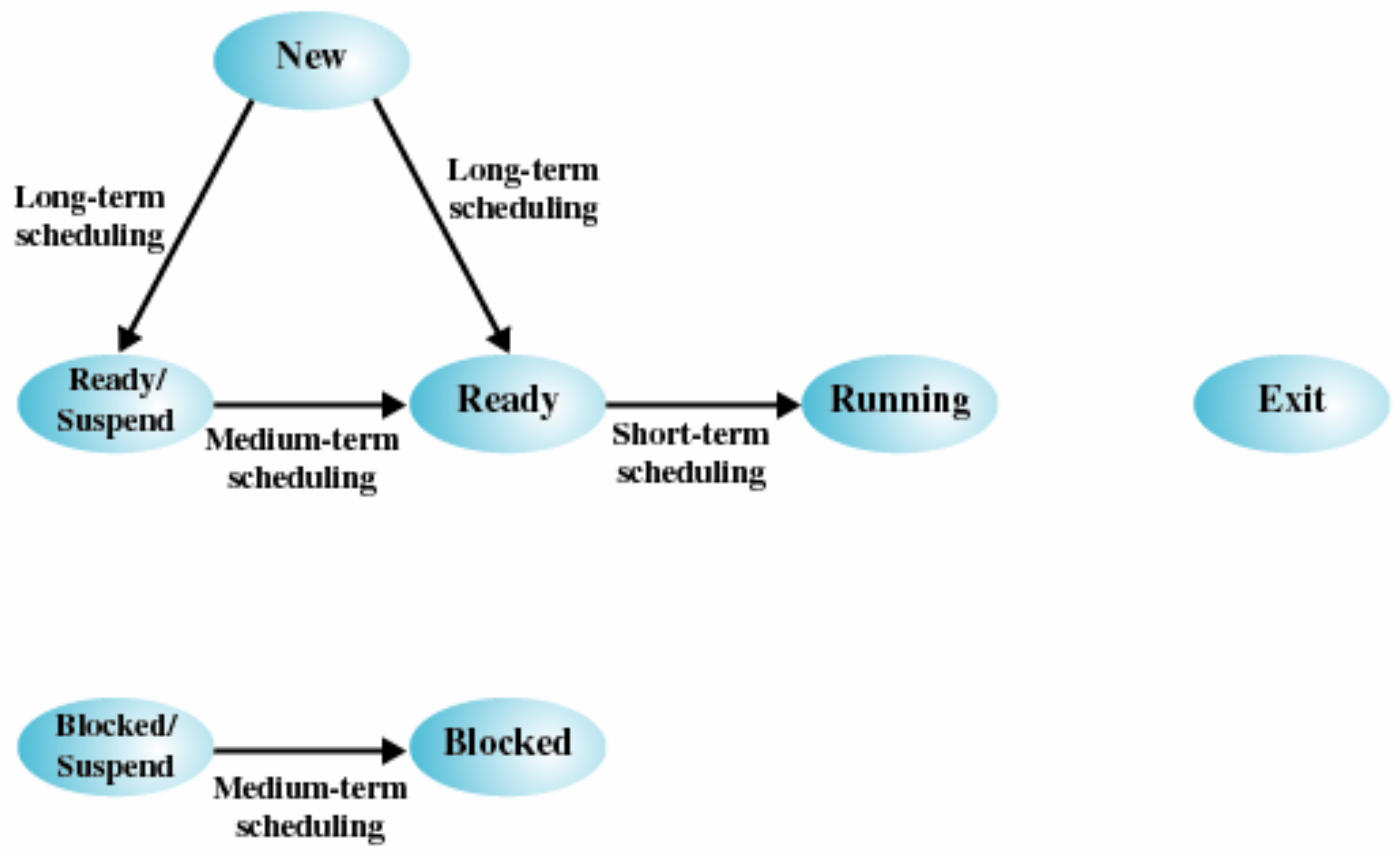
Aim of Scheduling(调度目标)

- Assign processes to be executed by the processor(s)(处理器分配)
- Response time (响应时间)
- Throughput (吞吐率)
- Processor efficiency (处理器效率)

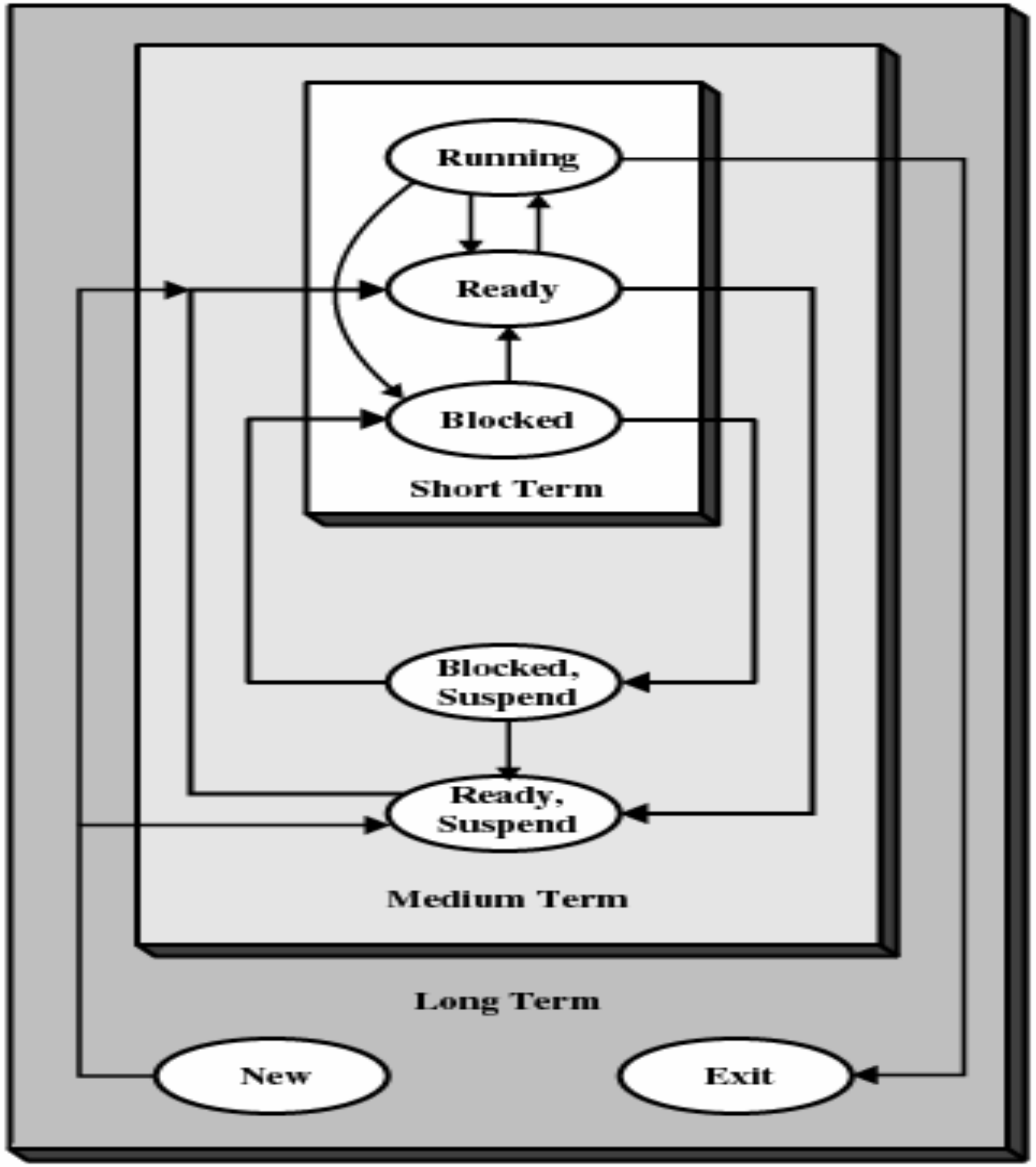
Types of Scheduling

Long-term scheduling	The decision to add to the pool of processes to be executed
Medium-term scheduling	The decision to add to the number of processes that are partially or fully in main memory
Short-term scheduling	The decision as to which available process will be executed by the processor
I/O scheduling	The decision as to which process's pending I/O request shall be handled by an available I/O device

Scheduling and Process State Transitions(调度与进程状态转换)



Levels of Scheduling(调度层次)



单选题 1分

设置

The type of scheduling that involves the decision to add a process to those that are at least partially in main memory and therefore available for execution is referred to as:

A I/O scheduling

B Long-term scheduling

C Medium-term scheduling

D None of the above

提交

9.1 Type of Processor Scheduling

- 9.1.0 Overview
- 9.1.1 Long-term Scheduling
- 9.1.2 Medium-term Scheduling
- 9.1.3 Short-term Scheduling

Long-Term Scheduling (长程调度)

- Determines which programs are admitted to the system for processing
- Controls the degree of multiprogramming
- More processes, smaller percentage of time each process is executed

9.1 Type of Processor Scheduling

- 9.1.0 Overview
- 9.1.1 Long-term Scheduling
- 9.1.2 Medium-term Scheduling
- 9.1.3 Short-term Scheduling

Medium-Term Scheduling (中程调度)

- Part of the swapping function
- Based on the need to manage the degree of multiprogramming

Typically, the swapping-in function for processes is based on the need to manage:

A Virtual memory

B Process priorities

C The degree of multiprogramming

D None of the above

提交

9.1 Type of Processor Scheduling

- 9.1.0 Overview
- 9.1.1 Long-term Scheduling
- 9.1.2 Medium-term Scheduling
- <u>9.1.3 Short-term Scheduling</u>

Short-Term Scheduling (短程调度)

- Known as the dispatcher(分派器)
- Executes most frequently
- Invoked when an event occurs
  - Clock interrupts(时钟中断)
  - I/O interrupts(I/O中断)
  - Operating system calls(操作系统调用)
  - Signals(信号)

单选题 1分

设置

In terms of frequency of execution, the short-term scheduler is usually the one that executes:

A Most frequently

B About the same as the other schedulers

C Least frequently

D None of the above

提交

Agenda

- 9.1 Type of Processor Scheduling
- 9.2 Scheduling Algorithms
- 9.3 Summary

9.2 Scheduling Algorithms

- <u>9.2.1 Short-term Criteria</u>
- 9.2.2 The Use of Priorities
- 9.2.3 Alternative of Scheduling Policies
- 9.2.4 Fair-Share Scheduling

Short-Tem Scheduling Criteria(短程调度准则)

- User-oriented(面向用户)
  - Behavior of OS as perceived by the user (用户感知到的系统行为)
  - E.g. Response Time(响应时间):Elapsed time between the submission of a request until there is output.
- System-oriented(面向系统)
  - Effective and efficient utilization of the processor(处理器的使用效率)

Short-Term Scheduling Criteria

- Performance-related(性能相关)
  - Quantitative 定量的
    - Measurable such as
      - response time(响应时间)
      - throughput(吞吐率)
- Not directly performance related(与性能无直接关系)
  - Qualitative 定性的
    - Unmeasurable such as
      - predictability(可预测性)

Scheduling Criteria

User Oriented, Performance Related

Turnaround time This is the interval of time between the submission of a process and its completion. Includes actual execution time plus time spent waiting for resources, including the processor. This is an appropriate measure for a batch job.

Response time For an interactive process, this is the time from the submission of a request until the response begins to be received. Often a process can begin producing some output to the user while continuing to process the request. Thus, this is a better measure than turnaround time from the user's point of view. The scheduling discipline should attempt to achieve low response time and to maximize the number of interactive users receiving acceptable response time.

Deadlines When process completion deadlines can be specified, the scheduling discipline should subordinate other goals to that of maximizing the percentage of deadlines met.

User Oriented, Other

Predictability A given job should run in about the same amount of time and at about the same cost regardless of the load on the system. A wide variation in response time or turnaround time is distracting to users. It may signal a wide swing in system workloads or the need for system tuning to cure instabilities.

Scheduling Criteria (Cont.)

吞吐量

System Oriented, Performance Related

Throughput The scheduling policy should attempt to maximize the number of processes completed per unit of time. This is a measure of how much work is being performed. This clearly depends on the average length of a process but is also influenced by the scheduling policy, which may affect utilization.

Processor utilization This is the percentage of time that the processor is busy. For an expensive shared system, this is a significant criterion. In single-user systems and in some other systems, such as real-time systems, this criterion is less important than some of the others.

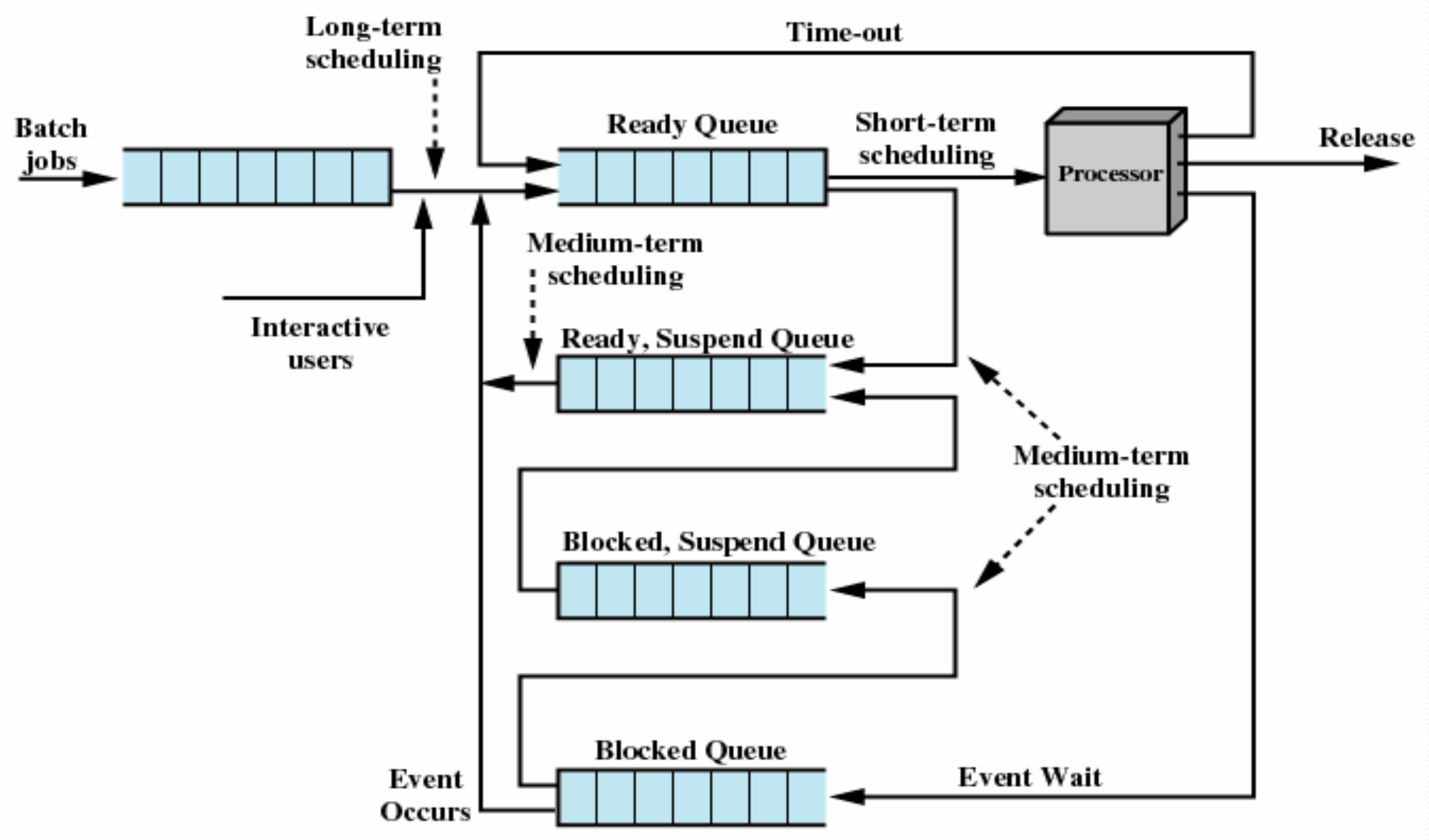
System Oriented, Other

Fairness In the absence of guidance from the user or other system-supplied guidance, processes should be treated the same, and no process should suffer starvation.

Enforcing priorities When processes are assigned priorities, the scheduling policy should favor higher-priority processes.

Balancing resources The scheduling policy should keep the resources of the system busy. Processes that will underutilize stressed resources should be favored. This criterion also involves medium-term and long-term scheduling.

Queuing Diagram for Scheduling调度队列图



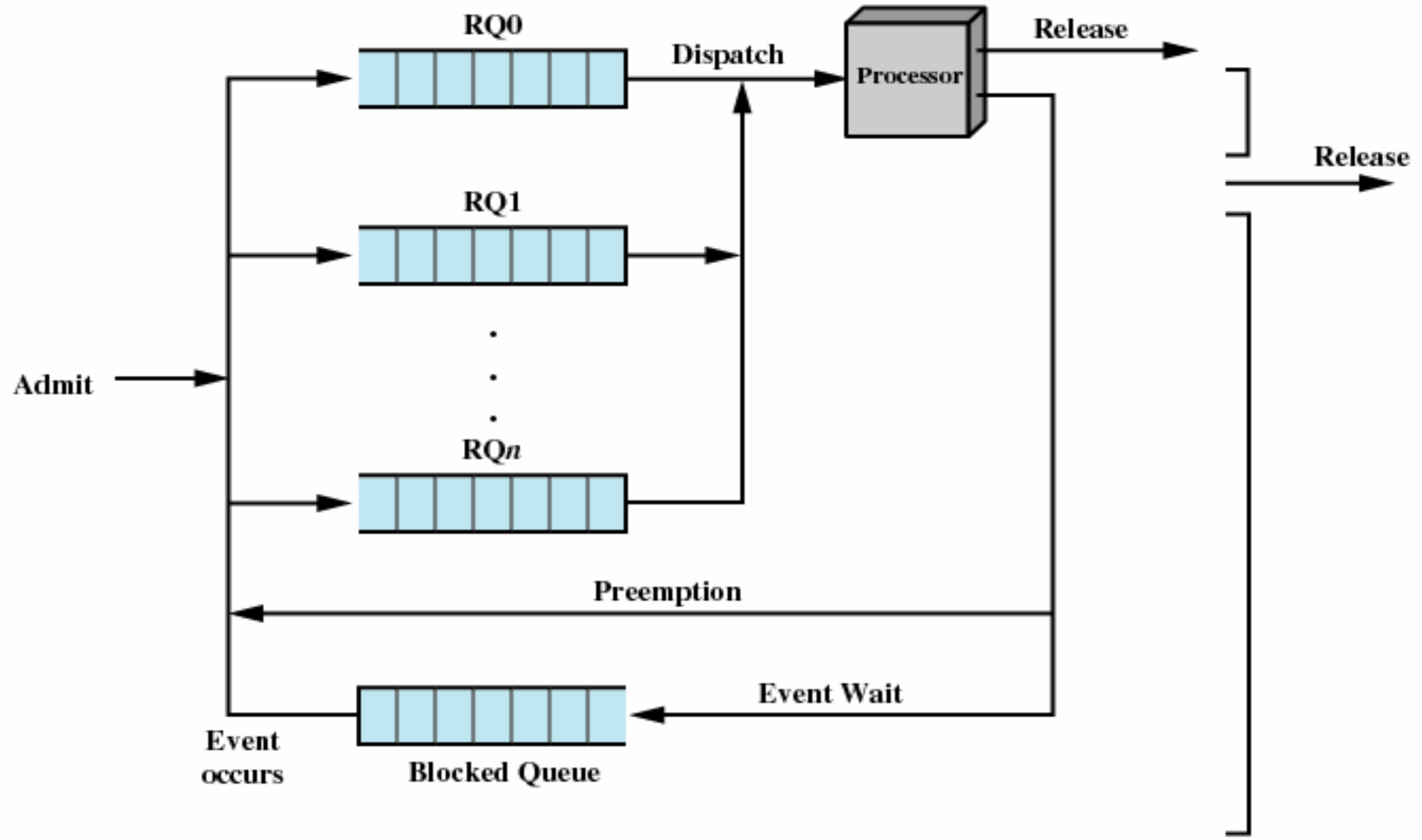
9.2 Scheduling Algorithms

- 9.2.1 Short-term Criteria
- 9.2.2 The Use of Priorities
- 9.2.3 Alternative of Scheduling Policies
- 9.2.4 Fair-Share Scheduling

Priorities(优先级)

- Scheduler will always choose a process of higher priority over one of lower priority
- Have multiple ready queues to represent each level of priority
- Lower-priority may suffer starvation
  - Allow a process to change its priority based on its age or execution history

Priorities Queuing



单选题 1分

设置

A typical way to overcome starvation of lower-priority processes in a priority-based scheduling system is to:

A Change a process priority with its age

B Change a process priority randomly

C Round-robin cycling of processes in a priority queue

D All of the above

提交

9.2 Scheduling Algorithms

- 9.2.1 Short-term Criteria
- 9.2.2 The Use of Priorities
- 9.2.3 Alternative of Scheduling Policies
- 9.2.4 Fair-Share Scheduling

Terms

- Response time(响应时间)
- Turnaround time(周转时间)
- Normalized turnaround time(归一化周转时间，周转时间/服务时间)
- Throughput (吞吐率)
- Predictability(可预测性)
- Selection function(选择函数，确定在就绪进程中选择哪一个进程在下一次执行)
- Decision mode(决策模式，选择函数被执行瞬间的处理方式)
  - Preemptive (抢占，当前正在执行的进程可能被操作系统中断，并转移到就绪态)
  - Nonpreemptive(非抢占，一旦处于运行态就不断执行直到终止或者被阻塞)

单选题 1分

设置

In terms of the queuing model, the total time that a process spends in a system (waiting time plus service time) is called:

A Finish time (FT)

B Normalized turnaround time (TAT)

C Turnaround or residence time (TAT)

D None of the above

提交

Which of the following scheduling policies allow the O/S to interrupt the currently running process and move it to the Ready state?

A Preemptive

B First-come-first-served

C Non-Preemptive

D None of the above

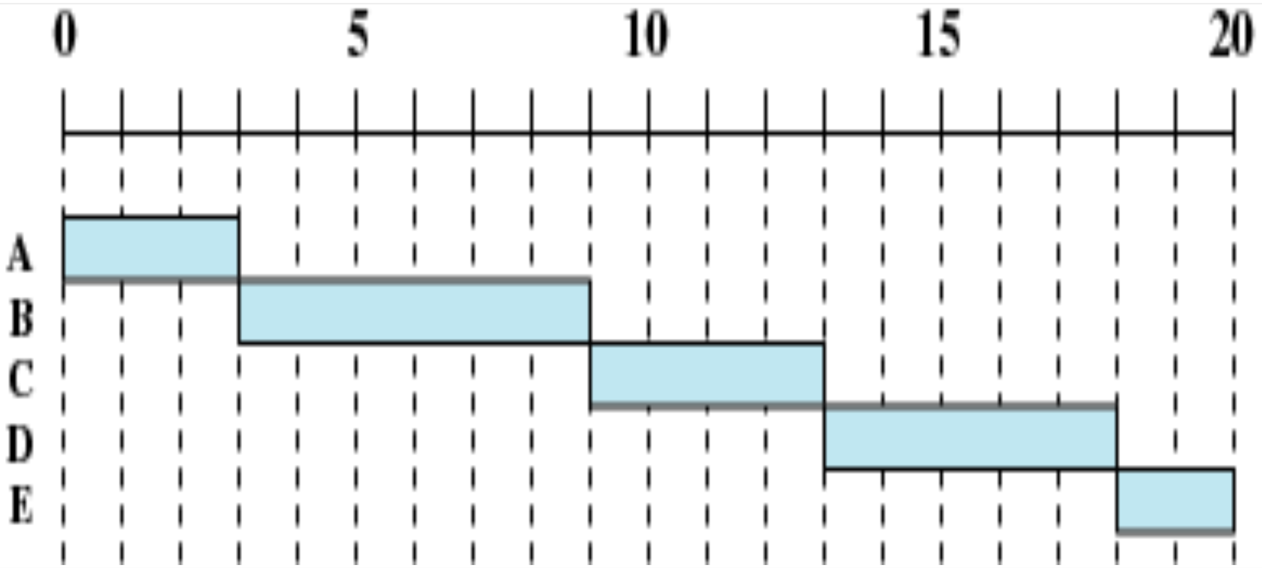
提交

Process Scheduling Example

Process	Arrival Time	Service Time
A	0	3
B	2	6
C	4	4
D	6	5
E	8	2

First-Come-First-Served (FCFS，先来先服务)

Process	Arrival Time	Service Time
A	0	3
B	2	6
C	4	4
D	6	5
E	8	2



- Decision mode: ***Nonpreemptive***
- Dispatch time: the current process ceases(终止) to execute
- Method: The oldest process in the Ready queue is selected

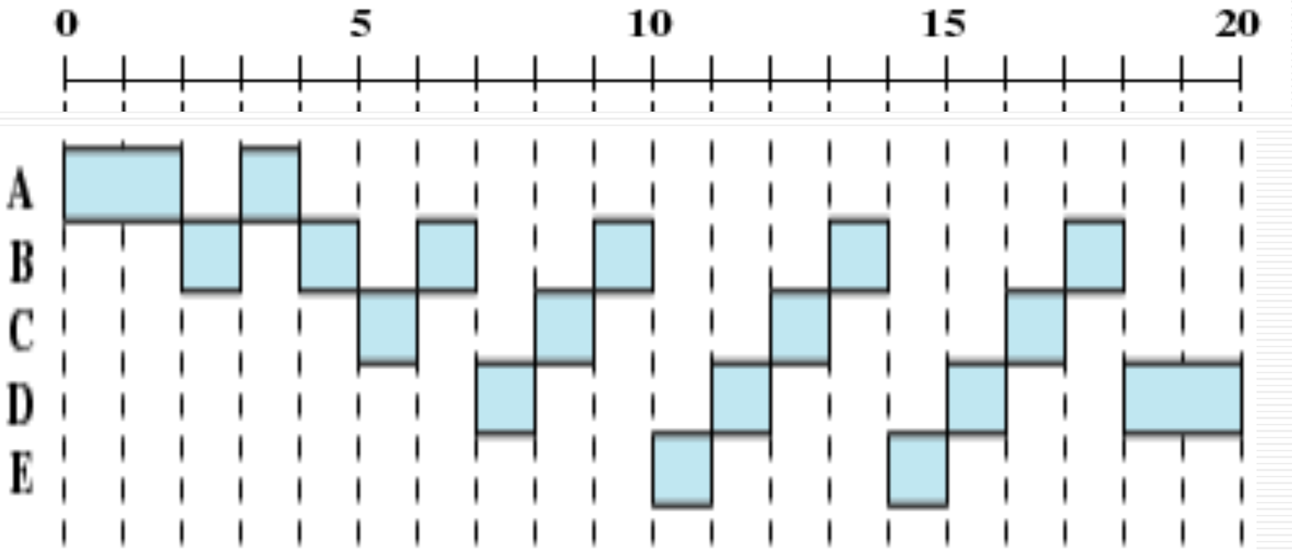
First-Come-First-Served (FCFS)

- Disadvantageous to short process
  - A short process may have to wait a very long time before it can execute
- Favors CPU-bound processes(偏向于受处理器限制的进程)
  - I/O processes have to wait until CPU-bound process completes

Round-Robin (RR，轮转)

Process	Arrival Time	Service Time
A	0	3
B	2	6
C	4	4
D	6	5
E	8	2

RR， q＝1

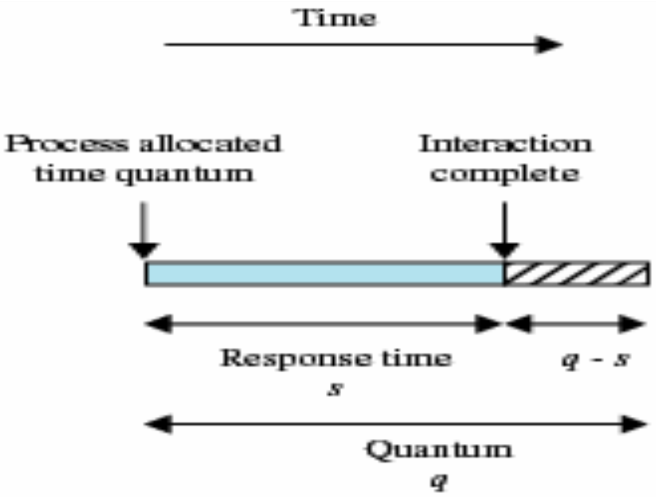


- Decision mode: ***preemption***
- Dispatch time: period q based on a clock
- Method： select next process based on FCFS

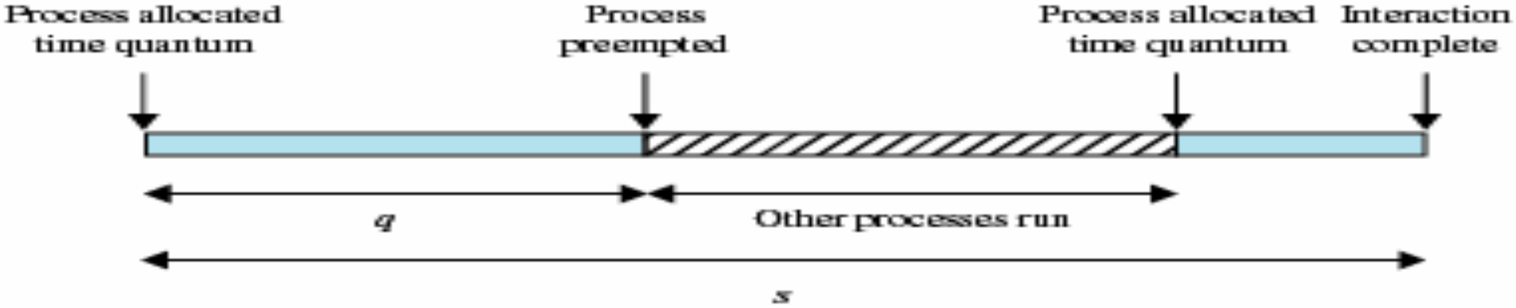
Round-Robin

- Clock interrupt is generated at periodic intervals
- When an interrupt occurs, the currently running process is placed in the read queue
  - Next ready job is selected based on FCFS
- Known as time slicing (时间片)
- Disadvantageous to I/O-bound processes（不利于io频繁进程）

Effect of Size of Preemption Time Quantum (时间片大小的影响)

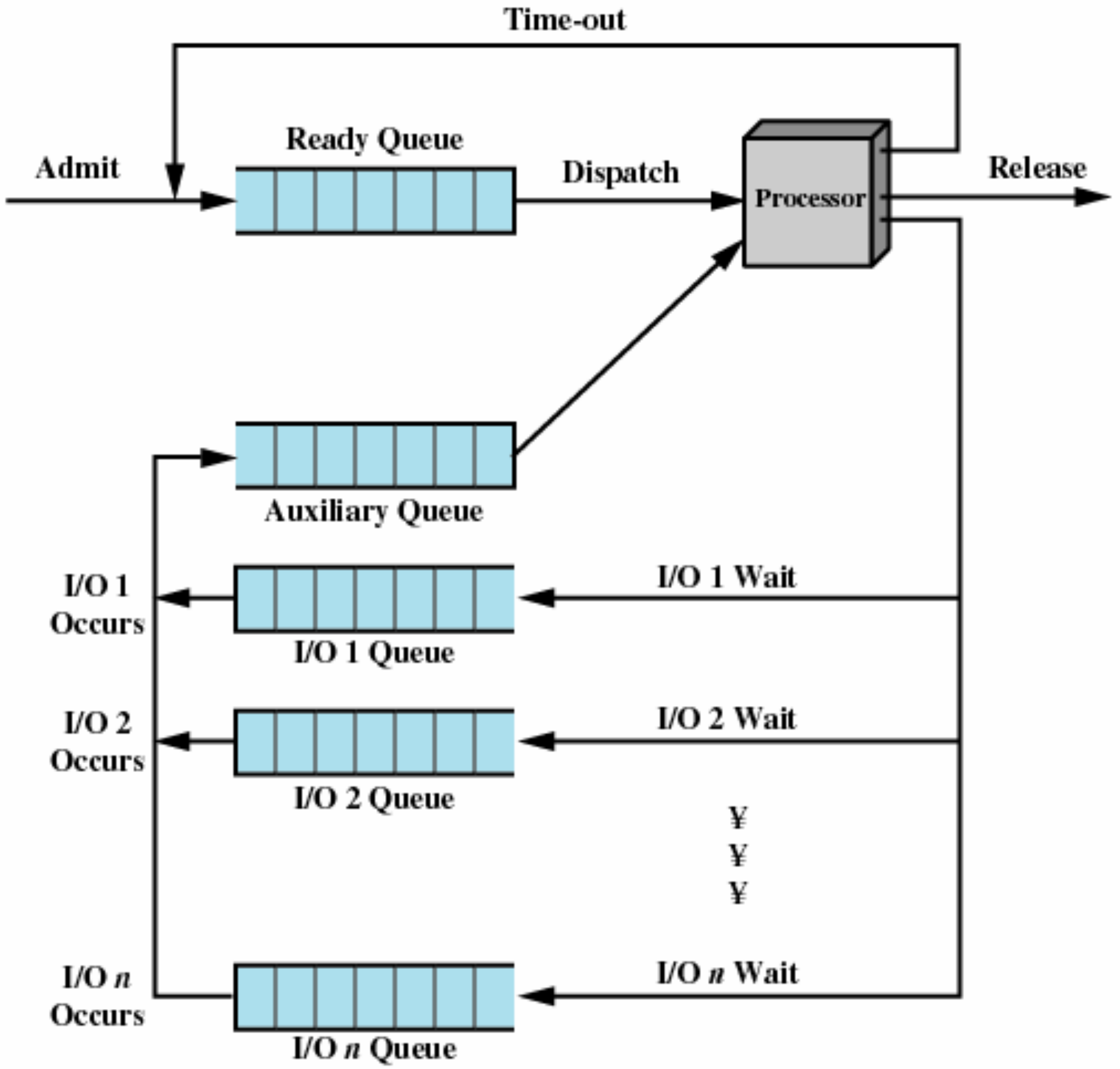


(a) Time quantum greater than typical interaction



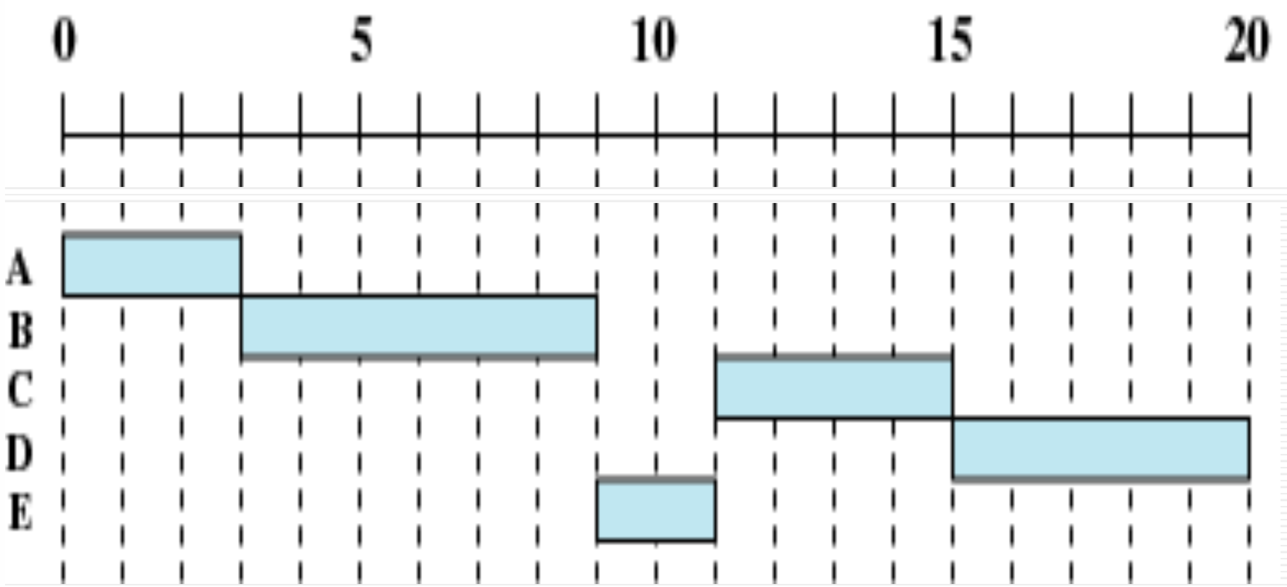
(b) Time quantum less than typical interaction

Queuing Diagram for virtual Round-Robin Scheduler (VRR，虚拟轮转法)



Shortest Process Next (SPN，最短进程优先)

Process	Arrival Time	Service Time
A	0	3
B	2	6
C	4	4
D	6	5
E	8	2



- Decision mode: ***Nonpreemptive***
- Dispatch time: the current process ceases to execute
- Method: Process with shortest expected processing time is selected next

Shortest Process Next

- Predictability of longer processes is reduced(长进程的可预测性降低了)
- Possibility of starvation(饥饿) for longer processes

单选题 1分

设置

One difficulty with the Shortest Process Next (SPN) scheduling technique is: (1.0分)

A The need to know or estimate required processing times for each process

B The starvation of longer processes

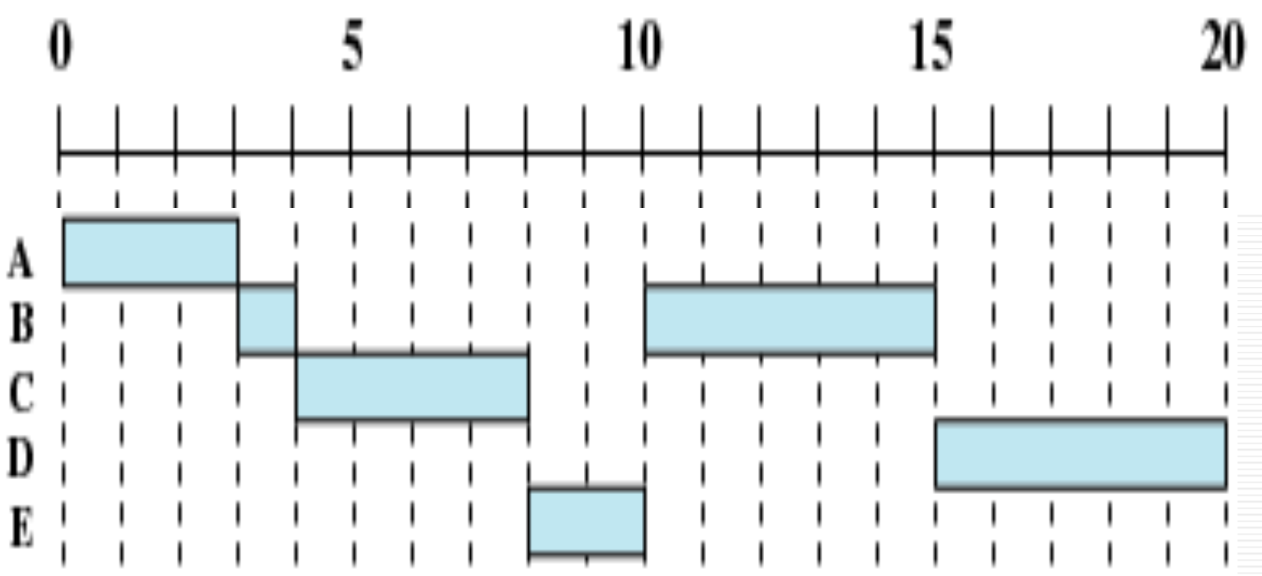
C The lack of preemption

D All of the above

提交

Shortest Remaining Time (SRT，最短剩余时间)

Process	Arrival Time	Service Time
A	0	3
B	2	6
C	4	4
D	6	5
E	8	2



- Decision mode: ***Preemptive*** version of shortest process next
- Dispatch time: decision is made when **a new process arrives**
- Method: Process with shortest remaining time is selected next

单选题 1分

设置

One difficulty with the Shortest Remaining Time (SRT) scheduling technique is:

A The starvation of shorter processes

B The lack of preemption

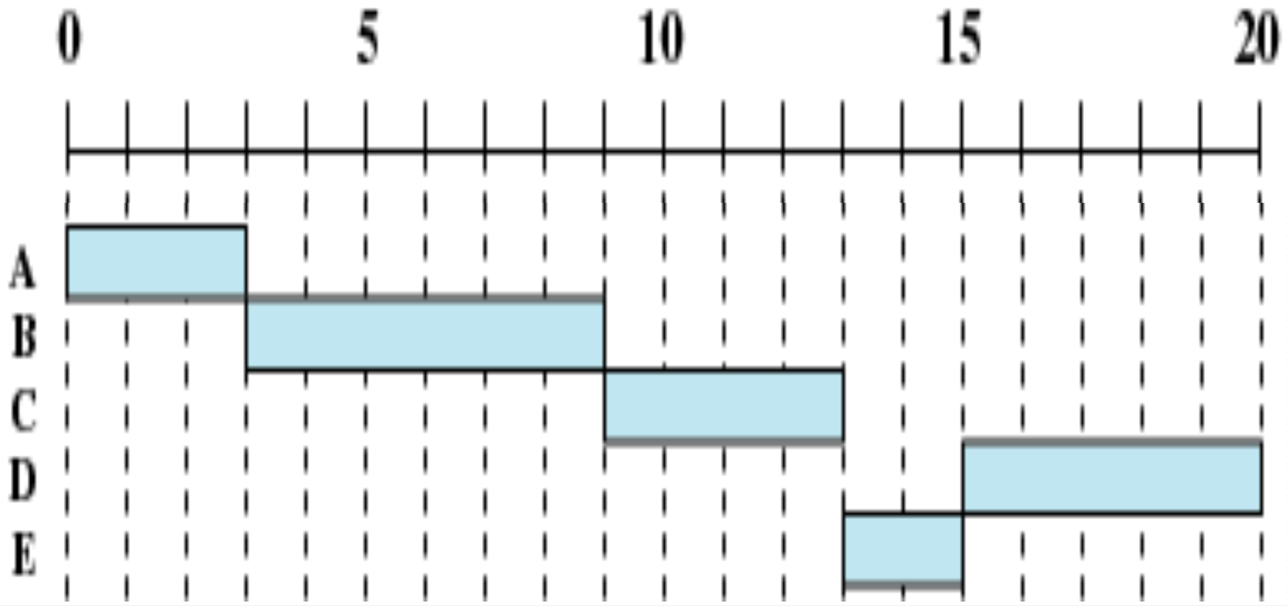
C The need to know or estimate required processing times for each process

D All of the above

提交

Highest Response Ratio Next (HRRN，最高响应比优先)

Process	Arrival Time	Service Time
A	0	3
B	2	6
C	4	4
D	6	5
E	8	2

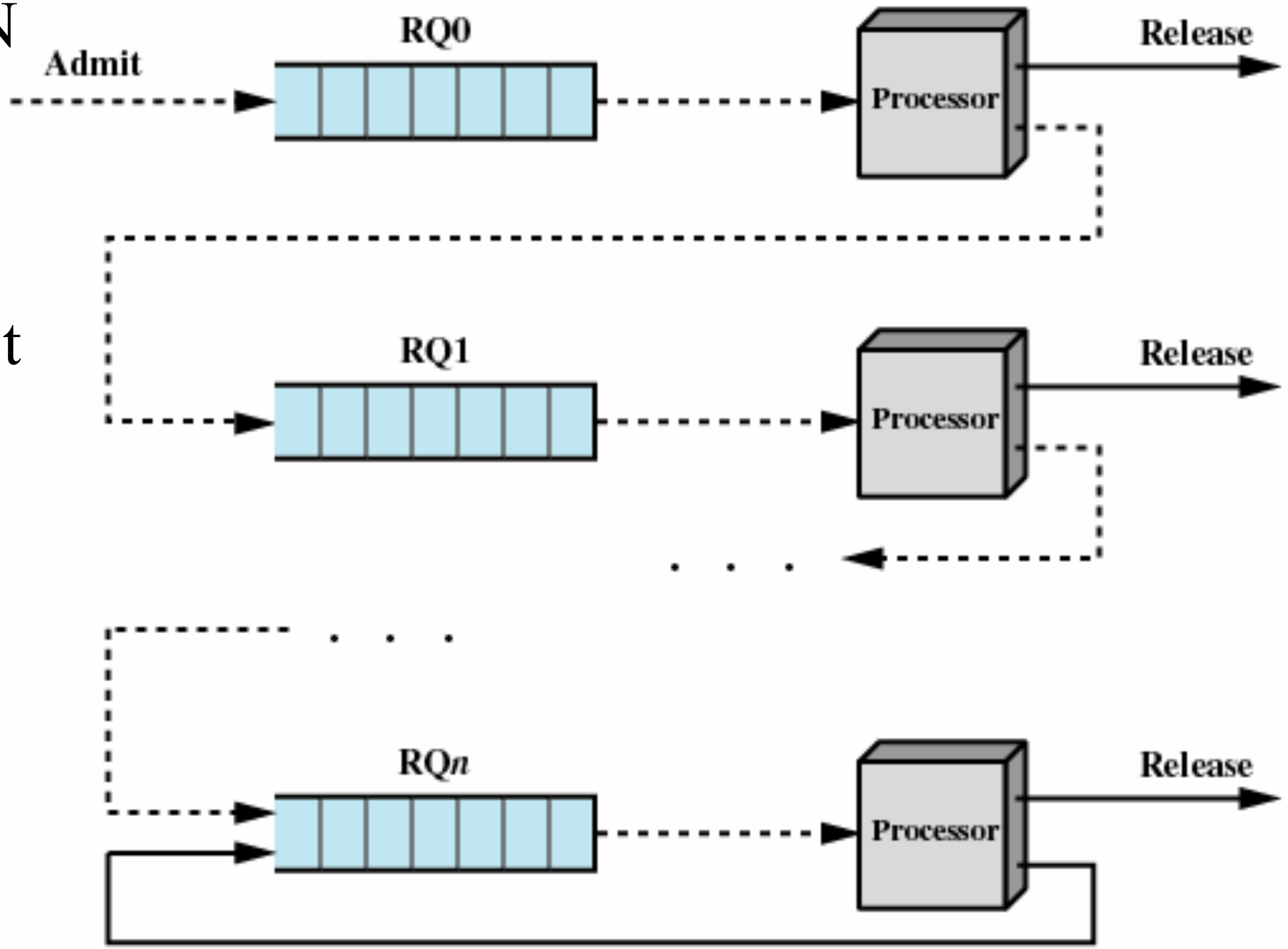


- Decision mode: ***Nonpreemptive***
- Dispatch time: the current process ceases to execute
- Method: Choose next process with the greatest response ratio

$$\mathbf{R = \frac{time\ spent\ waiting + expected\ service\ time}{expected\ service\ time}}$$

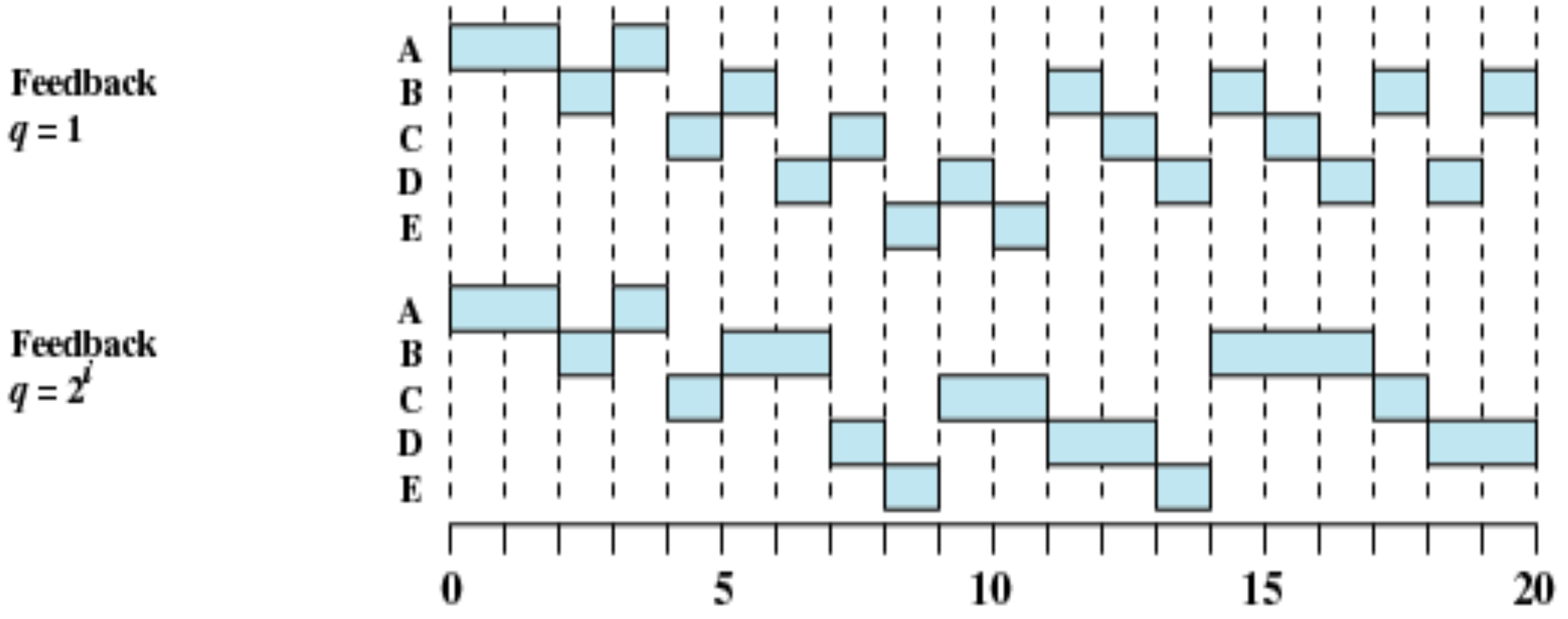
Feedback (反馈)

- SPN/SRT/HRRN need to know process service time, while Feedback doesn’t
- ***Preemptive***
- Penalize(惩罚) jobs that have been running longer



Feedback

Process	Arrival Time	Service Time
A	0	3
B	2	6
C	4	4
D	6	5
E	8	2



Characteristics of Various Scheduling Policies

	Selection Function	Decision Mode	Throughput	Response Time	Overhead	Effect on Processes	Starvation
FCFS	$\max[w]$	Nonpreemptive	Not emphasized	May be high, especially if there is a large variance in process execution times	Minimum	Penalizes short processes; penalizes I/O bound processes	No
Round Robin	constant	Preemptive (at time quantum)	May be low if quantum is too small	Provides good response time for short processes	Minimum	Fair treatment	No
SPN	$\min[s]$	Nonpreemptive	High	Provides good response time for short processes	Can be high	Penalizes long processes	Possible
SRT	$\min[s - e]$	Preemptive (at arrival)	High	Provides good response time	Can be high	Penalizes long processes	Possible
HRRN	$\max\left(\frac{w+s}{s}\right)$	Nonpreemptive	High	Provides good response time	Can be high	Good balance	No
Feedback	(see text)	Preemptive (at time quantum)	Not emphasized	Not emphasized	Can be high	May favor I/O bound processes	Possible

w = time spent waiting
e = time spent in execution so far
s = total service time required by the process, including e

A Comparison of Scheduling policies

进程	A	B	C	D	E	
到达时间	0	2	4	6	8	
服务时间（T_s）	3	6	4	5	2	平均值
			FCFS			
完成时间	3	9	13	18	20	
服务时间（T_s）	3	7	9	12	12	8.60
T_r/T_s	1.00	1.17	2.25	2.40	6.00	2.56
			RR $q = 1$			
完成时间	4	18	17	20	15	
服务时间（T_s）	4	16	13	14	7	10.80
T_r/T_s	1.33	2.67	3.25	2.80	3.50	2.71
			RR $q = 4$			
完成时间	3	17	11	20	19	
服务时间（T_s）	3	15	7	14	11	10.00
T_r/T_s	1.00	2.5	1.75	2.80	5.50	2.71
			SPN			
完成时间	3	9	15	20	11	
服务时间（T_s）	3	7	11	14	3	7.60
T_r/T_s	1.00	1.17	2.75	2.80	1.50	1.84
			SRT			
完成时间	3	15	8	20	10	
服务时间（T_r）	3	13	4	14	2	7.20
T_r/T_s	1.00	2.17	1.00	2.80	1.00	1.59
			HRRN			
完成时间	3	9	13	20	15	
服务时间（T_r）	3	7	9	14	7	8.00
T_r/T_s	1.00	1.17	2.25	2.80	3.5	2.14
			FB $q = 1$			
完成时间	4	20	16	19	11	
服务时间（T_r）	4	18	12	13	3	10.00
T_r/T_s	1.33	3.00	3.00	2.60	1.5	2.29
			FB $q = 2^i$			
完成时间	4	17	18	20	14	
服务时间（T_r）	4	15	14	14	6	10.60
T_r/T_s	1.33	2.50	3.50	2.80	3.00	2.63

单选题 1分

设置

Which of the following scheduling policies require prior knowledge or estimation of process length:

A Highest Response Ratio Next (HRRN)

B Shortest Remaining Time (SRT)

C Shortest Process Next (SPN)

D All of the above

提交

单选题 1分

设置

The strategy that schedules processes based on their group affiliation is generally referred to as: (1.0分)

A Simulation modeling

B Queuing analysis

C Fair share scheduling

D All of the above

提交

9.2 Scheduling Algorithms

- 9.2.1 Short-term Criteria
- 9.2.2 The Use of Priorities
- 9.2.3 Alternative of Scheduling Policies
- <u>9.2.4 Fair-Share Scheduling</u>

Fair-Share Scheduling(公平共享调度)

- User's application runs as a collection of processes (threads)
- User is concerned about the performance of the application
- Need to make scheduling decisions based on process sets

Fair-Share Scheduling(公平共享调度)

- Scheduling is done on the basis of
  - under-lying priority of the process
  - recent processor usage of the process
  - recent processor usage of the group
- The higher the numerical value of the priority, the lower the priority. The following formulas apply for process j in group k:

$$CPU_j(i) = \frac{CPU_j(i-1)}{2}$$

$$GCPU_k(i) = \frac{GCPU_k(i-1)}{2}$$

$$P_j(i) = Base_j + \frac{CPU_j(i)}{2} + \frac{GCPU_k(i)}{4 \times W_k}$$

Example of Fair share Scheduler

Time	Process A			Process B			Process C		
	Priority	Process CPU count	Group CPU count	Priority	Process CPU count	Group CPU count	Priority	Process CPU count	Group CPU count
0	60	0 1 2 . . 60	0 1 2 . . 60	60	0	0	60	0	0
1	90	30	30	60	0 1 2 . . 60	0 1 2 . . 60	60	0	0 1 2 . . 60
2	74	15 16 17 . . 75	15 16 17 . . 75	90	30	30	75	0	30
3	96	37	37	74	15	15 16 17 . . 75	67	0 1 2 . . 60	15 16 17 . . 75
4	78	18 19 20 . . 78	18 19 20 . . 78	81	7	37	93	30	37
5	98	39	39	70	3	18	76	15	18

Group 1 (Process A) — Group 2 (Process B, Process C)

Colored rectangle represents executing process

多选题 1分

设置

The strategy that schedules processes based on their group affiliation is generally referred to as:

A Simulation modeling

B Queuing analysis

C Fair share scheduling

D All of the above

提交

Agenda

- 9.1 Type of Processor Scheduling
- 9.2 Scheduling Algorithms
- 9.3 Summary