

Ingeniería de Software 1

Práctica 5 – Máquinas de estado

Ejercicio 1

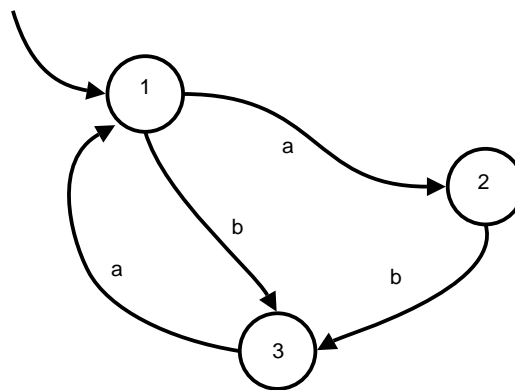
Definimos las FSMs:

1. $FSM_A = \langle \{1, 2, 3\}; \{a, d, c\}; \{(1, a, 2), (2, d, 3), (3, c, 1)\}; 1 \rangle$
2. $FSM_{Leer} = \langle \{ \text{"Luz apagada"}, \text{"Luz encendida"}, \text{"Vigilando"} \}; \{ \text{"Subir interruptor"}, \text{"Bajar interruptor"}, \text{"Activar sensor infrarrojo"}, \text{"Desactivar sensor infrarrojo"} \}; \{ (\text{"Luz apagada"}, \text{"Subir interruptor"}, \text{"Luz prendida"}), (\text{"Luz prendida"}, \text{"Bajar interruptor"}, \text{"Luz apagada"}), (\text{"Vigilando"}, \text{"Desactivar sensor infrarrojo"}, \text{"Luz apagada"}), (\text{"Luz apagada"}, \text{"Activar sensor infrarrojo"}, \text{"Vigilando"}) \}; \text{"Luz apagada"} \rangle$

Realice una versión gráfica de las FSMs anteriores.

Ejercicio 2

Dada la siguiente FSM:

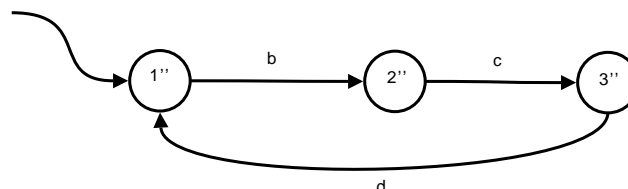


Describa al conjunto de trazas de la máquina. ¿Qué característica tiene este conjunto?

Ejercicio 3

Sean las siguientes FSMs representadas en distintas formas:

- $FSM_A = \langle \{1, 2, 3\}; \{a, d, c\}; \{(1, a, 2), (2, d, 3), (3, c, 1)\}; 1 \rangle$
- $FSM_B = \langle \{1', 2', 3'\}; \{a, b, c\}; \{(1', b, 2'), (2', a, 3'), (3', c, 1')\}; 1' \rangle$
- FSM_C dada por la siguiente figura:

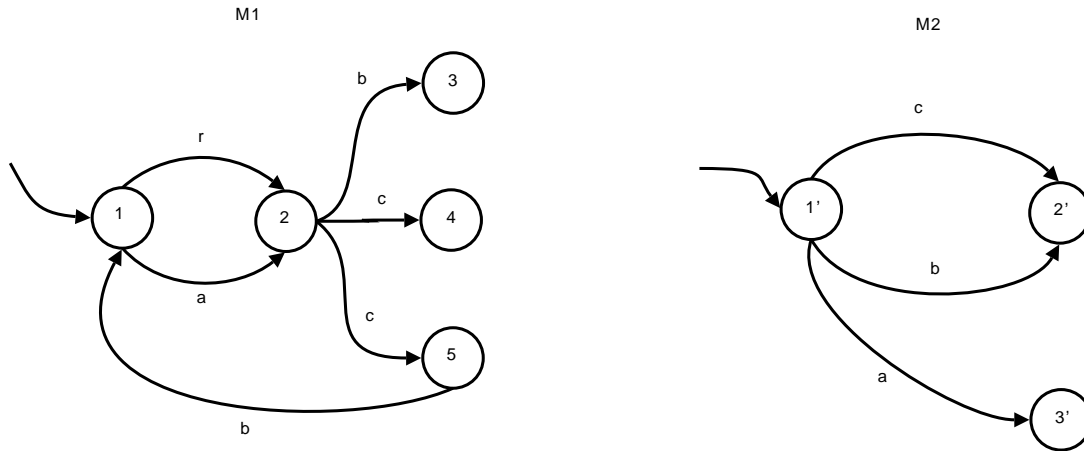


Calcule las siguientes composiciones:

- a) $A \parallel B$.
- b) $A \parallel C$. ¿Qué ocurre en este caso? Explique en sus propias palabras lo que está pasando y por qué cree que sucede.

Ejercicio 4

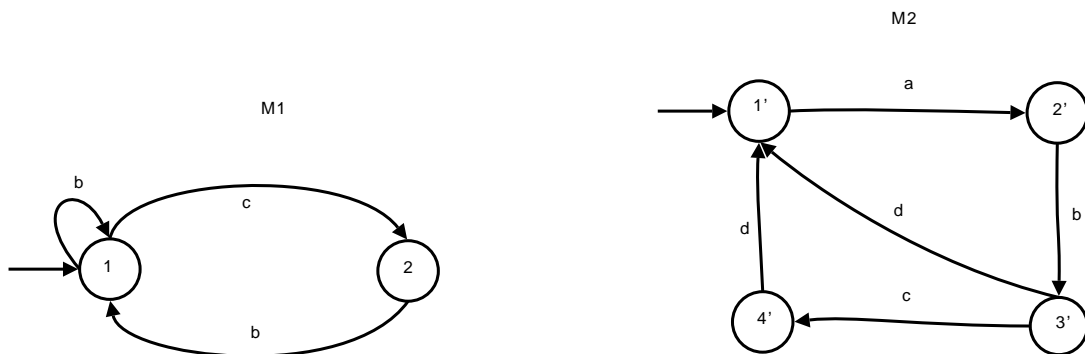
Dadas las siguientes FSMs:



- Calcule la FSM composición de ambas, $M_1 \parallel M_2$.
- Describa todas las trazas de ejecución de la composición. Justifique la completitud del conjunto exhibido.
- ¿Existe *deadlock* en la composición? Si es así, exhiba todas las trazas que culminan en deadlock.
- ¿Existe alguna traza de ejecución en la que *no* se alcance un *deadlock*?

Ejercicio 5

Sean las FSMs M_1 y M_2 definidas como en las siguientes figuras



- Realice las tareas pedidas en el ejercicio anterior. **Sugerencia:** utilice alguna notación clara para expresar las iteraciones.
- ¿Exhibe *deadlock* la composición? Si es así, proponga un agregado simple (es decir, sin quitar estados ni transiciones de las máquinas existentes) que lo resuelva. Justifique.

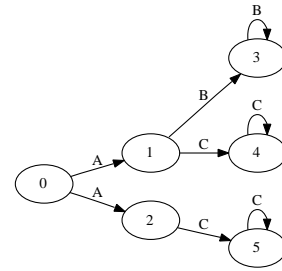
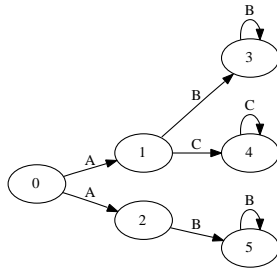
Ejercicio 6

Muestre que las siguientes FSMs *no* son bisimilares.



Ejercicio 7

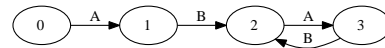
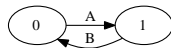
Sean M (izquierda) y M' (derecha) las siguientes FSMs:



1. Muestre que M *simula* a M' ($M' \preceq M$).
2. Muestre que M' *simula* a M ($M \preceq M'$).
3. Entonces... vale que $M' \preceq M$ y $M \preceq M'$, pero... ¿son M y M' *bisimilares*? Justifique.

Ejercicio 8

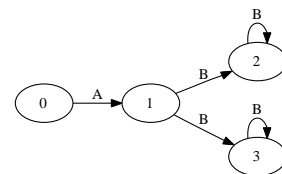
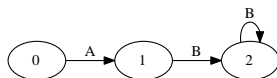
Sean las siguientes dos FSMs:



- Muestre que son bisimilares.
- Note que la FSM de la derecha resulta de “desovillar” (*unwind*) una vez el ciclo de la FSM de la izquierda. Muestre que cualquier ciclo *unwound* una cantidad finita de veces es bisimilar al ciclo original.

Ejercicio 9

Sean las siguientes dos FSMs:



- Muestre que son bisimilares.
- La FSM de la derecha es el resultado de duplicar exactamente sub-FSMs de la derecha. Muestre que cualquier duplicación finita es bisimilar a la máquina original.

Ejercicio 10

Supongamos que se quiere modelar el siguiente comportamiento del controlador de un horno microondas: el horno microondas posee una puerta. El horno sólo puede ponerse en marcha si la puerta está cerrada (el controlador puede detectar si la puerta está abierta o cerrada). Si se abre la puerta estando el horno prendido, el horno se apaga automáticamente. Por otra parte, en cualquier momento es posible establecer el modo de cocción. Los modos de cocción posibles son descongelar, calentar y *grill*.

- a) Modele el comportamiento anterior utilizando una única máquina de estados.
- b) Descomponga ahora su solución en tres partes: modele separadamente la puerta, el horno, y el modo de cocción. Verifique que la composición de sus nuevas máquinas sea equivalente a su solución del punto a).
- c) Utilice la extensión en la notación de ! y ? sobre las etiquetas compartidas de las máquinas. ¿Qué información agrega esta notación a su modelo? Describa esta semántica en el contexto del problema.

Ejercicio 11

Modele con FSMs el siguiente protocolo de un “Acelerador de Peticiones”: el acelerador realiza mediación de servicios entre un cliente y dos servidores. En un escenario normal, el acelerador recibe la petición del cliente y lo envía a ambos servidores. La primera respuesta que reciba es la que enviará al cliente, ya que los dos servidores otorgan el mismo servicio, contestan indefectiblemente y devuelven exactamente el mismo resultado, pero uno puede ser más rápido que el otro en distintas situaciones. El acelerador luego se encarga de la cancelación de la petición al otro servidor, para que no siga trabajando y asegurarse que no le responda. El cliente no puede hacer otra petición hasta que no reciba la respuesta del acelerador. El cliente nunca debe recibir una respuesta atrasada de una petición anterior. Suponga que no se pierden paquetes en ninguna comunicación.

Una vez especificado el comportamiento anterior, extienda el modelo a N servidores. ¿Es necesario cambiar algo?

Ejercicio 12

Se desea modelar el comportamiento de un proceso de tipo *pipe-and-filter*, desde el momento en que los datos ingresan al proceso, hasta que sale el resultado del mismo. Para transformar los datos iniciales en el resultado final, se suceden varios procesos intermedios *filters* que reciben el dato a procesar y devuelven un resultado intermedio. Cada uno de estos subprocesos es entonces el encargado de efectuar una porción de la transformación. Las entradas y salidas de los procesos están conectados *pipes* y la entrada de cada una es el resultado intermedio del proceso previo. Los datos crudos ingresan primero al subproceso 1, y luego el resultado del mismo es luego entrada para el proceso 2—en caso de existir—y así sucesivamente.

Existe la posibilidad que el resultado intermedio de algún subproceso se corrompa y no sirva como entrada al siguiente proceso. En caso de detectar un resultado inválido, debe ser descartado. No está dentro del alcance del modelado del sistema determinar los factores de detección.

La restricción más importante a satisfacer es que un subproceso k no puede recibir el resultado intermedio del proceso anterior ($k-1$) hasta tanto el proceso posterior ($k+1$) haya recibido el resultado generado por el mismo proceso k (en el caso de que k haya generado un resultado correcto), o bien se haya descartado el resultado de la máquina k , en el caso de ser un resultado inválido.

Notar que los procesos en sí funcionan en forma simultánea e independiente, es decir, dos procesos k y una k' deben poder procesar distintos resultados intermedios al mismo tiempo.

- Modele este comportamiento con FSMs, suponiendo que el proceso de *pipe-and-filter* consta de 4 subprocesos. Justifique.
- Explique claramente cómo debería ser el modelo si se generaliza a n procesos en lugar de 4.

Ejercicio 13

Se quiere realizar un sistema de auditoría para una cadena de restaurantes de comida rápida. Básicamente, cada sucursal de la cadena preparará, en los últimos días del mes, un informe detallado acerca de las ventas, stock, etc. El sistema en cuestión constará de un proceso en cada sucursal (encargado de hacer los informes locales), y un proceso centralizado encargado de recibir los reportes locales y de luego realizar el reporte general. Para la primera fase de puesta en marcha del proyecto, se plantea integrar sólo 3 sucursales al sistema.

Al llegar la hora de presentar los reportes, cada sucursal enviará sus datos al proceso central. Sin embargo, este proceso es bastante rudimentario y sólo soporta ser usado por una de las sucursales a la vez. A este fin, los procesos de las sucursales primero toman el control del proceso central, luego ingresan todos los datos, y finalmente liberan el control del proceso. Los gerentes de las sucursales no tienen comunicación entre sí, por lo que no pueden ponerse de acuerdo en un orden para el ingreso de los datos en el proceso central.

Una vez que el sistema de una sucursal ha subido los datos al proceso central, procede a imprimir el reporte local para la aprobación por parte del gerente. Esta fase no requiere interacción con el proceso centralizado. Asimismo, el proceso centralizado imprime un reporte general, pero recién cuando ha recibido los informes de *todas* las sucursales. Cuando una sucursal termina de imprimir su reporte, obtiene del proceso central el reporte global (una vez que el proceso central terminó de generarlo). Esta nueva comunicación con el proceso central tiene la misma restricción que las comunicaciones anteriores: sólo una de las sucursales puede comunicarse con el mismo a la vez.

Una vez que todas las sucursales han obtenido el reporte global, se encuentran en condiciones de comenzar a recabar datos para el siguiente mes, momento en el cual el proceso completo se repetirá.

- a) Modele el comportamiento de los sistemas de las sucursales y el proceso central, utilizando FSMs. No utilice variables para modelar la comunicación entre las sucursales y el proceso central.
- b) ¿Cómo extendería el modelo para contemplar n sucursales?

Ejercicio 14

Modele el proceso de lanzado de una moneda al aire. Primero suponga que el resultado de la tirada se resuelve una vez que la acción de lanzado terminó. Luego de realizar ese modelo, compárelo con el de suponer que el resultado es dependiente del lanzado, es decir, al momento que se lanza la moneda al aire, ya hay predeterminado un único resultado. ¿Qué diferencias encuentra entre ambos modelos? ¿Cuál le resulta más intuitivo respecto de la realidad a modelar? ¿Cuál más claro?

Muestre el conjunto de trazas para ambos modelos. ¿Qué relación hay entre ambos conjuntos?

Ejercicio 15

En una fábrica de juguetes se construyen soldaditos de madera utilizando dos robots. Cada soldadito se confecciona ensamblando dos piezas, y cada pieza es preparada por un robot distinto.

El robot R_1 levanta piezas de la cinta C_1 , y según las características de las mismas le aplica simultáneamente las herramientas H_1 y H_2 o solamente H_1 (en el primer caso, toma siempre primero H_1 y luego H_2 , las aplica a la pieza, y finalmente las suelta en idéntico orden). Luego de utilizar las herramientas, el robot R_1 interactúa en sincronía con el otro robot para ensamblar al soldadito y colocarlo en la cinta de salida. A su vez, el robot R_2 toma piezas de su propia cinta (C_2) y, análogamente al robot R_1 , aplica a la pieza las herramientas H_1 y H_2 simultáneamente (con la diferencia de que R_2 siempre toma (y luego suelta) H_2 antes que H_1), o bien aplica solamente H_2 . Los soldaditos en la cinta de salida son puestos en un depósito por un obrero para su posterior distribución.

Existe una restricción en cuanto al uso de las herramientas: cada una de ellas sólo puede ser usada por uno de los robots a la vez. Suponga, además, que el depósito es de tamaño infinito pero que en la cinta de salida sólo puede haber un máximo de 2 soldaditos.

Teniendo en cuenta las consideraciones antedichas:

- a) modele utilizando FSMs sin extensiones (ni guardas, ni variables) los brazos del robot, las herramientas, la cinta final y el obrero. Justifique cómo la composición en paralelo garantiza que las herramientas no sean utilizadas concurrentemente por los dos robots.
- b) Extienda las FSMs a autómatas temporizados para modelar el hecho de que cada herramienta tarda entre 4 y 6 segundos en realizar su función.

Ejercicio 16

En una herrería, un operario determina un tipo de pieza a fabricar, que puede ser una de tres posibles (A, B o C), y luego activa una alarma. Esta alarma es monitoreada por 8 robots que, inmediatamente, reciben en su área de trabajo una pieza a temperatura ambiente (22°C), que deben trabajar. Cada robot trabaja independientemente de los demás. Los robots sólo pueden calentar a la pieza, lo que provoca que cada vez que realicen esta acción, la temperatura de la pieza aumente 100°C . Si, al cabo de cinco minutos, los robots no recalientan la pieza, la temperatura de ésta disminuye 100°C . Recién cuando los 8 robots logran tener sus piezas respectivas a una temperatura superior a 200°C de manera simultánea, todos a la vez dejan su pieza en una cinta transportadora. La temperatura de la pieza trabajada nunca disminuye de la temperatura ambiente ni excede los 200°C adicionales. Existe una excepción a esta regla: 2 de estos 8 robots pueden trabajar sus piezas hasta la temperatura adicional de 300°C , y pueden dejar sus piezas trabajadas en la cinta transportadora cuando sus piezas se calientan tanto con los 200°C adicionales o 300°C adicionales.

Las 8 piezas son conducidas por la cinta transportadora al robot que realiza la construcción final, que a partir de ellas construye el tipo de pieza original determinada (A, B o C), tras lo cual el proceso vuelve a repetirse.

Describe, utilizando sistema de transiciones etiquetadas sin metavariables (salvo en los casos de temporizaciones), el comportamiento del proceso completo.

Ejercicio 17

Modele mediante un autómata temporizado un semáforo con interruptor que presente las siguientes características:

- La luz verde dura 30 segundos.
- La luz amarilla dura 5 segundos.
- La luz roja dura a lo sumo 30 segundos.
- Si la luz roja ha estado prendida por más de 20 segundos y se oprime el interruptor, no se produce ningún efecto.
- Si la luz roja ha estado prendida a lo sumo 20 segundos y se oprime el interruptor, la luz cambia a verde en 10 segundos más.
- Si el interruptor es oprimido cuando la luz verde o amarilla está encendida, el semáforo lo ignora.

Ejercicio 18

Se desea modelar con FSMs cierto teléfono celular (bastante simple). El teléfono posee un teclado, un display numérico (que puede mostrar hasta 8 dígitos) y una tapa que cubre el teclado. El teclado contiene los botones 0 al 9, *on*, *send*, *end*, *clear* y \leftarrow . Se enciende presionando el botón *on*. La tecla *clear* borra todo el número ingresado y \leftarrow borra de un dígito a la vez. Una vez que se ingresó el número deseado, presionando *send* se inicia la conversación, la cual se mantiene hasta que se cierre la tapa, se presione *end* o se apague el teléfono presionando nuevamente *on*. Al abrir la tapa, el número del display se revierte al que figuraba al momento de cerrarla o apagar el teléfono.

- a) Intente modelar al teléfono sin utilizar variables. ¿Qué problema encuentra?
- b) Modele el teléfono celular sin restricciones utilizando variables.

Elabore libremente cualquier aspecto del funcionamiento de este aparato que no esté explicado en el enunciado, documentando claramente lo que supone.

Ejercicio 19

Se desea modelar el sistema para una torre de control de un aeropuerto, que controle automáticamente el tráfico aéreo según las comunicaciones que tenga con los aviones circundantes. Cada vez que

un avión desea despegar o aterrizar, debe solicitar permiso a la torre de control la cual, en función de las lecturas del radar, puede dar la autorización o no. La autorización es otorgada sólo en el caso de que no haya ningún otro avión volando dentro del espacio aéreo del aeropuerto, ni otros aviones en proceso de despegar o aterrizar. Mientras que la autorización no sea otorgada, el avión seguirá pidiendo permiso hasta que lo consiga.

La torre cataloga la posición de los aviones según estén *fuera* o *dentro* de su zona de influencia, *despegando* o *aterrizando*. Siempre un avión que despega pasa a estar inicialmente *dentro* de la zona de influencia (para en algún momento salir de ella). Análogamente, un avión puede pasar a estar *aterrizando* sólo desde *dentro* de la zona de influencia.

En el momento en que un avión es autorizado para despegar o aterrizar, la torre de control envía una señal a todos los aviones ordenándoles no ingresar en el espacio aéreo del aeropuerto, a fin de evitar accidentes. De manera similar, al finalizar el despegue o aterrizaje en cuestión, la torre levanta la prohibición para volar sobre el espacio aéreo del aeropuerto enviándole a los aviones la señal correspondiente.

1. Modele la operatoria del sistema de la torre de control mediante la técnica de FSMs.
2. Muestre que el modelo ilustra el hecho de que es el sistema de la torre de control quien tiene la capacidad de otorgar las autorizaciones para las distintas maniobras (despegue, aterrizaje, etc.).
3. La torre de control indica a los aviones que no entren a la zona de influencia al permitir a un avión aterrizar o despegar. Muestre que en su modelo *no sucede* que esta situación se viole.

Ejercicio 20

Nos interesa modelar el comportamiento de las subastas virtuales organizadas por Vendotutti, una prestigiosa casa de subastas. Estas subastas en realidad son organizadas en un sistema similar a una sala de chat, lo cual posibilita crear “compradores virtuales”. De la misma manera, el “subastador” en sí también es virtual. A cierta hora, el *bot* subastador activa la sala para comenzar con las ventas. Las personas que desean comprar artículos a la venta, configuran uno de los *bots* compradores a tal efecto, y les indican conectarse a la sala virtual. Los *bots* pueden configurarse para indicar exactamente *qué* artículos ofertar (indicando su número de orden en la subasta), y también determinar un monto máximo a ofertar para cada uno de estos artículos. No pueden conectarse *bots* a la sala virtual una vez que la subasta ha comenzado.

Una vez que la sala virtual de subastas está abierta, el *bot* subastador envía un mensaje comenzando la primer subasta. A partir de este punto la subasta queda oficialmente abierta con un precio base de \$1 y los *bots* comienzan a realizar sus ofertas. Cada *bot* puede ofertar \$1, \$2 ó \$5 más que la oferta anterior, hasta el tope determinado por su configuración.

Cada oferta debe ser estrictamente superior a todas las anteriores para el artículo en venta; el precio base se considera la primera oferta. Por supuesto, no puede darse el caso de que un *bot* realice dos ofertas de manera consecutiva.

El *bot* subastador monitorea la actividad de la sala y, si transcurren 5 segundos sin una nueva oferta, envía un mensaje de advertencia indicando que la subasta está por cerrarse. Luego espera 3 segundos más y, si ningún *bot* oferta, asigna el artículo al *bot* correspondiente. Si algún *bot* oferta después de que el subastador realice la pregunta, vuelve a esperar a que transcurran otros 5 segundos sin actividad antes de volver a alertar acerca del cierre. Si el artículo termina sin ofertas por encima del precio base, se marca en el sistema como “sin interesados”.

Una vez que el primer artículo se vende, se comienza a vender el segundo y se procede de forma idéntica. Los *bots* compradores pueden desconectarse de la sala en cualquier momento, incluso en medio de una subasta (siempre y cuando no estén configurados para pujar por el mismo).

La subasta finaliza cuando el subastador envía un mensaje indicando el final, luego de la venta del último artículo. El *bot* subastador cierra la sala en este momento, desconectando automáticamente a cualquier otro *bot* que aún se encontrase conectado.

Modele los comportamientos de los distintos *bots* mediante máquinas de estados. *Hint*: piense en utilizar arreglos de variables para modelar la configuración de los *bots*.