

Faculty of Engineering & Technology Electrical & Computer Engineering Department

Operating Systems

ENCS 339

Project #1

"Simulation of CPU scheduling algorithms"

PREPARED BY:

Waseem sayara	ID: (#1182733).
Anas Barakat	ID: (#1180180).

INSTRUCTOR: Dr. Adnad H. Yahya.

SECTION: 2

DATE: 18-11-2020

***** Abstract:

Computer systems supporting multiprogramming or multitasking execute multiple programs or tasks concurrently. Multiprogramming increases CPU utilization by organizing jobs so that the CPU always has something to execute. To have several jobs ready to run, the system must keep all of them in memory at the same time for their selection one-by-one. This work involves development of a simulator for CPU scheduling. It has been developed as a comprehensive tool, which runs a simulation in real time and generates useful data to be used for evaluation. This simulator can be used for measuring performance of different scheduling algorithms. It simulates First Come First Serve (FCFS) scheduling, Shortest Job First (SJF) scheduling, Shortest Remaining Time First(SRTF) scheduling, priority scheduling with preemption (PRI), priority scheduling with out preemption (PRI), and Round Robin (RR) scheduling. A user-friendly and mouse- driven graphical user interface has been integrated.

Table of Contents

*	Abstract:	. 2
*	Theory:	4
*	Design Philosophy and Program Implementation:	5
*	How the program work:	. 7
*	Assumptions:	. 7
	Extra work:	
*	Results	. 8
-	Input file 1:	. 8
-	Input file 2:	12
-	Input file 3:	16
-	Generate process :	20
*	References	24

***** Theory:

CPU scheduling is an operation which allows one process to use the CPU while the execution of another process is on hold due to unavailability of any resource like I/O etc, thereby making full use of CPU. The aim of CPU scheduling is to make the system efficient, fast and fair. There are many different CPU-scheduling algorithms:

- 1. **FCFS:** It is a non-preemptive, and the simplest CPU-scheduling algorithm. With this scheme, the process that requests the CPU first is allocated the CPU first. The implementation of the FCFS policy is easily managed with a FIFO queue.
- 2 **SJF**: It is non-preemptive, ready queue treated as a priority queue based on smallest CPU time requirement, arriving jobs inserted at proper position in queue, dispatcher selects shortest job (1st in queue) and runs to completion.
- 3. **RR**: It is similar to FCFS scheduling, but preemption is added to enable the system to switch between processes. A small unit of time, called a time quantum or time slice, is defined. The ready queue is treated as a circular queue. The CPU scheduler goes around the ready queue, allocating the CPU to each process for a time interval of up to 1 time quantum.
- 4. **SRTF**: A preemptive version of SJF algorithm, which preempts the currently executing process when a less remaining time process becomes available.
- 5. **Priority Scheduling**: A priority is associated with each process, and the CPU is allocated to the process with the highest priority.
 - **a) Priority scheduling with preemption:** the task with higher priority than the current task being executed arrives then the control of the CPU is taken from the current task and given to the higher priority task.
 - **b) Priority scheduling without preemption:** the task with higher priority does arrive, it has to wait for the current task to release the CPU before it can be executed.

There are many different criteria's to check when considering the "best" scheduling algorithm, they are:

• CPU Utilization: To make out the best use of CPU and not to waste any CPU cycle, CPU would be working most of the time(Ideally 100% of the time). Considering a real system, CPU usage should range from 40% (lightly loaded) to 90% (heavily loaded.).

• Turnaround Time: It is the amount of time taken to execute a particular process, i.e. The interval from time of submission of the process to the time of completion of the process(Wall clock time).

• Waiting Time: The sum of the periods spent waiting in the ready queue amount of time a process has been waiting in the ready queue to acquire get control on the CPU.

* Design Philosophy and Program Implementation:

When you start the program a window looks like figure 1 opens.

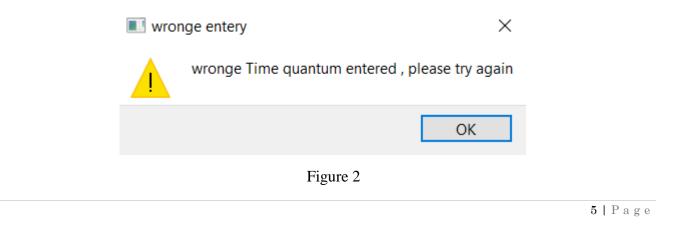
PU Scheduling Algorithms		:
Simulati	ion of CPU Schedulin	ng Algorithms
ELECT ALGORITHM : RR	V TIME QUANTUM :	SELECT FILE : input1 ~
	RUN	GENERET PROCESS
AVERAGE TURNAROUND TIME	AVERAGE WAITING TIME	OVERALL CPU USAGE
	EXIT	

Figure 1

The window gives the user the option to choose the scheduling algorithm using a combo box (the RR algorithm is default), and to choose from 3 input files also in a combo box (input1.txt is default), after the user chooses the algorithm and the input file he wants, he have to click in the RUN button to start the simulation.

The output of the simulation is printed in two tables, the upper one has the values of each process data, while the bottom one prints which process runs at every time unit in the CPU (X stands for none).

In case of user choosing the RR algorithm, he must enter the Time Quantum he wants to use and it must be an integer, else an error massage looks like Figure2 will pop up.



The program also gives the user the ability to simulate random generated processes by just entering the count of them, and then click on the Generate button.

The constrains on the random generated processes are:

- The names for the process starts from 1.
- The arrival time for each process is an integer between 1 and 100.
- The burst time for each process is an integer between 1 and 1000.
- The priority for each process is an integer between 1 and 100.
- The repeat is always 1.
- The interval and the deadline are not used so they are assigned to X.

If the user entered a wrong number format in the generate box a massage will pop up like the one in figure 3, and then the program will close.



Figure 3

***** How the program work:

We created a distinct code file for every schedule algorithm that can be called from the main file, which organize the program, and makes sure that it works correct and errors free.

When the user clicks on the RUN button the program will get the value of algorithm combo box and the input file too, then according to the value of the algorithm it will call the wanted file of code to run in the screen.

In non-preemptive algorithms like FCFS, SJF and PRI without preemption, at first the program reads the input file and store it two lists, one to keep the original data and other is to modified as the simulation goes, then the program enters a loop which presents the CPU that breaks only when all the processes are finished executing when the loop starts, first, it checks the available processes at the time it reached using a special function that compares the time using the arrival time for each process, the available processes enters the queue, then the program calls another function to choose the process to be excused according the algorithm chosen, once the CPU gets the process, it keeps running until it finished all its burst, then the current time increases by the process burst and start over the loop.

In preemptive algorithms like SRTF and PRI with preemption, at first the program reads the input file and store it two lists, one to keep the original data and other is to modified as the simulation goes, then the program enters a loop which presents the CPU that breaks only when all the processes are finished executing when the loop starts, first, it checks the available processes at the time it reached using a special function that compares the time using the arrival time for each process, the available processes enters the queue, then the program calls another function to choose the process to be excused according the algorithm chosen, once the CPU gets the process, it runs for only one time unit, then recheck if it's the turn of another process to run or not, if there is another process at that time with higher priority then it enters the CPU instead of the running one, else it keeps running.

Aging:

The priority for the waiting processes is increases every some time that is random value for every time the program runs, we used this method to solve the starvation in PRI algorithms.

***** Assumptions:

- **4 Priority:** the higher number presents the higher priority.
- **4** Aging: the aging time is random.
- **4 Priority value:** the process priority value is an input from the file.

*****Extra work:

- **4** Friendly user interface.
- **4** Extra algorithms (6 instead of 4).

* Results

We created 3 different input files each file contains different processes were generated for each algorithm.

Input file 1:

1) **RR** (with time quantum=1):

SE	LECT	ALGO	RITH	M: RR				UANTUM :	1		SELEC		: in	nput 1	
	ID		AT	Burst	Priorit	y Repe	eat Interv	al Deadline	e Start	Enc	I TurnA	Around	Wa	it	WTA
1	P1	2		8	1	1	×	×	2	21	19		11	2	.375
2	P2	4		5	4	1	x	x	5	19	15		10	3	.0
3	P3	6		2	3	1	x	x	8	13	7		5	3	.5
4	P4	3		4	2	1	x	x	3	15	12		8	3	.0
							RUN				GENERE	T PRO	CESS		
	AVER	AGE TUP	RNAROL	IND TIME			AVER	AGE WAITING	ТМЕ			OVER	ALL CPU	USAGE	
							8.5				19				
13	.25									1					
13	.25														

2) **RR** (with time quantum=2):

	LECT	ALGO	RITH	M:RF	ł		~	T	IME QU		: 2	2			SELEC	T FILE	: ir	nput1		~
	ID		AT	Burst	t I	Priority	Repea	at	Interval	Deadli	ne	Start	E	End	Turn/	Around	Wa	ait	WTA	
1	P1	2		8	1		1	x		x	2	2	21		19		11	2	375	
2	P2	4		5	4		1	x		x	6	5	19		15		10	3.0		
3	P3	6		2	3		1	x		x	1	10	12		6		4	3.0		
4	P4	3		4	2		1	x		x	4	4	14		11		7	2	75	
	AVER	AGE TUP	RNAROU	ND TIME					AVERAG	E WAITIN	G TIM	E				OVER	ALL CPU	J USAGE		
12	.75							8.0							19					
	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	
1	x	x	P1	P1	P4	P4	P2	P2	P1	P1 P	3	P3 F	94	P4	P2	P2	P1	P1	P2	
<																				>
٢.										EXIT										

3) FCFS:

CPU Scheduling Algorithms

Simulation of CPU Scheduling Algorithms

4) SJF:

CPU Scheduling Algorithms

Simulation of CPU Scheduling Algorithms SELECT ALGORITHM : SJF TIME QUANTUM : SELECT FILE : \sim input1 \sim Priority End TurnAround Wait WTA ID AT Burst Repeat Interval Deadline Start 1 P1 2 8 1 1 x Х 2 10 8 0 1.0 2 P2 5 4 х х 16 17 4 1 21 12 3.4 3 P3 6 2 3 1 х х 10 12 6 4 3.0 4 P4 3 4 2 1 х х 12 16 13 9 3.25 RUN GENERET PROCESS AVERAGE TURNAROUND TIME AVERAGE WAITING TIME OVERALL CPU USAGE 6.25 19 11.0 0 1 2 3 5 6 7 8 9 10 11 12 13 14 15 16 17 18 4 1 X х P1 P1 P1 P1 P1 P1 P1 P1 P3 P3 P4 P4 P2 P2 P4 P4 P2 P < > EXIT 9 | Page

- 🗆 🗙

5) **SRTF**:

CPU Scheduling Algorithms

- 🗆 ×

	LECT	ALGORI	THM : S	RTF		~	TIME QU	ANTUM :	0		SELECT FILE	in in	put1
	ID	AT	Bui	st P	riority	Repeat	Interval	Deadline	Start	End	TurnAround	Wai	t WTA
	P1	2	8	1		1	x	x	2	21	19	11	2.375
2	P2	4	5	4		1	x	х	9	14	10	5	2.0
;	P3	6	2	3		1	х	х	7	9	3	1	1.5
ł	P4	3	4	2		1	х	х	3	7	4	0	1.0
							DUN					CESS	
	A)(ED)			-			RUN			G	ENERET PRO		
.0		AGE TURNA	ROUND TIM	E		4		E WAITING T	IME	C		CESS RALL CPU	USAGE
.0		age turna	2 3	E	5	6	AVERAG	E WAITING T 9 10		12 13	OVEF		USAGE 17 18

6) PRI (PREEMPTIVE) :

- 🗆 🗙

Simulation of CPU Scheduling Algorithms

										EXIT									
c																			>
I	x	x	P1	P4	P2	P2	P2	P2	P2	P3 P3	P4	P4	P4	P1 F	1	P1	P1	P1	F
	0	1	2	3	4	5	6	7	8	9 10) 11	12	13	14	15	16	17	18	T
	.0							5.25						19					
	AVER	AGE TUI	RNAROU	ND TIME			Г	5.25	AVERAG	GE WAITING T	IME	٦		19	OVER	ALL CPU	USAGE		
								RUN	N				GI	ENERET	PRO	CESS			
1	P4	3		4	2		1	x		x	3	14		11		7	2	.75	
3	P3	6		2	3		1	x		x	9	11		5		3	2	.5	
2	P2	4		5	4		1	х		x	4	9		5		0	1	.0	
1	P1	2		8	1		1	x		x	2	21		19		11	2	.375	
			AT	Burst	: P	riority	Repea	at I	Interval	Deadline	Star	t	End	TurnArd	ound	Wa	it	WTA	

7) PRI (WITHOUT PREEMPTIVE) :

CPU Scheduling Algorithms

1	x	х	P1	P1	P1	P1	P1 F	P1	P1	P1 P	2	P2	P2	P2	P2	P3	P3	P4	P4
	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
12	2.0							7.25							19				
	AVEF	RAGE TU	RNAROU	IND TIME			_		AVERAG		G TIN	ME				OVER	ALL CPU	USAGE	
								RUN	1					GE	NERE	T PRO	CESS		
4	P4	3		4	2		1	X		X		17	21		18		14	4	.5
	P3	6		2	3		1	X		x		15	17		11		9	5.	
2	P2	4		5	4		1	х		х		10	15		11		6	2	
1	P1	2		8	1		1	x		х		2	10		8		0	1.	.0
	ID		AT	Burst	P	riority	Repea	it I	nterval	Deadl	ine	Start		End	Turn/	Around	Wai	it	WTA
SI							אי א חד וי ~		ME QU	ANTUN						T FILE		put1	

—

Input file 2:

1) **RR** (with time quantum=1):

CPU Scheduling Algorithms

Simulation of CPU Scheduling Algorithms SELECT ALGORITHM : RR SELECT FILE : input2 V TIME QUANTUM : 1 \sim ID AT Burst Priority Repeat Interval Deadline Start End TurnAround Wait WTA 1 1 P1 3 6 1 х х 3 19 16 10 2.667 2 P2 0 8 4 1 х х 0 17 17 9 2.125 3 P3 5 2 3 1 х х 6 10 5 3 2.5 2 17 4 P4 9 10 1 х х 11 26 7 1.7 RUN GENERET PROCESS AVERAGE TURNAROUND TIME AVERAGE WAITING TIME OVERALL CPU USAGE 13.75 7.25 26 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 1 P2 P2 P2 P1 **P3** P1 P2 P1 P2 P1 P2 P1 P2 P1 P2 P3 P4 P4 P4 P < > EXIT

 \times

2) **RR** (with time quantum=2):

	LECT	ALGO	RITHN	1: RR		~	TIME QU	JANTUM :	2		SELECT FIL	.E: inp	out2
	ID	1	AT	Burst	Priority	Repea	t Interval	Deadline	Start	End	TurnAroun	d Wait	WTA
1	P1	3		6	1	1	x	х	4	18	15	9	2.5
2	P2	0	1	8	4	1	x	х	0	14	14	6	1.75
3	P3	5	1	2	3	1	x	х	8	10	5	3	2.5
4	P4	9		10	2	1	х	х	14	26	17	7	1.7
	AVER	AGE TUR	RNAROUI	ND TIME			RUN	GE WAITING T	IME		ENERET PR	RALL CPU U	SAGE
12	.75					e	5.25				26		
	0	1	2	3	4 5	6	7 8	9 10	11	12 13	14 15	16	17 18
		P2	P2	P2	P1 P1	P2 F	2 P3	P3 P1	P1	P2 P2	P4 P4	P1 F	P1 P4

3) FCFS:

CPU Scheduling Algorithms

SELECT ALGORITHM : FCFS \sim TIME QUANTUM: 0 input2 SELECT FILE : \sim ID AT Burst Priority Repeat Interval Deadline Start End TurnAround Wait WTA 1 P1 3 х 8 14 5 1.833 6 1 1 Х 11 4 х 0 2 P2 8 1 х 0 8 8 1.0 0 3 P3 5 2 3 1 Х х 14 16 11 9 5.5 4 P4 9 10 2 1 х х 16 17 7 1.7 26 RUN GENERET PROCESS AVERAGE TURNAROUND TIME AVERAGE WAITING TIME OVERALL CPU USAGE 5.25 11.75 26 0 2 5 6 7 8 9 10 11 12 13 17 18 1 3 4 14 15 16 1 P2 P4 P2 P2 P2 P2 P2 P2 P2 P1 P1 P1 P1 **P1** P1 P3 P3 P4 P4 P < > EXIT 4) SJF: CPU Scheduling Algorithms \times Simulation of CPU Scheduling Algorithms

	ID		AT	Burst	P	riority	Repe	at Int	erval	Deadline	Start	End	Turn/	Around	Wa	it	WTA
1	P1	3		6	1	nonty	1	x	x		10	16	13	lound	7		.167
	P2	0		8	4		1	x	x		0	8	8		0		.0
	P3	5		2	3		1	x	x		8	10	5		3	2.	
	P4	9		10	2		1	x	×		16	26	17		3 7	2.	
	A1/ED	ACT						RUN	(EDACE V		MF	G	ENERE				
10	AVER	AGE TUP	RNAROU	ND TIME					/ERAGE V	VAITING TI	ME	G	ENERE		CESS ALL CPU	USAGE	
10		AGE TUP	RNAROU	ND TIME				A	/ERAGE V	VAITING TI	ME	G				USAGE	
10		AGE TUP	RNAROU 2	ND TIME	4	5	6	A		VAITING TI 9 10		12 13	26			USAGE	18
	.75				4 P2	5 P2	6	A 4.25	8	9 10	11		26	OVER	ALL CPU		18 P4
	0 P2	1	2	3			6	A 4.25 7	8 3 P3	9 10 P1	11	12 13	26	OVER 15	ALL CPU	17	
1	0 P2	1	2	3			6	A 4.25 7	8 3 P3	9 10	11	12 13	26	OVER 15	ALL CPU	17	

Simulation of CPU Scheduling Algorithms

5) **SRTF**:

CPU Scheduling Algorithms

Simulation of CPU Scheduling Algorithms SELECT ALGORITHM : SRTF \sim TIME QUANTUM : 0 input2 SELECT FILE : ID AT Priority TurnAround Wait WTA Burst Repeat Interval Deadline Start End 7 2.167 1 P1 3 6 1 1 Х Х 10 16 13 2 P2 0 8 4 1 х х 0 10 10 2 1.25 5 7 3 P3 5 2 3 1 Х Х 2 0 1.0 4 P4 9 10 2 1 х х 16 26 17 7 1.7 RUN GENERET PROCESS AVERAGE TURNAROUND TIME AVERAGE WAITING TIME OVERALL CPU USAGE 10.5 4.0 26 7 0 1 2 3 4 5 6 8 9 10 11 12 13 14 15 16 17 18 1 P2 P2 P2 P2 P2 **P**3 **P**3 P2 P2 P2 P1 P1 P1 P1 P1 P1 P4 P4 P4 < > EXIT

6) PRI (PREEMPTIVE) :

CPU Scheduling Algorithms

 \times

P

Simulation of CPU Scheduling Algorithms

	L																		
	0 P2	1 P2	2 P2	3 P2	4 P2	5 P2	6 P2	7 P2	8 P3	9 P3 P4	10 I	11 P1 F	12 21	13 P1	14 P1	15 P1	16 P1	17 P4	18 P4
																		1	
1	1.0							4.5							26				
	AVER	AGE TUF	NAROU	IND TIME	:			RU		GE WAITIN	6 TIM	E		GE	NERE	T PRO	CESS	USAGE	:
1	P4	9		10	2		1	×	(х		10	26		17		7	1	.7
3	P3	5		2	3		1	Х	(х	8	В	10		5		3	2	.5
2	P2	0		8	4		1	Х	(х	(D	8		8		0	1	.0
1	P1	3		6	1		1	х	(х		11	17		14		8	2	.333
1	ID P1		AT	Burs 6		Priority	Rep 1		Interval (Deadli X		Start 11		End		Around	Wai 8		

 \times

 \sim

7) PRI (WITHOUT PREEMPTIVE) :

CPU Scheduling Algorithms

E	LECT	ALGOR	ITHM :	PRI (N	ION-PREE	MPTI' ~	TIME QU	IANTUM :	0		SELECT	FILE	: in	put2	
	ID	AT	ГВ	urst	Priority	Repeat	Interval	Deadline	Start	End	TurnA	round	Wai	t	WTA
1	P1	3	6		1	1	х	х	10	16	13		7	2.	167
2	P2	0	8		4	1	х	х	0	8	8		0	1.0	0
3	P3	5	2		3	1	х	х	8	10	5		3	2.	5
4	P4	9	10		2	1	X	X	16	26	17 GENERET	PRO	7 CESS	1.	7
4			10		2	1	RUN				17 GENERET				7
					2	1	RUN						CESS		7
	AVER		IAROUND T	IME	2	1	RUN		(ME		GENERET		CESS		18

_

Input file 3:

1) **RR** (with time quantum=3):

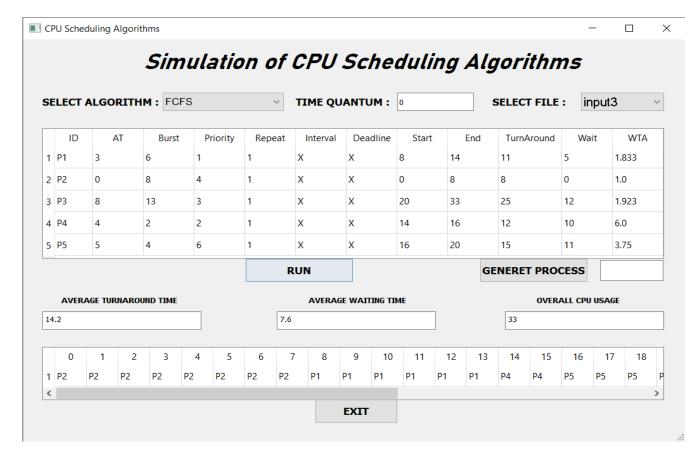
CPU Scheduling Algorithms

Simulation of CPU Scheduling Algorithms SELECT ALGORITHM : RR TIME QUANTUM : 3 SELECT FILE : input3 \sim Priority Wait WTA ID AT Burst End Repeat Interval Deadline Start TurnAround 1 P1 3 6 1 х х 3 17 14 8 2.333 1 2 P2 0 8 4 1 х х 0 22 22 14 2.75 3 P3 8 13 3 1 х x 17 33 25 12 1.923 4 P4 4 2 2 1 х х 9 11 7 5 3.5 5 P5 5 4 6 1 х х 11 23 18 14 4.5 RUN GENERET PROCESS AVERAGE WAITING TIME OVERALL CPU USAGE AVERAGE TURNAROUND TIME 17.2 10.6 33 0 2 3 4 5 6 7 8 9 10 11 12 13 14 15 17 18 1 16 1 P2 P2 P2 P1 **P1** P1 P2 P2 P2 P4 P4 P5 P5 P5 P1 P1 P1 **P3 P3** Ρ > < EXIT

2) **RR** (with time quantum=5):

	LECI	ALGO	RITH	M:RF	R			~	IME QU	IANTUM	: 5	i			SELEC	T FILE	: ir	nput3	
	ID	A	۸T	Burs	t F	riority	Rep	eat	Interval	Deadlin	e	Start		End	Turn	Around	Wa	it	WTA
1	P1	3		6	1		1)	K	x	5	5	25		22		16	3	.667
2	P2	0		8	4		1)	ĸ	х	0)	19		19		11	2	.375
3	P3	8		13	3		1)	×	x	1	9	33		25		12	1	.923
4	P4	4		2	2		1)	×	x	1	0	12		8		6	4	.0
5	P5	5		4	6		1)	×	х	1	2	16		11		7	2	.75
								RU	JN					GE	NERE	T PRO	CESS		
	AVER	AGE TUR	NAROU	IND TIME					AVERAG	E WAITING	тімі	E				OVER	ALL CPU	USAGE	
	.0							10.4							33				
17				3	4	5	6	7	8	9 1	0	11	12	13	14	15	16	17	18
17	0	1	2	5															

3) FCFS:



4) SJF:

	LECT	ALGORIT	HM: SJF		\sim	TIME QU	ANTUM :	0		SELECT FILE	: inpu	ut3
	ID	AT	Burst	Priority	Repeat	Interval	Deadline	Start	End	TurnAround	Wait	WTA
1	P1	3	6	1	1	x	x	14	20	17	11	2.833
2	P2	0	8	4	1	x	x	0	8	8	0	1.0
3	P3	8	13	3	1	x	x	20	33	25	12	1.923
4	P4	4	2	2	1	x	х	8	10	6	4	3.0
5	P5	5	4	6	1	x	х	10	14	9	5	2.25
						RUN			GI	ENERET PRO	CESS	
	AVER/	AGE TURNAR	OUND TIME			AVERAG	E WAITING TI	ME		OVER	ALL CPU US	AGE
13		AGE TURNAR	OUND TIME		6.4		E WAITING TI	ME		OVER	ALL CPU US	AGE
13			2 3	4 5			E WAITING TI 9 10	11	12 13			AGE 17 18

5) **SRTF**:

	LECT	ALGOR	ITHM :	SRTF		\sim	TIME QU	JANTUM :	0		SELECT FIL	E: in	put3	
	ID	AT	B	urst	Priority	Repea	Interval	Deadline	Start	End	TurnAround	d Wai	it .	WTA
1	P1	3	6		1	1	x	x	14	20	17	11	2.8	
2	P2	0	8		4	1	x	x	0	10	10	2	1.2	5
3	P3	8	13		3	1	x	x	20	33	25	12	1.9	23
4	P4	4	2		2	1	x	x	4	6	2	0	1.0	
5	P5	5	4		6	1	x	x	10	14	9	5	2.2	5
							RUN		1		GENERET PRO	DCESS		
	AVER	AGE TURN	AROUND TI	МЕ			AVERA	GE WAITING T	IME		OVE	RALL CPU	USAGE	
12	.6					e	.0				33			
													47	
	0	1	2	3 .	4 5	6	7 8	9 10	11	12 13	3 14 15	16	17	18

6) PRI (PREEMPTIVE) :

SE	LECT	ALGO	ORITH	M: PR	(PRE	EMPTI	VE)	~	TIME QU		:	0			SELEC	T FILE	: ir	nput3		
	ID		AT	Burst	Pi	iority	Rep	eat	Interval	Deadli	ne	Start		End	Turn	Around	Wa	ait	WTA	4
1	P1	3		6	1		1		х	x		27	33		30		24	5	.0	
2	P2	0		8	4		1		Х	x		0	12		12		4	1	.5	
3	P3	8		13	3		1		х	x		12	25		17		4	1	.308	
4	P4	4		2	2		1		х	х		25	27		23		21	1	1.5	
5	P5	5		4	6		1		х	х		5	9		4		0	1	.0	
								R	UN					G	ENERE	T PRO	CESS			
	AVER	AGE TU	RNAROL	UND TIME					AVERAG		G TII	МЕ				OVER	ALL CPU	J USAGE		
17	.2							10.6							33					_
	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	3
1	P2	P2	P2	P2	P2	P5	P5	P5	P5	P2 P	2	P2	P3	P3	P3	P3	P3	P3	P3	
<																				

7) PRI (WITHOUT PREEMPTIVE) :

CPU Scheduling Algorithms

- 🗆 X

Simulation of CPU Scheduling Algorithms

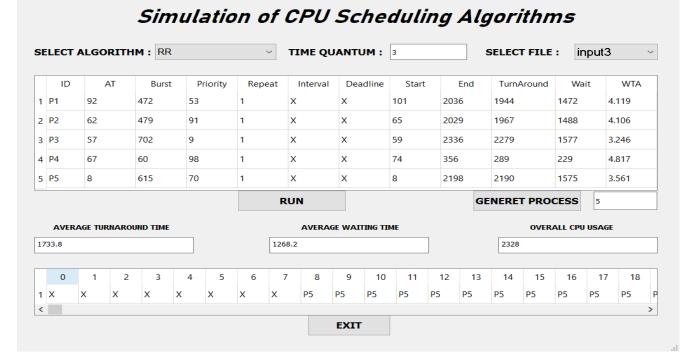
SELECT ALGORITHM : PRI (NON-PREEMPTI' V TIME QUANTUM : 0 SELECT FILE : input3 \sim Priority ID AT Burst Deadline TurnAround Wait WTA Repeat Interval Start End 1 P1 3 6 1 1 Х Х 27 33 30 24 5.0 4 Х 0 8 0 2 P2 0 8 1 Х 8 1.0 3 P3 8 13 3 1 х х 12 25 17 4 1.308 2 2 х 25 4 P4 4 1 х 27 23 21 11.5 5 P5 5 4 6 1 х х 8 12 7 3 1.75 RUN GENERET PROCESS AVERAGE TURNAROUND TIME OVERALL CPU USAGE AVERAGE WAITING TIME 17.0 10.4 33 4 5 7 8 9 0 1 2 3 6 10 11 12 13 14 15 16 17 18 1 P2 P2 P2 P2 P2 P2 P2 P5 P5 P5 P3 **P**3 P2 P5 P3 **P**3 P3 **P**3 P3 P < > EXIT

19 | Page

Generate process :

1) RR (with time quantum=3 & generate process = 5):

CPU Scheduling Algorithms



 \times

2) RR (with time quantum=2 & generate process =4):

E	LECT	ALGOR	ITH	M:RR				~ 1	TIME QU		:	2			SELEC	r FILE	: ir	nput3		
	ID	A	г	Burst	Pr	iority	Repe	eat	Interval	Deadli	ne	Start		End	TurnA	Around	Wa	it	WTA	
1	P1	65		154	66		1	:	x	x		70	646		581		427	3.	773	
2	P2	33		153	83		1	:	х	х		36	624		591		438	3.	863	
3	P3	0		163	2		1	1	х	х		0	589		589		426	3.	613	
4	P4	9		809	93		1	1	x	х		10	127	9	1270		461	1.	57	
	AVER	AGE TURI	IARO	JND TIME					AVERAG	E WAITIN	G TIN	ME				OVER	ALL CPU	USAGE		
75	7.75							438.0)						1279					_
	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	
1	P3	P3	P3	P3	P3	P3	P3	P3	P3	P3 P	4	P4	93	P3	P4	P4	P3	P3	P4	
<																				2

3) FCFS (with generate process = 3):

CPU Scheduling Algorithms

6E	LECT	ALGORI	THM :	FCFS	\$	\sim	TIME QU	JANTUM :			SELECT	FILE	: in	nput3		~
	ID	AT	В	urst	Priority	Repeat	Interval	Deadline	Start	End	TurnArd	round	Wa	it	WTA	
	P1	34	438		53	1	х	x	1274	1712	1678		1240	3	3.831	
	P2	21	276		39	1	х	x	998	1274	1253		977	4	1.54	
3	P3	7	991		62	1	х	х	7	998	991		0	1	.0	
							RUN			G	ENERET	PRO	CESS	3		
	AVERA	AGE TURNA	ROUND T	ІМЕ				GE WAITING T	IME	G			CESS			
3(AGE TURNA 333333333	ROUND T	IME				GE WAITING TI	ME	C					Ξ	
3(IME 3	4 5		AVERAG	SE WAITING TI		12 13	1705				18	
	07.33333	333333333	2			7.	AVERAC 39.0 7 8		11		1705	OVER	ALL CPU	USAGE		

4) SJF (with generate process =6):

CPU Scheduling Algorithms Simulation of CPU Scheduling Algorithms SELECT ALGORITHM : SJF \sim **TIME QUANTUM :** SELECT FILE : input3 WTA ^ Deadline ID AT Burst Priority Repeat Interval Start End TurnAround Wait 39 52 1 Х х 181 220 134 95 3.436 1 P1 86 73 178 0 1.0 2 P2 3 178 1 Х Х 3 181 774 3 P3 8 467 3 1 Х Х 1241 1233 766 2.64 4 P4 79 454 67 1 Х Х 320 774 695 241 1.531) 27) > 5 D5 5 50 862 15 Y Y 17/1 2102 2015 1102 1 RUN GENERET PROCESS 6 AVERAGE TURNAROUND TIME AVERAGE WAITING TIME OVERALL CPU USAGE 751.0 401.0 2100 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 P2 P2 1 X х х P2 < EXIT 21 | Page

\times

 \sim

P

>

Simulation of CPU Scheduling Algorithms

5) SRTF (with generate process =3):

CPU Scheduling Algorithms

SELECT ALGORITHM : SRTF TIME QUANTUM : input3 \sim SELECT FILE : \sim AT Priority Start Wait WTA ID Burst Repeat Interval Deadline End TurnAround 1 P1 62 450 1 1 х Х 234 684 622 172 1.382 2 P2 5 857 42 1 Х Х 5 1535 1530 673 1.785 3 P3 11 223 82 1 х х 11 234 223 0 1.0 RUN GENERET PROCESS 3 AVERAGE TURNAROUND TIME AVERAGE WAITING TIME OVERALL CPU USAGE 1530 791.6666666666666 281.6666666666666 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 P2 P2 P2 1 X Х х Х х P2 P2 P2 P3 P3 P3 **P**3 P3 **P**3 P3 **P3** P < > EXIT

Simulation of CPU Scheduling Algorithms

_

 \times

6) **PRI(PREEMPTIVE)** (with generate process =5):

sE	LECT	ALGO	RITH	1 : PR	I (PREE	EMPTI	VE) ~	T	IME QU	JANTUM	1:				SELEC	T FILE	: ir	nput3	
	ID	4	AT	Burst	Pri	iority	Repea	at	Interval	Deadli	ne	Start	E	Ind	TurnA	Around	Wa	it	WTA
1	P1	23	!	944	67		1	X		х		23	967		944		0	1.	0
2	P2	1	1	98	43		1	x		х		1	1043	3	1042		944	1(0.633
3	P3	3		304	12		1	x		x		2008	2312	2	2309		2005	7.	595
4	P4	47	1	965	50		1	х		x		1043	2008	3	1961		996	2.	032
5	P5	80		288	17		1	х		х		2312	2600)	2520		2232	8.	75
	AVER/	AGE TUR	NAROU	ND TIME			r		AVERAG	E WAITIN	G TIN	ME				OVER	ALL CPU	USAGE	
17	55.2						[1235.4							2599				
	0	1	2	3	4	5	6	7	8		10	11	12	13	14	15	16	17	18
1	X	P2	P2	P2	P2	P2	P2	P2	P2	P2 P2	2	P2 F	2	P2	P2	P2	P2	P2	P2
<																			
										EXIT									

7) **PRI(NON PREEMPTIVE)** (with generate process =4):

CPU Scheduling Algorithms

- 🗆 🗙

SE	LECT	ALGORI	THM: PR	I (NON-PRE	EMPTI' ~	TIME QU	IANTUM :			5	SELECT F	ILE :	input	13	`
	ID	AT	Burst	Priority	Repeat	Interval	Deadline	Start	E	nd	TurnArou	und	Wait	WTA	\ \
1	P1	40	82	52	1	x	x	1456	1538		1498	141	16	18.268	
2	P2	2	443	51	1	х	х	2	445		443	0		1.0	
3	P3	80	465	91	1	х	х	991	1456		1376	911	I	2.959	
4	P4	6	546	86	1	х	х	445	991		985	439)	1.804	
						RUN				GF		ROCES	S 4		
						RUN				GE	NERET P	ROCES	S 4		
	AVERA	AGE TURNAF	ROUND TIME				E WAITING TI	ME		GE				\GE	
10	AVER/ 75.5	AGE TURNAF	ROUND TIME		6		E WAITING TI	ME		GE				ıge	
10			ROUND TIME	4 5	6	AVERAG	E WAITING TI 9 10		12	GE 13	0 1536	OVERALL (CPU USA	JGE 7 18	
	75.5		2 3	4 5 P2 P2		AVERAG 91.5 7 8		11			0 1536	15 1	CPU USA		

Simulation of CPU Scheduling Algorithms

*****References

- <u>https://cutt.ly/6tqQF7</u> Wednesday 18-11-2020 @ 5:00 pm

- <u>https://www.geeksforgeeks.org/difference-between-preemptive-priority-based-and-non-preemptive-priority-based-cpu-scheduling-algorithms/</u> Wednesday 18-11-2020 @ 5:30 pm

- A SIMULATION PROJECT FOR AN OPERATING SYSTEMS COURSE(John K. Estell, The University of Toledo)