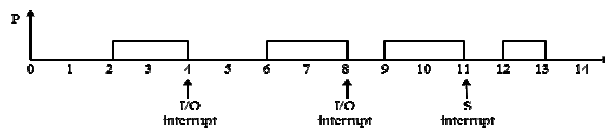


Q21

w0	1
C	7
start	2
end	13
RT	12
WT	5

Burst	C (Burst)	C(T)	T(t)	WT(t)	E(t)
1-4	2	7-5	0-2	0-1	0-12
4-8	2	5-3	2-4	1-3	
8-13	3	3-0	4-7	3-5	



Q22

a =	0,7
P	
ti	Ti
2,00	1,00
2,00	1,70
3,00	1,91

MSE
1,00
0,09
1,19
0,76

Q24

burst		2-3	3-4	4-5	5-6
A	T(t)	0-1	1	1	1-2
	E(t)	0-1	1-2	2-3	3-4
	R(t)	0/0*1=0	1/1*2=2	1/2*3=3/2 = 1,5	1/3*3=1
B	T(t)			0-1	1
	E(t)			0-1	1-2
	R(t)			0/0*3=0	=1/1*3=3
C	T(t)		0-1	1	1-2
	E(t)		0-1	1-2	2-3
	R(t)		0/0*2=0	1/1*3=3	=1/2*3=3/2=1,5

Q31

scheduling		RA	RB	RC	Lock	Section
Process	Code					
B	1	∅	∅	∅	0	∅
A	1,2	0			1	A
B	2,3,2,3		1			
A	3,4,5				0	∅
C	1,2			0	1	C
B	2,3,2					

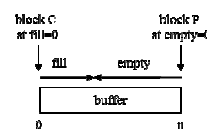
Q33

scheduling		buffer	fill		empty	
Process	Code		value	Q	value	Q
C	1 2 3	1	1	∅	0	∅
C	1 2	0	0			
P	2	1				
C	3 1	0	0	C	1	
P	3 1 2 3	1	1	∅	0	
P	1					P

fill = 0, empty = n are semaphores
buffer is the data structure

consumer
loop
(1) down fill
(2) pop item from buffer
(3) up empty

producer
loop
(1) down empty
(2) push a new item in buffer
(3) up fill



Q34

scheduling		buffer	count	Conditions			Section	entry queue
Process	Code			full	empty	signal		
C1	1,3,4,5	2	2	P3				P1 C3 C2 P2 C1
		1	1					P3 P1 C3 C2 P2

with the mesa implementation,
C1 will access the monitor
and will free P3 to be pushed in the entry queue

scheduling		buffer	count	Conditions			Section	entry queue
Process	Code			full	empty	signal		
C1	1,3,4,5	2	2	P3				P1 C3 C2 P2 C1
		1	1			C1	C1-P3	P1 C3 C2 P2

with the Hoare implementation,
P3 will access right now
to the monitor after C1, the "if" is here

Q35

Consumer

```

loop
(1) down fill
(2) down mutex1
(3) pop item from buffer
(4) up mutex1
(5) up empty

```

Producer

```

(1) down empty
(2) down mutex2
(3) push a new item in buffer
(4) up mutex2
(5) up fill

```