CSIE 5452 HW I P10942A05 鄭閔軒

1. (1) worst case response of M_0

LHS	B_0	RHS	STOP
30	30	30	True

$R_0 = 30 + 10 = 40$ (msec)

(2) worst case response of M_1

Q_1	B_1	j	$Q_1+\tau$	T_j	$\frac{Q_1+\tau}{T_j}$	C_j	RHS	STOP?
30	30	0	30.1	50	1	10	40	False
40	30	0	30.1	50	1	10	40	True

$R_1 = Q_1 + C_1 = 40 + 30 = 70$ (msec)

(3)

Q_2	B_2	j	$Q_2+\tau$	T_j	$\frac{Q_2+\tau}{T_j}$	C_j	RHS	STOP?
20	20	0 1	20.1	50 200	1 1	10 30	60	False
60	20	0 1	60.1	50 200	2 1	10 30	70	true
70	20	0 1	70.1	50 200	2 1	10 30	70	true

$R_2 = Q_2 + C_2 = 70 + 20 = 90$ (msec)

3.

(1) worst case response time of τ_0

R_0	C_0	RHS	Stop?
10	10	10	True

$R_0 = 10$ (msec)

(2)

worst case response time of τ_1

R_1	C_1	j	R_1	T_j	$\frac{R_1}{T_j}$	C_j	RHS	STOP?
30	30	0	30	50	1	10	40	False
40	30	0	40	50	1	10	40	true

$R_1 = 40$ (msec)

(3) worst case response of τ_2

R_2	C_2	j	R_2	T_j	$\lceil \frac{R_2}{T_j} \rceil$	C_j	RHS	STOP?
20	20	0 1	20	50 200	1 1	10 30	60	false
60	20	0 1	60	50 200	2 1	10 30	70	false
70	20	0 1	70	50 200	2 1	10 30	70	true

$R_2 = 70$ (msec)

(4) Due to preemptive characteristic, we can ignore blocking time impact in response time analysis.

4、(1) frame arrival pattern (4, 10, 0, 3, 5, 6)

schedule pattern (4, 10, 1, 2, 6, 7)

(2) Frame arrival pattern : (m = 4 , p = 10 , a = 0, 3, 5, 6, 10, 13, 15, 16)

(3) Schedule pattern : (n = 4 , q = 10 , s = 1, 2, 6, 7, 11, 12, 16, 17)

(4)

k	$\max_{1\le j\le n}(s_{j+k}-s_j)$	=	$\min_{1\le i\le m}(a_{i+k-1}-a_i)$	=	Col3 − Col5
1	$\max_{1\le j\le 4}(s_{j+1}-s_j)$	$(j=2)\rightarrow 4$	$\min_{1\le i\le 4}(a_i-a_i)$	$(i=1,2,3,4)\rightarrow 0$	4
2	$\max_{1\le j\le 4}(s_{j+2}-s_j)$	$(j=1,2,3,4)\rightarrow 5$	$\min_{1\le i\le 4}(a_{i+1}-a_i)$	$(i=3)\rightarrow 1$	4
3	$\max_{1\le j\le 4}(s_{j+3}-s_j)$	$(j=2,4)\rightarrow 9$	$\min_{1\le i\le 4}(a_{i+2}-a_i)$	$(i=2)\rightarrow 3$	$\underline{6}$
4	$\max_{1\le j\le 4}(s_{j+4}, s_j)$	$(j=1,2,3,4)\rightarrow 10$	$\min_{1\le i\le 4}(a_{i+3}-a_i)$	$(i=0)\rightarrow 6$	4

(5)

worst response time = worst wait + transmission time

$= 6 + 1 = \underline{7}$ #

2、

(1)

signal: 0 response time: 1.44

signal: 1 response time: 2.04

signal: 2 response time: 2.56

signal: 3 response time: 3.16

signal: 4 response time: 3.68

signal: 5 response time: 4.28

signal: 6 response time: 5.2

signal: 7 response time: 8.4

signal: 8 response time: 9.0

signal: 9 response time: 9.68

signal: 10 response time: 10.2

signal: 11 response time: 19.36

signal: 12 response time: 19.8

signal: 13 response time: 20.32

signal: 14 response time: 29.400000000000002

signal: 15 response time: 29.759999999999998

signal: 16 response time: 30.279999999999998

(2)

```python
import numpy as np
import math
signal_num = 0
one_bit_trans = 0
total_qi = 0
for idx, line in enumerate(open("input.dat", 'r')):
    item = line.rstrip()
    split_item = item.split()
    if idx == 0:
        signal_num = int(split_item[0])
        trans_time = np.zeros(signal_num)
        period_time = np.zeros(signal_num)
    elif idx == 1:
        one_bit_trans = float(split_item[0])
    else:
        trans_time[int(split_item[0])] = float(split_item[1])
        period_time[int(split_item[0])] = float(split_item[2])
```

```
for i in range(signal_num):
    block_time = np.max(trans_time[i:])
    high_priority_signal = trans_time[:i]
    LHS = block_time
    while 1:
        RHS = block_time
        for j in range(len(high_priority_signal)):
            RHS += math.ceil((one_bit_trans +
LHS)/period_time[j])*high_priority_signal[j]
        if RHS == LHS:
            print("signal: %s response time: %s"%(i, (RHS +
trans_time[i])))
            break
        elif RHS >= LHS:
            LHS = RHS
        else:
            print("error in message %s"% (i))
            break
```