

SEGUNDO PARCIAL

Nota: La interpretación de las consignas es parte del examen. El examen se aprueba con al menos dos problemas bien y no menos del 65% de la puntuación total. Problemas parcialmente correctos no necesariamente suman puntos.

Se evalúa: Formalización de requerimientos y conocimiento de dominio en Statecharts; marco de referencia para requerimientos y especificaciones WRSPM.

Problemas

1. Una interfaz gráfica de usuario (GUI) consta de tres pantallas: *A*, *B* y *C*. El usuario puede seleccionar cualquiera de ellas en cualquier orden pero si durante más de *N* segundos permanece en cualquier pantalla sin interactuar con el sistema, este le mostrará la pantalla *A*. Cada vez que el sistema muestra una pantalla debe dibujar la última selección del usuario.

En cada pantalla el usuario puede hacer lo siguiente:

- A* Seleccionar uno de cuatro rectángulos cada uno de un color (*rojo*, *verde*, *azul* y *blanco*). Inicialmente no hay ninguno seleccionado.
- B* Si el rectángulo seleccionado en *A* es *verde* o *rojo* y el tamaño seleccionado en *C* es *chico* entonces puede seleccionar un número entre 1 y 3. En cualquier otro caso no puede hacer nada.
- C* Si el rectángulo seleccionado en *A* es el *verde* entonces el usuario no puede seleccionar nada en esta pantalla. En cualquier otro caso puede aumentar o disminuir el tamaño del rectángulo seleccionado en *A*, eligiendo un elemento de la siguiente escala: *chico*, *mediano* o *grande*.

Designar los términos básicos del enunciado y modelar en Statecharts la especificación de la GUI.

2. Muestre un ejemplo sencillo donde se pueda apreciar la diferencia de los modelos de concurrencia de Statecharts y CSP.
3. Modelar en CSP el conocimiento de dominio y la especificación de los siguientes requerimientos. Si es necesario incluir tiempos de espera para completar los requerimientos.

Una empresa posee una balanza para pesar camiones cargados con materia prima. El camión debe ubicarse más o menos sobre el centro de la balanza para que la pesada sea correcta. Con este fin la empresa instaló cuatro sensores en los vértices de un rectángulo imaginario de forma tal que cuando detectan que el camión está dentro de ese rectángulo, se debe bajar una barrera detrás del camión. Si el camión rebasa alguno de los laterales del rectángulo se enciende una (de dos) luz ubicada delante del camión que indica qué lado está rebasado.

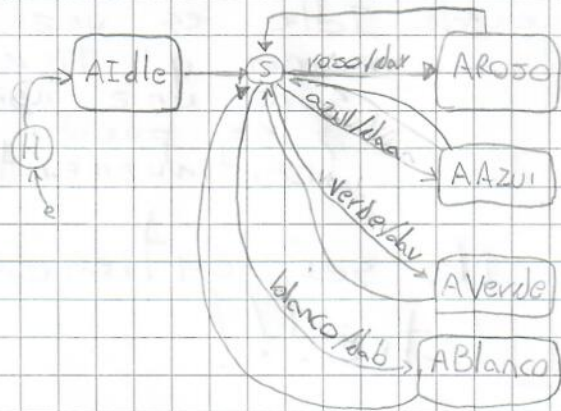
Una vez que el camión está correctamente ubicado y se bajaron las barreras, el chofer debe deslizar una tarjeta magnética que lo identifica. Si la tarjeta es válida, se activa la balanza. Cuando el pesaje finaliza, se debe imprimir un tique con los datos del conductor y el peso. Luego se levantan las barreras.

Hora 1
de
3

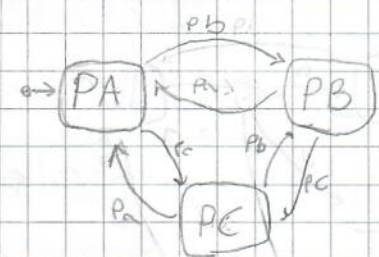
Nielsen Maximiliano Legajo: N-1218/1

- 1) se elige la pantalla A $\approx$ pa ECS se dibuja la pantalla A $\approx$ da MCS
 se elige la pantalla B $\approx$ pb ECS se dibuja la pantalla B $\approx$ db MCS
 se elige la pantalla C $\approx$ pc ECS se dibuja la pantalla C $\approx$ dc MCS
 se elige rojo en A $\approx$ rosa ECS
 se elige azul en A $\approx$ azul ECS
 se elige verde en A $\approx$ verde ECS
 se elige blanco en A $\approx$ blanco ECS
 se elige el 1 en B $\approx$ b1 ECS
 se elige el 2 en B $\approx$ b2 ECS
 se elige el 3 en B $\approx$ b3 ECS
 se elige chico en C $\approx$ chico ECS
 se elige mediano en C $\approx$ mediano ECS
 se elige grande en C $\approx$ grande ECS
 se dibuja la opcion rosa de A $\approx$ dar MCS
 se dibuja la opcion azul de A $\approx$ daa MCS
 se dibuja la opcion verde de A $\approx$ dav MCS
 se dibuja la opcion blanca de A $\approx$ dab MCS
 se dibuja la opcion 1 en B $\approx$ db1 MCS
 se dibuja la opcion 2 en B $\approx$ db2 MCS
 se dibuja la opcion 3 en B $\approx$ db3 MCS
 se dibuja la opcion chico en C $\approx$ dchico MCS
 se dibuja la opcion mediana en C $\approx$ dmediana MCS
 se dibuja la opcion grande en C $\approx$ dgrande MCS
 segundos de inactividad $\approx$ N

Pantalla A

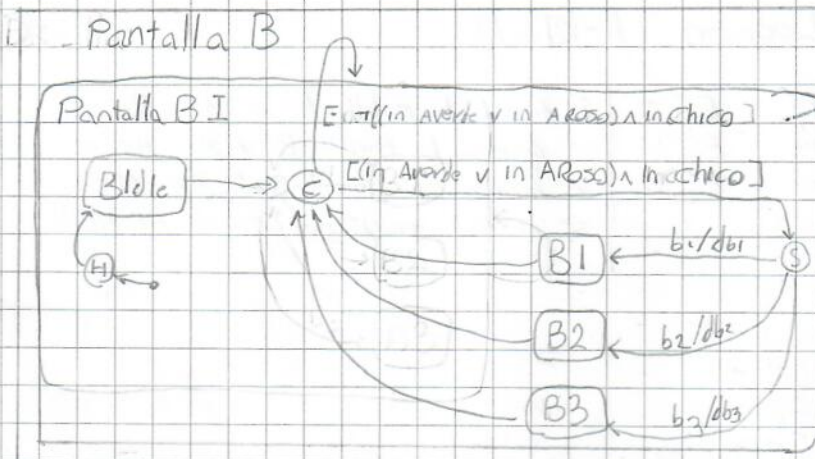


DK - Pantalla

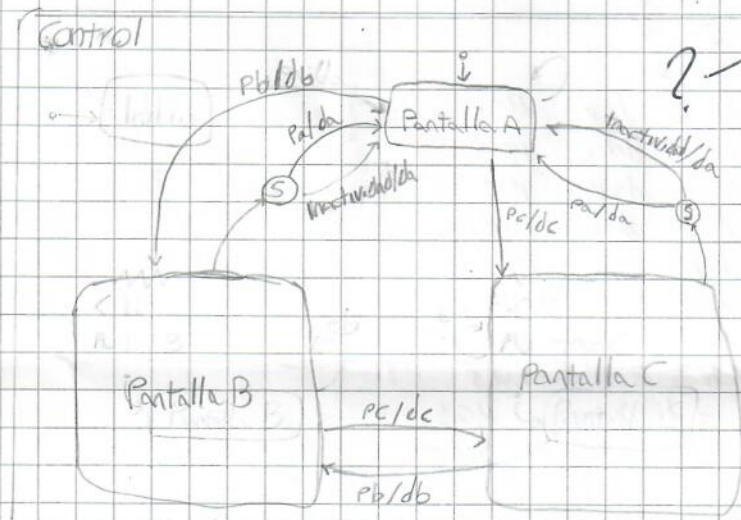


Pantalla C

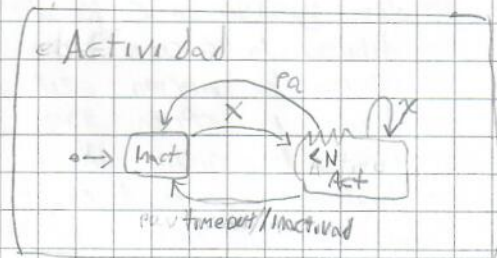




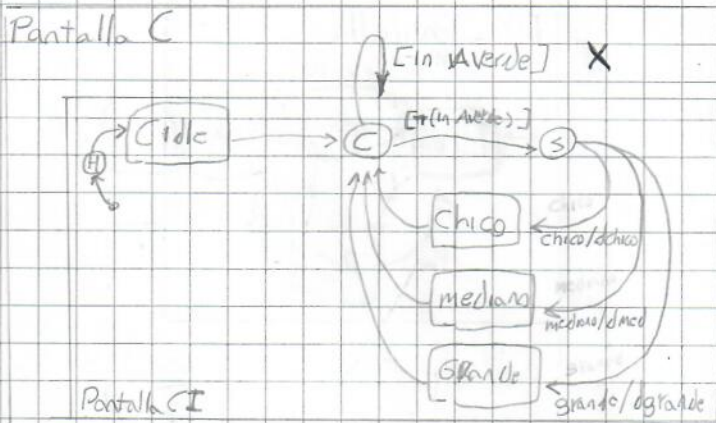
X no es necesario



letra ?

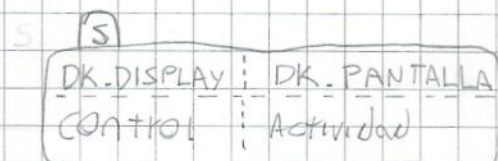
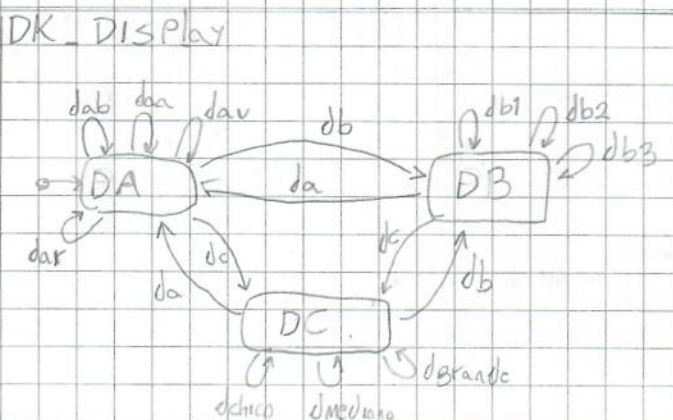


$x \stackrel{def}{=} Bb \vee Pc \vee b_1 \vee b_2 \vee b_3 \vee$
chico $\vee$ mediano $\vee$ grande



desde Idle es una traza, sign q tiene solo una cond. x se puede dar continuamente

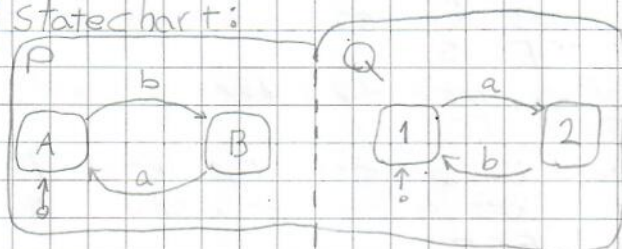
No se entiende la letra!!!



Hora 2
de
3

Nielsen Maximiliano Legajo: N-1218/1

2) Statechart:



CSP:

$P = a \rightarrow b \rightarrow P$

$Q = b \rightarrow a \rightarrow Q$

$R = P \parallel Q$

Podemos ver que en statechart ambos pueden avanzar, si llega a a Q transiciona a 2 y si llega a b P transiciona a B y Q a 1.

En CSP como $a \in \alpha Q$, $b \in \alpha P$ se produce deadlock y $R = STOP$ y ningun proceso avanza.

La diferencia es que en statechart la concurrencia es de tipo broadcast mientras que en CSP se deben seguir las reglas, interleaving, deadlock, synchronization y IEQF

3) se enciende una luz, $i=1$ es la derecha y $i=2$ es la izquierda $\approx i.ON$ MC S
 se apaga la luz, $i=1$ es la derecha y $i=2$ es la izquierda $\approx i.OF$ MC S
 se activa un sensor, $i=1$ es adelante, $i=2$ es el derecha, $i=3$ es el de atras y $i=4$ es el de la izquierda $\approx i.act$ EC S
 se desactiva un sensor, $i=1$ es el de adelante, $i=2$ es el de la derecha, $i=3$ es el de atras y $i=4$ es el de la izquierda $\approx i.des$ EC S
 se baja una barrera, $i=1$ es adelante y $i=2$ es atras $\approx i.bajar$ MC S
 se sube una barrera, $i=1$ es adelante y $i=2$ es atras $\approx i.subir$ MC S
 se lee una tarjeta valida $\approx valida$ EC S
 se lee una tarjeta invalida $\approx invalida$ EC S
 datos de un conductor $\approx c$
 se comienza a pesar $\approx pesar$ MC S
 se termina de pesar $\approx pesado$ EC S
 peso leído por la balanza $\approx p$
 se comienza a imprimir $\approx imprimir$ MC S
 se termina de imprimir $\approx impreso$ EC S

DK_LUZ = ON $\rightarrow$ OFF $\rightarrow$ DK_LUZ

LUCES = $\prod_{i=1}^2 i : DK_LUZ$

DK_SENSOR = $\overline{act} \rightarrow DK_SENSOR \mid des \rightarrow DK_SENSOR$

SENSORES = $\prod_{i=1}^4 i : DK_SENSOR$

DK_BARRERA = bajar $\rightarrow$ subir $\rightarrow$ DK_BARRERA

BARRERAS = $\prod_{i=1}^2 i : DK_BARRERA$

DK_LECTOR = valida $\rightarrow$ $choer!c \rightarrow DK_LECTOR \mid invalida \rightarrow \overline{if}i \rightarrow DK_LECTOR$

DK_BALANZA = pesar $\rightarrow$ pesado $\rightarrow$ $peso!p \rightarrow DK_BALANZA$

CONTROL = $3.act \rightarrow ESPERAR(false, false, true, false)$

DK_IMPRESORA = $!choer?c \rightarrow !reso?p \rightarrow imprimir \rightarrow \overline{impreso} \rightarrow DK_IMPRESORA$

Nielsen
Hoxa 3
de
3

ESPERAR (sn, se, ss, so) = 1.act → ESPERAR (true, se, ss, so)
| 2.act → ESPERAR (sn, true, ss, so) | se | 1.on → ESPERAR (sn, true, ss, so)
| 3.act → ESPERAR (sn, se, true, so)
| 4.act → ESPERAR (sn, se, ss, true) | so | 2.on → ESPERAR (sn, se, ss, true)
| 1.des → ESPERAR (false, se, ss, so)
| 2.des → (1.off → ESPERAR (sn, false, ss, so)) | so | ESPERAR (sn, false, ss, so)
| 3.des → ESPERAR (sn, se, false, so)
| 4.des → (2.off → ESPERAR (sn, se, ss, false)) | so | ESPERAR (sn, se, ss, false)

ESPERAR (false, false, false, false) = 1.barajar → 2.barajar → (chofer? c → reser? p
→ 1.chofer! c → 1.reser! p → maxim → impres → 1.subir
→ 2.subir → 1.act → 1.des → CONTROL | xi → INV)

S = LUCES || SENSORES || BARRERAS || DK-LECTOR || DK-BALANZA
|| DK-BALANZA || DK-IMPRESORA || CONTROL

INV = 1.subir → 2.subir → 1.act → 1.des → CONTROL

