CSE237a Final Exam
Winter 2016
Prof. Tajana Simunic Rosing

Name: ____________________________
PID: ____________________________

<table>
<thead>
<tr>
<th>Problem</th>
<th>Max points</th>
<th>Points</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>20</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>15</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>15</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>15</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>25</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>20</td>
<td></td>
</tr>
<tr>
<td>Total</td>
<td>110</td>
<td></td>
</tr>
</tbody>
</table>

Instructions
1. Please make sure your writing is clear and readable.
2. Show your work.
3. This exam is open book and open notes. No electronics are allowed.
4. Please sign the statement below before you begin:

Honor code
By signing my name below I hereby certify that I have neither given, nor received assistance in completing this examination:

____________________________________________________

March 10, 2016
### Problem 1: Short questions [20 pts]

For each statement, please indicate whether it is True (T) or False (F). [1 pt each]

<table>
<thead>
<tr>
<th></th>
<th>Statement</th>
<th>Answer</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>(Example) I have completed my CSE 237A final project</td>
<td>T</td>
</tr>
<tr>
<td>1</td>
<td>The I^2C protocol achieves low physical communication overhead as a parallel bus.</td>
<td>F</td>
</tr>
<tr>
<td>2</td>
<td>EDF scheduling is optimal on a single processor and with independent tasks.</td>
<td>T</td>
</tr>
<tr>
<td>3</td>
<td>Unlike Lamport time, vector time provides total ordering of a distributed system.</td>
<td>F</td>
</tr>
<tr>
<td>4</td>
<td>Kahn Process network communication is handled through bounded FIFO channels.</td>
<td>F</td>
</tr>
<tr>
<td>5</td>
<td>Delta delays cause nondeterminism when modeling asynchronous elements in Verilog.</td>
<td>F</td>
</tr>
<tr>
<td>6</td>
<td>The base Android task scheduler is a completely fair scheduler.</td>
<td>T</td>
</tr>
<tr>
<td>7</td>
<td>DRAM’s energy consumption per bit is less than that of SRAM.</td>
<td>F</td>
</tr>
<tr>
<td>8</td>
<td>Unlike StateCharts, SDL uses message passing.</td>
<td>T</td>
</tr>
<tr>
<td>9</td>
<td>EDF and RM belong in the class of static scheduling algorithms.</td>
<td>F</td>
</tr>
<tr>
<td>10</td>
<td>The pthread implementation is a POSIX-compliant multithreading implementation for real-time operating systems (RTOS)</td>
<td>T</td>
</tr>
</tbody>
</table>

For each of the following questions, circle the correct option.

1. Under vector time, what is the timestamp of event X of process P2? [2 pts]

   ![Diagram](image)

   a. [3, 3, 2]
   b. [3, 3, 2]
   c. [3, 0, 2]
   d. [0, 3, 0]
2. Given the following Esterel code and sequence of input signals run on this module, what is the sequence of outputs (if any) observed at end of the timeline? [3 pts]

*Note: you do not have to account for timing in your answers, just the sequence of output*

```esterel
module exam:
  input A, B, C;
  output M, N;

  loop
    [await A || await B];
    emit M
    every A do emit N end
  each C
end module
```

a. N, M, M, N, N  
b. N, M, N, M, N, N  
c. M, N, M, N, N  
d. M, M, N, N  
e. M, N, M, N, N, N

3. Select the lowest sampling frequency for the following sensing platform architecture [5 pts]:

- **Maximum clock speed**: 100 kHz
- **Maximum processor speed**: 50 kHz
- **Input signal processing WCET**: 100 processor cycles
- **Input signal frequency**: 100 Hz

a. 50 Hz  
b. 100 Hz  
c. 200 Hz  
d. 500 Hz  
e. 1 kHz
Problem 2: Petri Nets [15 pts]
Given the following Petri Net specification:
\[ P = \{ P_1, P_2, P_3 \} \]
\[ T = \{ t_1, t_2, t_3 \} \]
\[ F = \{ (P_1, t_1), (P_2, t_2), (P_2, t_3), (P_3, t_3), (t_1, P_2), (t_2, P_2), (t_3, P_3) \} \]

a. Draw the Petri Net. [3 pts]

b. Given the initial marking: \([2, 0, 1]\)
   i. Draw the reachability graph. [5 pts]

   \[
   \begin{align*}
   &t_2 &\rightarrow & [2, 0, 1] \\
   &t_1 &\rightarrow & [1, 1, 1] \\
   &t_3 &\rightarrow & [0, 0, 1] \\
   \end{align*}
   \]
   ii. What is the liveness of the Petri Net? [2 pts]

   L1

   c. Now consider changing the weight of \{t_2, P_1\}. What is the liveness of this Petri Net given the same initial marking as part b? [5 pts]
   Note: You must justify your answer for credit.

   L3. Here is a segment of the new reach. graph with an infinite loop

   \[
   \begin{align*}
   & [2, 1, 1] \\
   & \downarrow \quad \uparrow \quad \downarrow \\
   & t_3 \quad t_2 \quad t_1 \\
   & [2, 1, 1] \quad [1, 1, 1] \\
   & \uparrow \quad \downarrow \\
   & t_1 \quad t_2 \\
   & [3, 0, 1] \\
   & \downarrow \\
   \end{align*}
   \]
Problem 3: State charts and Finite State Machines [15 pts]
Given the following StateChart, draw the equivalent Finite State Machine (FSM):
Problem 4: Synchronous data flow [15 pts]

Given the following SDF:

a. Write the incidence matrix. Use columns to represent the nodes, and rows to represent each edge. [2 pts]

<table>
<thead>
<tr>
<th></th>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>-2</td>
<td>3</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>6</td>
<td>-9</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>3</td>
<td>0</td>
<td>-6</td>
<td>0</td>
<td>2</td>
</tr>
<tr>
<td>4</td>
<td>0</td>
<td>-9</td>
<td>0</td>
<td>3</td>
</tr>
<tr>
<td>5</td>
<td>-1</td>
<td>0</td>
<td>2</td>
<td>0</td>
</tr>
<tr>
<td>6</td>
<td>2</td>
<td>0</td>
<td>-4</td>
<td>0</td>
</tr>
<tr>
<td>7</td>
<td>0</td>
<td>0</td>
<td>-k1</td>
<td>2</td>
</tr>
<tr>
<td>8</td>
<td>0</td>
<td>0</td>
<td>12</td>
<td>-k2</td>
</tr>
</tbody>
</table>

b. Solve for $k_1$ and $k_2$ such that a minimum PASS schedule can be created. [5 pts]

\[
\begin{align*}
A &= 2C; 2A = 3B; 3B = D; k_1C = 2D; 12C = k_2D \\
12C &= 6A = 9B = 3D \\
k_1 &= 8 \\
k_2 &= 3
\end{align*}
\]

c. Show a PASS schedule given the following ordering of tasks: D → C → A → B [5 pts]

12D 3C 6A 4B

d. In SDF, each edge is associated with a buffer: Find the initial elements in each buffer for your PASS schedule. [3 pts]

a=0, b=0, c=24, d=36, e=0, f=0, g=24, h=0
Problem 5: Scheduling [25 pts]
A system runs three periodic processes, where each process’s period is its priority.

<table>
<thead>
<tr>
<th>Task</th>
<th>Worst case execution time (WCET)</th>
<th>Period</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>1</td>
<td>3</td>
</tr>
<tr>
<td>B</td>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td>C</td>
<td>1</td>
<td>x</td>
</tr>
</tbody>
</table>

a. Is there a period for task C that fulfills the rate-monotonic (RM) inequality? What does this mean for this system? [5 pts]

*Note: show all your work to get credit.*

*Hint: $2^{1/2} \approx 1.414; 2^{1/3} \approx 1.260; 2^{1/4} \approx 1.189; 2^{1/5} \approx 1.149$*

No. This means that an RM schedule *may* not exist (but is not provably true)

$$\frac{1}{3} + \frac{2}{4} + \frac{1}{x} \leq n \times \left(2^{\frac{1}{n}} - 1\right) = 3 \times 0.26 = 0.78$$

$$\frac{1}{x} \leq -0.05$$

b. Construct a static schedule for this system in the order of task priority. What is the *minimum* period of C such that it is schedulable? Assume that all tasks arrive at t=0. [10 pts]

Task C period: **8**

<table>
<thead>
<tr>
<th>time</th>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
<th>11</th>
<th>12</th>
<th>13</th>
<th>14</th>
</tr>
</thead>
<tbody>
<tr>
<td>task</td>
<td>A</td>
<td>B</td>
<td>B</td>
<td>A</td>
<td>B</td>
<td>B</td>
<td>A</td>
<td>C</td>
<td>B</td>
<td>A</td>
<td>B</td>
<td>C</td>
<td>A</td>
<td>B</td>
<td>B</td>
</tr>
</tbody>
</table>

time 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29
| task | A  | B  | B  | A  | C  | B  | A  | B  | A  | B  | A  | B  | A  | B  | B  |

  c. Construct an earliest-deadline first (EDF) schedule for this system. What is the *minimum* period of C such that it is schedulable? Assume that all tasks arrive at t=0. [10 pts]

Task C period: **6**

<table>
<thead>
<tr>
<th>time</th>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
<th>11</th>
<th>12</th>
<th>13</th>
<th>14</th>
</tr>
</thead>
<tbody>
<tr>
<td>task</td>
<td>A</td>
<td>B</td>
<td>B</td>
<td>C</td>
<td>A</td>
<td>B</td>
<td>A</td>
<td>C</td>
<td>A</td>
<td>B</td>
<td>B</td>
<td>A</td>
<td>C</td>
<td>B</td>
<td></td>
</tr>
</tbody>
</table>

time 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29
| task | B  | A  | B  | B  | C  | A  | B  | B  | A  | C  | A  | B  | B  | A  | C  |
Problem 6: Task graphs [20 pts]

Consider the following sensor application:

i. Microphone samples main sound, input 1, and background sound, input 2. These must be done independently, and input 1 must be sampled first.

ii. An independent ambient noise sensor samples input 3, simultaneously to input 1.

iii. Input 4 is independently received from the mixing board, to modulate output. This value is always available when needed.

iv. The value of the echo sensor is sampled simultaneously with the background sound, input 2, to provide input 5.

v. Inputs 1 and 2 are processed to create the output sound, output 1.

vi. Inputs 2 and 3 are processed to determine output volume, output 2.

vii. Inputs 3 and 4 are processed to modulate output 2 into a transient signal $t$.

viii. Transient signal $t$ is further processed with the echo sensor to produce output 3.

Assume that processing can only handle 2 inputs at a time, and sensing and processing are completely independent tasks. All inputs and intermediate values are available in the interval after they have been sampled.

a. Draw the task graph for the above problem, with each circle representing an action (the appropriate sensing or processing task) and labeling inputs, intermediate data and outputs. Assume any simultaneous processing can be done in parallel. [5 pts]
b. For the above task graph, create an ALAP processing schedule with no restrictions on processing (multiple processing instances can happen in parallel). Make sure that inputs and intermediate values are only available at the interval at which they are generated. [5 pts]

c. You are switching to low-capability processors that can no longer perform simultaneous operations in parallel, but you have two different such processors available. Steps v and vi above can only run on processor A, but steps vii and viii can run on either processor. Construct the most time-efficient LIST schedule for the above tasks. [10 pts]