2019-04-11-extra questions

Real-time Systems (Technische Universiteit Delft)

Delft University of Technology
Faculty of Electrical Engineering, Mathematics, and Computer Science (EEMCS)

IN4343 – Real Time Systems

April 11th 2019, from 09:00 to 12:00

Mitra Nasri

Question:	1	2	3	4	Total
Points:	6	15	5	2	28
Score:					

- This is a closed book exam
- You may use a **simple** calculator only (i.e. graphical calculators are not permitted)
- Write your answers with a black or blue pen, not with a pencil
- **Always justify your answers, unless stated otherwise**
- Question 8 is an **optional question (bonus).**

- The exam covers the following material:
  (a) chapters 1-9 of the book "Hard Real-Time Computing Systems (3rd ed)" by G. Buttazzo
  (b) the paper "The Worst-Case Execution-Time Problem" by Wilhelm et al. (except Section 6)
  (c) the paper "Non-Work-Conserving Non-preemptive Scheduling: Motivations, Challenges, and Potential Solutions" by Mitra Nasri and Gerhard Fohler
  (d) the paper "An Exact and Sustainable Analysis of Non-Preemptive Scheduling" by Mitra Nasri and Björn B. Brandenburg
  (e) the paper "Best-case response times and jitter analysis of real-time tasks" by R.J. Bril, E.F.M. Steffens, and W.F.J. Verhaegh

Liu and Layland (LL) bound	$U_{lub}^{RM} = n(2^{1/n} - 1)$
Hyperbolic (HB) bound	$\prod_{i=1}^{n}(U_i + 1) \leq 2$
Response Time Analysis	$WR_i = C_i + \sum_{k=1}^{i-1} \left\lceil \frac{WR_i + AJ_k}{T_k} \right\rceil C_k$ $BR_i = C_i + \sum_{k=1}^{i-1} \left(\left\lceil \frac{BR_i - AJ_k}{T_k} \right\rceil - 1 \right)^+ C_k$ $w^+ = \max(w, 0)$
Processor Demand	$g(t_1, t_2) = \sum_{r_i \geq t_1}^{d_i \leq t_2} C_i \qquad g(0, L) = \sum_{i=1}^{n} \left\lfloor \frac{L + T_i - D_i}{T_i} \right\rfloor C_i$
schedulability	$\forall L \in D, \quad g(0, L) \leq L$ $D = \{d_k \mid d_k \leq \min(H, \max(D_{max}, L^*))\}$ $H = lcm(T_1, \ldots, T_n)$ $L^* = \frac{\sum_{i=1}^{n}(T_i - D_i) \cdot U_i}{1 - U}$
CBS Server arrival of a job J_k	**if** ($\exists$ a pending aperiodic job) **then** enqueue J_k; **else if** $(q_s \geq (d_s - a_k) \cdot U_s)$ **then** $\{\ q_s \leftarrow Q_s; \quad d_s \leftarrow d_s + T_s;\ \}$
budget exhaustion	$q_s \leftarrow Q_s$ $d_s \leftarrow d_s + T_s$
NP scheduling response time NP-FP	pre-requisites: $D \leq T$ and preemptive-schedulable $WO_i^{(n)} = B_i + \sum_{k=1}^{i-1} \left(\left\lfloor \frac{WO_i^{(n-1)}}{T_k} \right\rfloor + 1 \right) \cdot C_k$ $B_i = \max\{C_j \mid \forall \tau_j, P_i < P_j\}$ $R_i = C_i + WO_i$ $WO_i^{(0)} = B_i + \sum_{k=1}^{i-1} C_k$
response time NP-EDF	$\forall \tau_i, 1 < i \leq n, \forall L, T_1 < L < T_i$: $L \geq C_i + \sum_{k=1}^{i-1} \left\lfloor \frac{L - 1}{T_k} \right\rfloor \cdot C_k$

Question 1 [6 points]

Consider the following code in the context of dynamic techniques to estimate the worst-case execution time of a task on a given platform.

```
int foo (int x, int y) {
   int z = 0;
   if (x > 7 && y < 5) {
      z = x;
   }
   else {
      if (x > 4 && y > 0){
         z = 2 * x + y;
      }
   }
   return z;
}
```

(a) [3 points] Generate the minimum number of test cases (i.e., calls to the foo function with different inputs) such that the statement coverage is satisfied.

(b) [3 points] Generate the minimum number of test cases such that the branch coverage is satisfied.

Question 2 [15 points]

A rate-monotonic scheduler is used to run the following task set [1].

	C_i	T_i	D_i
τ_1	3	6	6
τ_2	7	18	18
τ_3	8	24	24

(a) [10 points] Compute the WCRT (worst-case response time) of each task.

(b) [5 points] Determine the worst-case response times of the three tasks. *Hint: draw a time line.*

Question 3 [5 points]

Park's test may be used with fixed-priority schedules to verify that the workload to be completed before a task's deadline is smaller than the deadline of the task itself. The test condition is formalized as[2]

$$C_i + \sum_{k=1}^{i-1} \lceil \frac{D_i}{T_k} \rceil \cdot C_k \leq D_i$$

[1]Thanks to Charles Randolph for preparing this numerical example.
[2]Thanks to Charles Randolph for his suggestion.

(a) $\boxed{\text{5 points}}$ Is Park's test a *sufficient*, *necessary*, or *exact* test? Justify your answer.

Question 4 [2 points]

(a) $\boxed{\text{1 point}}$ **Claim**: In general, preemptive fixed-priority scheduling with rate-monotonic priorities has a larger number of preemptions than EDF.
Cause: because each time a high-priority task is released, it will certainly preempt a low-priority task.

(a) claim is true, cause is true (b) claim is true, cause is false (c) claim is false

(b) $\boxed{\text{1 point}}$ Which resource access policy may result in deadlock?

(a) Priority Ceiling Protocol (PCP) (b) Highest-Locker Priority (HLP)
(c) Priority Inheritance Protocol (PIP) (d) Non-Preemptive Protocol (NPP)