## 1 Timing Analysis of the CAN Protocol — Part I (20pts)

Given a set of periodic messages $\mu_0, \mu_1, \mu_2$ with their priorities, transmission times, and periods as follows:

| Message | Priority ($P_i$) | Transmission Time ($C_i$) (msec) | Period ($T_i$) (msec) |
|---|---|---|---|
| $\mu_0$ | 0 | 10 | 50 |
| $\mu_1$ | 1 | 30 | 200 |
| $\mu_2$ | 2 | 20 | 100 |

The worst-case response time $R_i$ of $\mu_i$ can be computed as

$$R_i = Q_i + C_i, \tag{1}$$

and

$$Q_i = B_i + \sum_{\forall j, P_j < P_i} \left\lceil \frac{Q_i + \tau}{T_j} \right\rceil C_j, \tag{2}$$

where $\tau = 0.1$ in this question. You can consider using the following tables to help you.

1. (4pts) What is the worst-case response time of $\mu_0$? 30+10=40

| Iteration | LHS ($Q_0$) | $B_0$ | RHS | Stop? |
|---|---|---|---|---|
| 1 | 30 | 30 | 30 | Yes |

2. (8pts) What is the worst-case response time of $\mu_1$? 40+30=70

| Iteration | LHS ($Q_1$) | $B_1$ | $j$ | $Q_1 + \tau$ | $T_j$ | $\left\lceil \frac{Q_1+\tau}{T_j} \right\rceil$ | $C_j$ | RHS | Stop? |
|---|---|---|---|---|---|---|---|---|---|
| 1 | 30 | 30 | 0 | 30.1 | 50 | 1 | 10 | 40 | No |
| 2 | 40 | 30 | 0 | 40.1 | 50 | 1 | 10 | 40 | Yes |
| 3 | | | 0 | | | | | | |

3. (8pts) What is the worst-case response time of $\mu_2$? 70+20=90

| Iteration | LHS ($Q_2$) | $B_2$ | $j$ | $Q_2 + \tau$ | $T_j$ | $\left\lceil \frac{Q_2+\tau}{T_j} \right\rceil$ | $C_j$ | RHS | Stop? |
|---|---|---|---|---|---|---|---|---|---|
| 1 | 20 | 20 | 0 | 20.1 | 50 | 1 | 10 | 60 | No |
| | | | 1 | | 200 | 1 | 30 | | |
| 2 | 60 | 20 | 0 | 60.1 | 50 | 2 | 10 | 70 | No |
| | | | 1 | | 200 | 1 | 30 | | |
| 3 | 70 | 20 | 0 | 70.1 | 50 | 2 | 10 | 70 | Yes |
| | | | 1 | | 200 | 1 | 30 | | |

```
0: 1.44
1: 2.04
2: 2.56
3: 3.16
4: 3.68
5: 4.28
6: 5.2
7: 8.4
8: 9.0
9: 9.68
10: 10.2
11: 19.360000000000003
12: 19.8
13: 20.32
14: 29.400000000000002
15: 29.76
16: 30.28
```

```python
'''
Read in data and pruning
'''
with open("input.dat") as f:
    data = f.read().split('\n')[:-1]
    n = int(data[0])
    tow = float(data[1])
    data = data[2:]
    P = []
    C = []
    T = []
    for i in data:
        tmp = i.split(' ')
        tmp2 = []
        for j in tmp:
            if len(j) != 0:
                tmp2.append(float(j))
        P.append(tmp2[0])
        C.append(tmp2[1])
        T.append(tmp2[2])
'''
Computing worst-case response time
'''
for i in range(n):
    B = max(C[i:])
    Q = B
    while True:
        counting = 0.0
        for j in range(i):
            counting += math.ceil((Q+tow)/T[j])*C[j]
        if B+counting+C[i] > T[i]:
            print("not schedulable")
            break
        elif B+counting == Q:
            print(str(i)+":", Q+C[i])
            break
        else:
            Q = B+counting
```

## 3 Timing Analysis of TDMA-Based Protocols (16pts)

Following the assumptions (each time slot has the same length, each time slot serves exactly one frame, and a frame is transmitted only if the whole time slot is available) in the lecture, please compute the worst-case response time of the "asynchronous" message with the frame arrival pattern $(4, 10, 0, 3, 5, 6)$ and the schedule pattern $(2, 5, 1, 2)$ by completing the following steps.

1. (2pts) Please duplicate the schedule pattern (hint: $(4, 10, 1, 2, \ldots)$). No intermediate work is needed here. (4, 10, 1, 2, 6, 7)
2. (2pts) Please duplicate the arriving times of frames in the frame arrival pattern but fix $m = 4$ and $p = 10$. No intermediate work is needed here. (4, 10, 0, 3, 5, 6, 10, 13, 15, 16)
3. (2pts) Please duplicate the starting times of time slots in the schedule pattern but fix $n = 4$ and $q = 10$. No intermediate work is needed here. (4, 10, 1, 2, 6, 7, 11, 12, 16, 17)
4. (8pts) Please complete the following table:

| $k$ | $\max_{1\leq j\leq n}(s_{j+k} - s_j)$ | = | $\min_{1\leq i\leq m}(a_{i+k-1} - a_i)$ | = | (Column-3) − (Column-5) |
|---|---|---|---|---|---|
| 1 | $\max_{1\leq j\leq 4}(s_{j+1} - s_j)$ | 4 | $\min_{1\leq i\leq 4}(a_i - a_i)$ | 0 | 4 |
| 2 | $\max_{1\leq j\leq 4}(s_{j+2} - s_j)$ | 5 | $\min_{1\leq i\leq 4}(a_{i+1} - a_i)$ | 1 | 4 |
| 3 | $\max_{1\leq j\leq 4}(s_{j+3} - s_j)$ | 9 | $\min_{1\leq i\leq 4}(a_{i+2} - a_i)$ | 3 | 6 |
| 4 | $\max_{1\leq j\leq 4}(s_{j+4} - s_j)$ | 10 | $\min_{1\leq i\leq 4}(a_{i+3} - a_i)$ | 6 | 4 |

5. (2pts) Please compute the worst-case response time (which is waiting time plus transmission time) of the message. max -> 6
6+1 = 7