ECE - Computer Networks II
Winter 2007

Problem Set #1

To be discussed on: January 25, 2007.

Problem 1. Consider a single CPU computer system with N processes. The operations of a process are triggered by input signals that arrive from an input device (e.g., a sensor). There is one input device for each process. There is no contention for accessing the devices, and reading the input does not involve the CPU. The device for Process m writes input signals into a FIFO buffer that is accessed by Process m. Process m removes an input signal from its FIFO buffer, processes the signal, and then removes the next signal. The process blocks if it tries to remove a signal and its FIFO buffer is empty. The processing of a signal cannot be interrupted. The CPU selects a signal from one of the buffers only after completing the processing of a signal, or when a signal arrives to an empty system. There are two types of signals: Type-1 and Type-2. The CPU processing time required by each process is 1 msec for a Type-1 signal, and 2 msec for a Type-2 signal. The type and the arrival times of signals cannot be predicted.

Let $a_i(t)$ denote the workload due to the arrival of a signal for Process i. The arrival of workload is measured in the time needed to process a signal (in milliseconds). We have $a_i(t) = 1$, if there is an Type-1 arrival for Process i at time t. We have $a_i(t) = 2$, if there is a Type-2 arrival for Process i at time t. And we have $a_i(t) = 0$ if there is no arrival for Process i at time t. Assume that $N = 3$ and that the system is idle at times $t < 0$. Consider the following arrival scenario:

$$
\begin{aligned}
&a_1(0) = 1, \quad a_1(5) = 2, \quad a_1(9) = 2,\\
&a_2(0) = 2, \quad a_2(3) = 1, \quad a_2(10) = 1,\\
&a_3(0) = 1, \quad a_3(4) = 1.
\end{aligned}
$$

At any time, we can determine the unfinished work in the system in terms of the time it takes to complete the processing of all signals currently in the system. If there is no unfinished work, the system is said to be idle.

(a) In the time interval [0, 12], when is the system idle?

(b) The completion time of a signal is the time difference between the arrival of the signal and the time when processing of the signal is completed. What is the longest completion time for any signal in the given arrival scenario?

(c) Suppose that Process 1 has highest priority, in the sense that Process 1 always gets the CPU if there is a signal waiting in its FIFO buffer. Process 2 has the second highest priority, and Process 3 has the lowest priority. In the above scenario, what are the departure times for each of the arrivals?

(d) Assume that the processing of a Type-1 signal must be finished within 2 msec after its arrival, a Type-2 signal after 4 msec after its arrival. The system should trigger an alarm, if it detects that it cannot complete the processing of all signals before their targeted finishing times. Devise an algorithm that can trigger the alarm as early as possible (i.e., before a finishing time is missed). The simpler your algorithm, the better.

Problem 2. Given the functions

$$f(t) = \begin{cases} 0 & \text{if } t < 0 \\ t+3 & \text{if } t \geq 0 \end{cases}$$

$$g(t) = \begin{cases} 0 & \text{if } t < 0 \\ 2t+1 & \text{if } t \geq 0 \end{cases}$$

Compute the *convolution* of $f * g$.

Problem 3.

Given the functions $A(t) = \sigma + \rho t$ (for some fixed values of $\rho > 0$ and $\sigma \geq 0$) and $S(t) = C \cdot t$ (for a constant $C > \rho$).

(a) Sketch the function $A * S$.

(b) Use Reich's equation to show that the queue length at any time t is bounded by σ.

Problem 4. Prove the following properties for causal processes A, B, and C:

(a) *Commutativity of* $*$*:* $A * B = B * A$.

(b) *Associativity of* $*$*:* $(A * B) * C = A * (B * C)$.

(c) *Distributivity of* $*$ *over* $\wedge$*:* $A * (B \wedge C) = A * B \wedge B * C$.

(d)

(e) $A * A \leq A$.

(f) *Composition of* $\oslash$*:* $(A \oslash B) \oslash C = A \oslash (B * C)$.

(g) *Composition of* $\oslash$ *and* $*$*:* $(A * B) \oslash C \leq A * (B \oslash C)$.

(h) *Duality between* $\oslash$ *and* $*$*:* $A \oslash B \leq C$ if and only if $f \leq g * h$.