TEST2a_paper_Processes-SchedulingComputations

TEST2--PROCESSES & CPU SCHEDULING YOUR NAME: KEY

===

PART I: Essay Questions (max 27 points)

===

E.1. Essay (10 pts)

In CPU scheduling, name and describe the different states that a process can exist at various times. Describe what happens to cause the process to ENTER the state, and to EXIT the state.

Provide your answer as a list of states numbered (a), (b), ... in the format below:

(a) running state. has CPU
ENTER when ... chosen by scheduler.
EXIT when ... wait for I/O, interrupt, preempted. Completed

(b) ready state - can be scheduled.
ENTER: arrival; end of wait.
EXIT: scheduled for CPU.

(c) wait ~~state~~: I/O request; interrupt; preemption. [ENTER]
EXIT: Completion of I/O, interrupt;

(d). Completed. Enter when ~~used~~ no longer needs CPU.

(e) new

E.2. Essay (6 points)

Ordinarily a fork() system call is followed by the exec() system call.

(a) Explain what the fork() does; Creates a child process for parent.

(b) Explain what the exec() does. Loads child process with prog.

(c) What do the child and parent process share? The resources of the parent.

E.3. Essay (4 points)

Define response time and turnaround time. Answer in the format below.

(a) Define Response time - time between arrival in ready queue to first access to CPU.

(b) Define turnaround time - time between arrival and completion.

E.4. Essay (7 points)

(a) In which CPU scheduling algorithm is starvation a problem.

(b) What causes starvation?

(c) Explain how aging can be used to prevent it.

==

PART II: Scheduling Calculations (min 29 point, max 38 points)

==

ALL THE PROBLEMS USE THIS SET OF PROCESSES.
Suppose that the following processes arrive for execution at the times indicated. In order a for a process to complete, its total CPU time must be the runt time listed.

MANDATORY: PROBLEMS 1, 2, and 5; either 3 or 4 -- CIRCLE the one you pick. Others OPTIONAL.

Arrival Time:	1	0	2	
Process:	P1	P2	P3	
Run Time:	8	4	3	(total time needed)
Priority:	1	2	3	(only for priority scheduling, 3-HIGHEST)

C.1. (8 pts) FCFS SCHEDULING: Show schedule as a Gantt chart (4pts). Compute THROUGHPUT.

process:	P2	P1	P3				
time used:	4	8	3				

Show your THROUGHPUT calculations (4 pts):

tot time = 8+4+3 = 15

#completed processes = 3

Thruput = $\frac{3}{15}$

C.2. ROUND-ROBIN SCHEDULING. Quantum=3. Show the schedule as a Gantt chart (5pts).

process:	P2	P1	P3	P2	P1	P1		
time used:	3	3	3	~~3~~ 1	3	2		

C.3. SJF SCHEDULING: Show the schedule as a Gantt chart (4 pts). Compute AVERAGE TURNAROUND TIME.

process:	P2	P3	P1	P2	P3	P2	P1	
time used:	4	3	8	2	3	2	8	

Show your calculations for AVERAGE TURNAROUNG (5pts):

$t_2C_1 = 15-1 = 14$

$t_2C_2 = 4$

$t_2C_3 = 7-2 = 5$

Avg = (14+4+5)/3 = 23/3

t_2C_1 = ~~8~~ 15 − 1 = 14

t_2C_2 = 7 − 0 = 7

t_2C_3 = 5 − 2 = 3

avgT = (14+7+3)/3 = $\frac{24}{3}$ = 8.0

(a) In which CPU scheduling algorithm is starvation a problem. Priority.

(b) What causes starvation?
High priority processes always preempt/block low priority process

(c) Explain how aging can be used to prevent it.
Periodically raise priority of all processes (not to exceed a max value)

==
PART II: Scheduling Calculations (min 29 point, max 38 points)
==

ALL THE PROBLEMS USE THIS SET OF PROCESSES.
Suppose that the following processes arrive for execution at the times indicated. In order a for a process to complete, its total CPU time must be the runt time listed.

MANDATORY: PROBLEMS 1, 2, and 5; either 3 or 4 -- CIRCLE the one you pick. Others OPTIONAL.

Arrival Time:	0	1	2	
Process:	P1	P2	P3	
Run Time:	8	4	1	(total time needed)
Priority:	2	3	1	(only for priority scheduling, 3-HIGHEST)

C.1. (8 pts) FCFS SCHEDULING: Show schedule as a Gantt chart (4pts). Compute THROUGHPUT.

process:	P1	P2	P3				
time used:	8	4	1				

Show your THROUGHPUT calculations (4 pts):

total time = 8 + 4 + 1 = 13
completed processes = 3

Thruput = # completions / time = 4/13

C.2. ROUND-ROBIN SCHEDULING. Quantum=3. Show the schedule as a Gantt chart (5pts).

process:	P1	P2	P3	P1	P2	P1		
time used:	3	3	1	3	1	2		

AVG = 27/3 = 9

$t_2C_1 = 12$
$t_2C_2 = 5-1$
$t_2C_3 = 12+1-2 = 11$

C.3. SJF SCHEDULING: Show the schedule as a Gantt chart (4 pts). Compute AVERAGE TURNAROUND TIME.

4pts

process:	P1	P3	P2	P1	P2	P1	P3	
time used:	8	1	4	1	4	7	1	

Show your calculations for AVERAGE TURNAROUNG (5pts): Time to Complete (t2C

$t_2C_1 = 8$
$t_2C_2 = 8+1-1 = 8$
$t_2C_3 = 8+1+4-2 = 11$

Avg = (8+8+11)/3 = 27/3 = 9

====================

Arrival Time:	0	1	2	
Process:	P1	P2	P3	
Run Time:	8	4	1	(total time needed)
Priority:	2	3	1	(only for priority scheduling, 3-HIGHEST)

==================

C.4. SHORTEST REMAINING TIME SCHEDULING: Show the schedule as a Gantt chart (4 pts).
Compute the AVERAGE WAITING TIME for this schedule.

process:	P1	P2	P3	P2	P1			
time used:	1	1	1	3	7			

Show your AVG WAITING TIME calculations (5 pts):

$w_1 = 5$

$w_2 = 1$

$w_3 = 0$

Avg = 6/3 = 2

C.5. PRIORITY SCHEDULING. Show the schedule as a Gantt chart (4 pts). Then calculate the
RESPONSE TIME for process P3.

process:	P1	P2	P1	P3				
time used:	1	4	7	1				

Show your calculations for P3 response time (3 pts):

P3: arrives at 2
served at 12
response time = 12 - 2 = 10