

Universidad Nacional de Rosario

FACULTAD DE CIENCIAS EXACTAS, INGENIERÍA Y AGRIMENSURA

PRACTICA Z

Autor:
Joaquin Arroyo

3 de mayo de 2023

1. Sistema de iluminacion

- La lampara puede estar encendida \approx on
- La lampara puede estar apagada \approx off
- Podemos cambiar el estado de la lampara a partir del boton \approx Boton()
- El sistema tiene como estado, el estado de la lampara $estado : Estado \approx Lampara(estado)$

Estado ::= on | off

$\frac{Lampara}{estado : Estado}$

$\frac{LamparaInit \quad Lampara}{estado = off}$
--

$\frac{Boton \quad \Delta Lampara}{\begin{array}{l} estado = off \Rightarrow on \\ estado = on \Rightarrow off \end{array}}$
--

2. Despachante de mensajes

- m es un Mensaje $\approx m \in Msg$
- El despachante esta enviando/recibiendo un mensaje \approx enviando/recibiendo
- No hubo error al recibir mensaje \approx ok
- Hubo error al recibir mensaje \approx error
- El mensaje m tiene paridad correcta $\approx m \in ParidadOk$
- El despachante tiene estado s y almacena los mensajes en el buffer b \approx Despachante(s, b)
- Mientras el despachante esta recibiendo por el canal c_n , se envia por el canal r el resultado de la operacion \approx Canal $_n(c_n?, e_n!)$
- El despachante esta recibiendo y su buffer se lleno \approx BufferLleno
- El despachante esta enviando los mensajes por el canal $r?$ \approx BufferVaciando($r!$)
- El despachante esta enviando y su buffer esta vacio \approx BufferVacio

[Msg]
estado ::= recibiendo | enviando
mensaje ::= ok | error

$\frac{ParidadOk : \mathbb{P} Msg}{ParidadOk \neq \emptyset}$

$\frac{TamanoBuffer : \mathbb{N}}{TamanoBuffer = 10}$

<i>Despachante</i>	$s : estado$ $b : seq\ Msg$
<i>DespachanteInit</i>	$s = recibiendo$ $b = \langle \rangle$
<i>CanalOk</i>	$\Delta Despachante$ $c_1? : Msg$ $e_1! : mensaje$ $b \leq TamanoBuffer - 1$ $c_1? \in ParidadOk$ $s = recibiendo$ $s' = s$ $b' = b \frown \langle c_1? \rangle$ $e_1! = ok$
<i>BufferLleno</i>	$\Delta Despachante$ $e_1! : mensaje$ $b \geq TamanoBuffer$ $s = recibiendo$ $s' = enviando$ $b' = b$ $e_1! = error$
<i>MensajeErroneo</i>	$\Xi Despachante$ $c_1? : Msg$ $e_1! : mensaje$ $c_1? \notin ParidadOk$ $s = recibiendo$ $e_1! = error$
<i>CanalEnviando</i>	$\Xi Despachante$ $e_1! : mensaje$ $s = enviando$ $e_1! = error$
<i>BufferVacio</i>	$\Delta Despachante$ $b = 0$ $s = enviando$ $s' = recibiendo$ $b' = b$

$BufferVaciando$
$\Delta Despachante$
$r! : Msg$
$b > 0$
$s = enviando$
$s' = s$
$b' = tail(b)$
$r! = head(b)$

$Canal1 = Canal1Ok \vee BufferLleno \vee MensajeErroneo \vee CanalEnviando \vee BufferVacio \vee BufferVaciando$
 $Canal2 = Canal1[c_1?/c_2?, e_1!/e_2!]$

3. BD Cinematografica

- g es un guionista $\approx g \in Guionista$
- d es un director $\approx d \in Director$
- p es una pelicula $\approx p \in Pelicula$
- Hubo error en una consulta $\approx error$
- No hubo error en una consulta $\approx ok$
- La BD tiene una tabla *peliculas* en su estado $\approx BD(peliculas)$
- Se modifica el director $do?$ de la pelicula $p?$ por el director $dn?$ y se devuelve la respuesta $rep!$
 $\approx ModificarPelicula(p?, do?, dn?, rep!)$
- Se crea la pelicula $p?$ con los directores $ds?$ y los guionistas $gs?$, y se devuelve la respueesta $rep!$
 $\approx CrearPelicula(p?, ds?, gs?, rep!)$
- Se borra la pelicula $p?$ y se devuelve la respuesta $rep! \approx BorrarPelicula(p?, rep!)$
- Se buscan peliculas $ps!$ a partir de un director $d?$ y se devuelven $\approx BuscarPeliculaPorDirector(d?, ps!)$
- Se buscan peliculas $ps!$ a partir de un guionista $g?$ y se devuelven $\approx BuscarPeliculaPorGuionista(g?, ps!)$

[PELICULA, DIRECTOR, GUIONISTA]

MSG ::= error | ok

BD
$peliculas : PELICULA \rightarrow (\mathbb{P} DIRECTOR \times \mathbb{P} GUIONISTA)$

$BDInit$
BD
$peliculas = \emptyset$

<i>ModificarPeliculaOk</i>
ΔBD
$p? : PELICULA$
$do? : DIRECTOR$
$dn? : DIRECTOR$
$rep! : MSG$
$p? \in dom(películas)$
$do? \in películas(p?).1$
$películas' = películas \oplus \{p? \mapsto ((películas(p?) \setminus \{do?\}) \cup \{dn?\}, películas(p?).2)\}$
$rep! = ok$

<i>PeliculaInexistente</i>
ΞBD
$p? : PELICULA$
$rep! : MSG$
$p? \notin dom(películas)$
$rep! = error$

<i>DirectorInexistente</i>
ΞBD
$p? : PELICULA$
$do? : PELICULA$
$rep! : MSG$
$do? \notin películas(p?).1$
$rep! = error$

$ModificarPelicula == ModificarPeliculaOk \vee PeliculaInexistente \vee DirectorInexistente$

<i>CrearPeliculaOk</i>
ΔBD
$p? : PELICULA$
$ds? : \mathbb{P} DIRECTOR$
$gs? : \mathbb{P} GUIONISTA$
$rep! : MSG$
$p? \notin películas$
$películas' = películas \cup \{p? \mapsto (ds?, gs?)\}$
$rep! = ok$

<i>PeliculaExistente</i>
ΞBD
$p? : PELICULA$
$rep! : MSG$
$p? \in películas$
$rep! = error$

$CrearPelicula == CrearPeliculaOk \vee PeliculaExistente$

<i>BorrarPeliculaOk</i>
ΔBD
$p? : PELICULA$
$rep! : MSG$
$p? \in peliculas$
$peliculas' = \{p : Pelicula \mid p \in dom(peliculas) \wedge p \neq p?\}$
$rep! = ok$

$BorrarPelicula == BorrarPeliculaOk \vee PeliculaInexistente$

<i>BuscarPeliculaPorDirector</i>
ΔBD
$d? : DIRECTOR$
$ps! : \mathbb{P} PELICULA$
$ps! = \{p \mid p \in dom(peliculas) \wedge d? \in peliculas(p).1\}$

<i>BuscarPeliculaPorGuionista</i>
ΔBD
$g? : GUIONISTA$
$ps! : \mathbb{P} PELICULA$
$ps! = \{p \mid p \in dom(peliculas) \wedge g? \in peliculas(p).2\}$

4. BDCinematografica con promocion

[TITULO, DIRECTOR, GUIONISTA]

$Msg ::= ok \mid error$

<i>Pelicula</i>
$dirs : \mathbb{P} DIRECTOR$
$guions : \mathbb{P} GUIONISTA$

<i>PeliculaInit</i>
<i>Pelicula</i>
$dirs = \emptyset$
$guions = \emptyset$

<i>BD</i>
$peliculas : TITULO \leftrightarrow Pelicula$

<i>BDInit</i>
<i>BD</i>
$peliculas = \emptyset$

<i>BuscarPelículasPorDir</i>
$\exists BD$ $d? : DIRECTOR$ $p! : \mathbb{P} TITULO$ $rep! : MSG$
$p! = \{t \mid t \in dom(películas) \wedge p = películas(t) \wedge d? \in p.dirs\}$ $rep! = ok$

<i>BuscarPelículasPorGui</i>
$\exists BD$ $g? : GUIONISTA$ $p! : \mathbb{P} TITULO$ $rep! : MSG$
$p! = \{t \mid t \in dom(películas) \wedge p = películas(t) \wedge g? \in p.guions\}$ $rep! = ok$

<i>ModificarDirOk</i>
$\Delta Película$ $do? : DIRECTOR$ $dn? : DIRECTOR$ $rep! : MSG$
$do? \in dirs$ $dn? \notin dirs$ $dirs' = (dirs \setminus do?) \cup dn?$ $guions' = guions$ $rep! = ok$

<i>DirectorExistente</i>
$\exists Película$ $dn? : DIRECTOR$ $rep! : MSG$
$dn? \in dirs$ $rep! = error$

<i>DirectorInexistente</i>
$\exists Película$ $dno? : DIRECTOR$ $rep! : MSG$
$do? \notin dirs$ $rep! = error$

$ModificarDir == ModificarDirOk \vee DirectorExistente \vee DirectorInexistente$

<i>ModificarDirBDok</i>
ΔBD $ModificarDir$ $t? : TITULO$ $rep! : MSG$
$t? \in dom(películas)$ $películas(t) = \theta Película$ $películas' = películas \oplus \{t? \mapsto \theta Película'\}$ $rep! = ok$

<i>PeliculaInexistente</i>
ΞBD
$t? : TITULO$
$rep! : MSG$
$t? \notin dom(peliculas)$
$rep! = error$

$ModificarDirBD == ModificarDirBDok \vee PeliculaInexistente$

<i>CrearPelicula</i>
$\Delta Pelicula$
$ds? : \mathbb{P} DIRECTOR$
$gs? : \mathbb{P} GUIONISTA$
$rep! : MSG$
$dirs' = ds?$
$guions' = gs?$
$rep! = ok$

<i>CrearPeliculaBDok</i>
ΔBD
<i>CrearPelicula</i>
$t? : TITULO$
$rep! : MSG$
$t? \notin dom(peliculas)$
$peliculas' = peliculas \cup \{t? \mapsto \theta Pelicula'\}$
$rep! = ok$

<i>PeliculaExistente</i>
ΞBD
$t? : TITULO$
$rep! : MSG$
$t? \in dom(peliculas)$
$rep! = error$

$CrearPeliculaBD == CrearPeliculaBDok \vee PeliculaExistente$

5. Servidor de FTP

- u es un usuario $\approx u \in USUARIO$
- p es una contrasenia $\approx u \in CONTRASENIA$
- n es el nombre de un archivo $\approx n \in NOMBREARCHIVO$
- a es un archivo $\approx a \in ARCHIVO$
- Hubo error en una operacion $\approx error$
- No hubo error en una operacion $\approx ok$
- Dado un usuario $u?$ y una contrasenia $c?$, el usuario se puede conectar al servidor, y se devuelve una respuesta $rep! \approx Connect(u?, c?, rep!)$

- Dado el nombre de un archivo $n?$, un usuario conectado al servidor $u?$, se devuelve como respuesta el archivo solicitado $a!$, en caso de que este exista, y tambien se devuelve una respuesta $rep! \approx \text{Get}(n?, u?, a!, rep!)$
- Dado el nombre de un archivo $n?$, el archivo $a?$, y un usuario $u?$ conectado al servidor, se agrega el archivo al servidor, y se devuelve una respuesta $rep! \approx \text{Put}(n?, a?, rep!)$
- Dado un usuario $u?$ conectado al servidor, se cierra la conexion, y se devuelve un mensaje $rep! \approx \text{Close}(u?, rep!)$
- El servidor tiene un conjunto de usuarios $usuarios : USUARIO \rightarrow CONTRASENIA$, un conjunto de usuarios conectados $usuariosConectados : \mathbb{P} USUARIO$ y un conjunto de archivos $archivos : NOMBREARCHIVO \rightarrow ARCHIVO$

[USUARIO, CONTRASENIA, NOMBREARCHIVO, ARCHIVO]

MSG ::= error | ok

<i>Servidor</i>
$usuarios : USUARIO \rightarrow CONTRASENIA$
$usuariosConectados : \mathbb{P} USUARIO$
$archivos : NOMBREARCHIVO \rightarrow ARCHIVO$

<i>ServidorInit</i>
<i>Servidor</i>
$usuarios = \emptyset$
$usuariosConectados = \emptyset$
$archivos = \emptyset$

<i>ConnectOk</i>
$\Delta \text{Servidor}$
$u? : USUARIO$
$c? : CONTRASENIA$
$rep! : MSG$
$u? \in \text{dom}(usuarios)$
$c? = usuarios(u?)$
$usuariosConectados' = usuariosConectados \cup \{u?\}$
$usuarios' = usuarios$
$archivos' = archivos$
$rep! = ok$

<i>UsuarioInexistente</i>
$\exists \text{Servidor}$
$u? : USUARIO$
$rep! : MSG$
$u? \notin \text{dom}(usuarios)$
$rep! = error$

ContraseniaIncorrecta

$\exists \text{Servidor}$
 $u? : \text{USUARIO}$
 $c? : \text{CONTRASENIA}$
 $\text{rep!} : \text{MSG}$

$u? \in \text{dom}(\text{usuarios})$
 $c? \neq \text{usuarios}(u?)$
 $\text{rep!} = \text{error}$

$\text{Connect} == \text{ConnectOk} \vee \text{UsuarioInexistente} \vee \text{ContraseniaIncorrecta}$

GetOk

$\exists \text{Servidor}$
 $n? : \text{NOMBREARCHIVO}$
 $u? : \text{USUARIO}$
 $a! : \text{ARCHIVO}$
 $\text{rep!} : \text{MSG}$

$u? \in \text{usuariosConectados}$
 $n? \in \text{dom}(\text{archivos})$
 $a! = \text{archivos}(n?)$
 $\text{rep!} = \text{ok}$

UsuarioNoConectado

$\exists \text{Servidor}$
 $u? : \text{USUARIO}$
 $\text{rep!} : \text{MSG}$

$u? \notin \text{usuariosConectados}$
 $\text{rep!} = \text{error}$

ArchivoInexistente

$\exists \text{Servidor}$
 $n? : \text{NOMBREARCHIVO}$
 $\text{rep!} : \text{MSG}$

$n? \notin \text{dom}(\text{archivos})$
 $\text{rep!} = \text{error}$

$\text{Get} == \text{GetOk} \vee \text{UsuarioNoConectado} \vee \text{ArchivoInexistente}$

PutOk

$\Delta \text{Servidor}$
 $n? : \text{NOMBREARCHIVO}$
 $a? : \text{ARCHIVO}$
 $u? : \text{USUARIO}$
 $\text{rep!} : \text{MSG}$

$u? \in \text{usuariosConectados}$
 $n? \notin \text{dom}(\text{archivos})$
 $\text{archivos}' = \text{archivos} \cup \{n? \mapsto a?\}$
 $\text{usuarios}' = \text{usuarios}$
 $\text{usuariosConectados}' = \text{usuariosConectados}$
 $\text{rep!} = \text{ok}$

ArchivoExistente $\exists \text{Servidor}$ $n? : \text{NOMBREARCHIVO}$ $\text{rep!} : \text{MSG}$
$n? \in \text{dom}(\text{archivos})$ $\text{rep!} = \text{error}$

$\text{Put} == \text{PutOk} \vee \text{UsuarioNoConectado} \vee \text{ArchivoExistente}$

CloseOk $\Delta \text{Servidor}$ $u? : \text{USUARIO}$ $\text{rep!} : \text{MSG}$
$u? \in \text{usuariosConectados}$ $\text{archivos}' = \text{archivos}$ $\text{usuarios}' = \text{usuarios}$ $\text{usuariosConectados}' = \{u \mid u \in \text{usuariosConectados} \wedge u \neq u?\}$ $\text{rep!} = \text{ok}$

$\text{Close} == \text{CloseOk} \vee \text{UsuarioNoConectado}$

6. SIA: Sistema de iluminacion automatico

- o indica una obra $\approx o \in \text{OBRA}$
- g indica la cantidad de grados que rotaron las varillas horizontales de las persianas americanas $\approx g \in \mathbb{Z}$
- c indica la elevacion de las persianas americanas en centimetros $\approx c \in \mathbb{N}$
- i indica la intensidad de la luz artificial $\approx i \in \mathbb{N}$
- s indica la cantidad de luz que devuelven los sensores $\approx s \in \mathbb{N}$
- Si la persiana no puede realizar un movimiento, o si la luz llego a su intensidad maxima/minima $\approx \text{limite}$
- Si la persiana puede realizar un movimiento, o si la luz puede aumentar/disminuir su intensidad $\approx \text{ok}$
- La operacion abre 1 grado las persianas, se devuelve como respuesta $\text{rep!} \approx \text{Abrir}(\text{rep!})$
- La operacion cierra 1 grado las persianas, se devuelve como respuesta $\text{rep!} \approx \text{Cerrar}(\text{rep!})$
- La operacion sube 1cm las persianas, se devuelve como respuesta $\text{rep!} \approx \text{Subir}(\text{rep!})$
- La operacion baja 1cm las persianas, se devuelve como respuesta $\text{rep!} \approx \text{Bajar}(\text{rep!})$
- La operacion aumenta en uno la intensidad de la luz artificial, se devuelve como respuesta $\text{rep!} \approx \text{Aumentar}(\text{rep!})$
- La operacion disminuye en uno la intensidad de la luz artificial, se devuelve como respuesta $\text{rep!} \approx \text{Disminuir}(\text{rep!})$
- La persiana americana va a tener como estado $\text{grados} : \mathbb{N}$ y $\text{elevacion} : \mathbb{N}$
- $\text{GradosMaximos} : \mathbb{N}$, es el valor maximo de rotacion permitido para las persianas
- $\text{ElevacionMaxima} : \mathbb{N}$, es el valor maximo de elevacion permitido para las persianas

- La luz artificial va a tener como estado *intensidad* : \mathbb{N}
 - El sistema va a tener como estado, *obraLuz* : $OBRA \rightarrow \mathbb{N}$, *obraLuzNecesaria* : $OBRA \rightarrow \mathbb{N}$
- [OBRA]

Msg ::= ok | error

CorregirIluminacionNat $\subseteq \mathbb{P} OBRA$

CorregirIluminacionArt $\subseteq \mathbb{P} OBRA$

Porcentaje ::= {n | n $\in \mathbb{N} \wedge 0 \leq n \leq 100$ }

<i>Persiana</i> <i>vertical</i> : Porcentaje <i>horizontal</i> : Porcentaje

<i>PersianaInit</i> <i>vertical</i> = 0 <i>horizontal</i> = 0

<i>SubirPersianaOk</i> $\Delta Persiana$ <i>rep!</i> : Msg
<i>vertical</i> < 100 <i>vertical'</i> = <i>vertical</i> + 1 <i>horizontal'</i> = <i>horizontal</i> <i>rep!</i> = ok

<i>AlturaMaxima</i> $\exists Persiana rep! : Msg$
<i>vertical</i> ≥ 100 <i>rep!</i> = error

SubirPersiana == SubirPersianaOk \vee AlturaMaxima

<i>BajarPersianaOk</i> $\Delta Persiana$
<i>vertical</i> > 0 <i>vertical'</i> = <i>vertical</i> - 1 <i>horizontal'</i> = <i>horizontal</i>

<i>AlturaMinima</i> $\exists Persiana rep! : Msg$
<i>vertical</i> ≤ 0 <i>rep!</i> = error

BajarPersiana == BajarPersianaOk \vee AlturaMinima

<i>AbrirPersianaOk</i>
$\Delta Persiana$
$rep! : Msg$
$horizontal < 100$
$vertical' = vertical$
$horizontal' = horizontal + 1$
$rep! = ok$

<i>RotacionMaxima</i>
$\Xi Persianarep! : Msg$
$horizontal = 100$
$rep! = error$

$AbrirPersiana == AbrirPersianaOk \vee RotacionMaxima$

<i>CerrarPersianaOk</i>
$\Delta Persiana$
$horizontal > 0$
$vertical' = vertical$
$horizontal' = horizontal - 1$

<i>RotacionMinima</i>
$\Xi Persianarep! : Msg$
$horizontal = 0$
$rep! = error$

$CerrarPersiana == CerrarPersianaOk \vee RotacionMinima$

<i>LuzArtificial</i>
$intensidad : Porcentaje$

<i>LuzArtificialInit</i>
$intensidad = 0$

<i>SubirIntensidadOk</i>
$\Delta LuzArtificial$
$rep! : Msg$
$vertical < 100$
$intensidad' = intensidad + 1$
$rep! = ok$

<i>IntensidadMaxima</i>
$\Xi LuzArtificialrep! : Msg$
$intensidad = 100$
$rep! = error$

$SubirIntensidad == SubirIntensidadOk \vee IntensidadMaxima$

$BajarIntensidadOk$
$\Delta LuzArtificial$
$intensidad > 0$ $intensidad' = intensidad - 1$

$IntensidadMinima$
$\exists LuzArtificial rep! : Msg$
$intensidad = 0$ $rep! = error$

$BajarIntensidad == BajarIntensidadOk \vee IntensidadMinima$

SIA
$obras : OBRA \rightarrow (\mathbb{P} Persiana \times LuzArtificial)$ $luzObras : OBRA \rightarrow (Porcentaje \times Porcentaje)$ $\forall o \in dom(luzObras) \Rightarrow o \in dom(obras)$

$SIAInit$
SIA
$obras = \emptyset$ $luzObras = \emptyset$

$ChequearObraOk$
$\exists SIA$ $o? : OBRA$ $rep! : Msg$
$o? \in dom(obras)$ $luzObras(o?).1 = luzObras(o?).2$ $rep! = ok$

$ChequearYCorregirObraPersiana1$
ΔSIA $o? : OBRA$ $rep! : Msg$
$o? \in dom(obras)$ $luzObras(o?).1 < luzObras(o?).2$ $obras' = obras \oplus \{o? \mapsto (ps, obras(o?).2) \mid ps =$ $\{SubirPersiana; AbrirPersiana \mid p \in obras(o?).2 \wedge p = \theta Obra'\}\}$ $rep! = ok$

$ChequearYCorregirObraPersiana2$
ΔSIA $o? : OBRA$ $rep! : Msg$
$o? \in dom(obras)$ $luzObras(o?).1 > luzObras(o?).2$ $-- Cerrarpersianas$ $rep! = ok$

$ChequearObra ==$

7. Control de versiones (CVS)

- n es un nombre de archivo $\approx n \in \text{NOMBREARCHIVO}$
- e es un encabezado de archivo $\approx n \in \text{ENCABEZADOARCHIVO}$
- c es el cuerpo de un archivo $\approx c \in \text{CUERPOARCHIVO}$
- u es un usuario $\approx u \in \text{USUARIO}$
- Si la operacion termino con exito $\approx \text{ok}$
- Si la operacion no termino con exito $\approx \text{error}$
- Los usuarios pueden ser: *Lector*, *Editor*, *Autor* $\approx \text{Rol} ::= \text{Lector} \mid \text{Editor} \mid \text{Autor}$
- En *AxiomaRoles* definimos que $\text{Autor} \Rightarrow \text{Editor}$, y $\text{Editor} \Rightarrow \text{Lector}$
- El sistema, tiene como estado un repositorio, llamado linea base, el cual contiene todos los programas junto a sus versiones, un conjunto de archivos editados, un conjunto de usuarios junto a sus permisos $\approx \text{CVS}(\text{lineabase}, \text{archivosEditados}, \text{usuarios})$
- Un usuario $u?$ con rol *Autor*, podra crear un archivo en linea base, a partir de un nombre $n?$, un encabezado $e?$ y un cuerpo $c?$, devolvemos una respuesta $rep!$ indicando como termino la operacion $\approx \text{Create}(u?, n?, e?, c?, rep!)$
- Un usuario $u?$ con rol *Lector*, podra obtener un archivo, a partir de un nombre $n?$, devolvemos una respuesta $rep!$ indicando como termino la operacion y una copia del archivo solicitado $a! \approx \text{Get}(u?, n?, rep!, a!)$
- Un usuario $u?$ con rol *Editor*, podra editar un archivo de la linea base, a partir de un nombre $n?$, un encabezado $e?$ y un cuerpo $c?$, devolvemos una respuesta $rep!$ indicando como termino la operacion, se agrega la edicion a *archivoseditados* $\approx \text{Edit}(u?, n?, e?, c?, rep!)$
- El usuario $u?$ *Editor* del archivo $n?$ podra hacerle Delta, se devuelve como respuesta $rep! \approx \text{Delta}(u?, n?, rep!)$
- Un usuario $u?$ *Autor*, podra borrar todo el historial de un archivo $n?$, se devuelve como respuesta $rep! \approx \text{Delete}(u?, n?, rep!)$

[NOMBREARCHIVO, ENCABEZADOARCHIVO, CUERPOARCHIVO, USUARIO]

$\text{Rol} ::= \text{Lector} \mid \text{Editor} \mid \text{Autor}$

<i>AxiomaRoles</i>
$\text{Autor} \Rightarrow \text{Editor}$
$\text{Editor} \Rightarrow \text{Lector}$
<i>CSV</i>
$\text{lineabase} :$
$\text{NOMBREARCHIVO} \rightarrow \text{Seq}(\text{ENCABEZADOARCHIVO} \times \text{CUERPOARCHIVO})$
$\text{archivoseditados} :$
$\text{NOMBREARCHIVO} \rightarrow (\text{ENCABEZADOARCHIVO} \times \text{CUERPOARCHIVO}) \times \text{USUARIO}$
$\text{usuarios} : \text{USUARIO} \rightarrow \text{Rol}$
<i>CSVInit</i>
<i>CSV</i>
$\text{lineabase} = \emptyset$
$\text{archivoseditados} = \emptyset$
$\text{usuarios} = \emptyset$

CreateOk ΔCSV $u? : USUARIO$ $n? : NOMBREARCHIVO$ $e? : ENCABEZADOARCHIVO$ $c? : CUERPOARCHIVO$ $rep! : Msg$
$u? \in \text{dom}(\text{usuarios})$ $\text{usuarios}(u?) = \text{Autor}$ $n? \notin \text{dom}(\text{lineabase})$ $\text{lineabase}' = \text{lineabase} \cup \{n? \mapsto \langle (e?, c?) \rangle\}$ $\text{archivoseditados}' = \text{archivoseditados}$ $\text{usuarios}' = \text{usuarios}$ $rep! = \text{ok}$

$\text{UsuarioInexistente}$ ΞCSV $u? : USUARIO$ $rep! : Msg$
$u? \notin \text{dom}(\text{usuarios})$ $rep! = \text{error}$

UsuarioNoEsAutor ΞCSV $u? : USUARIO$ $rep! : Msg$
$u? \in \text{dom}(\text{usuarios})$ $\text{usuarios}(u?) \neq \text{Autor}$ $rep! = \text{error}$

ArchivoExistente ΞCSV $n? : NOMBREARCHIVO$ $rep! : Msg$
$n? \in \text{dom}(\text{lineabase})$ $rep! = \text{error}$

$\text{Create} == \text{CreateOk} \vee \text{UsuarioInexistente} \vee \text{UsuarioNoEsAutor} \vee \text{ArchivoExistente}$

GetOk ΞCSV $u? : USUARIO$ $n? : NOMBREARCHIVO$ $rep! : Msg$ $a! : (\text{ENCABEZADOARCHIVO} \times \text{CUERPOARCHIVO})$
$u? \in \text{dom}(\text{usuarios})$ $n? \in \text{dom}(\text{lineabase})$ $a! = \text{head}(\text{lineabase}(n?))$ $rep! = \text{ok}$

<i>ArchivoInexistente</i>
ΞCSV
$n? : NOMBREARCHIVO$
$rep! : Msg$
$n? \notin dom(lineabase)$
$rep! = error$

$Get == GetOk \vee UsuarioInexistente \vee ArchivoInexistente$

<i>EditOk</i>
ΔCSV
$u? : USUARIO$
$n? : NOMBREARCHIVO$
$e? : ENCABEZADOARCHIVO$
$c? : CUERPOARCHIVO$
$rep! : Msg$
$u? \in dom(usuarios)$
$usuarios(u?) = Editor$
$n? \in dom(lineabase)$
$n? \notin dom(archivoeditados)$
$lineabase' = lineabase$
$archivoseditados' = archivoseditados \cup \{n? \rightarrow ((e?, c?), u?)\}$
$usuarios' = usuarios$
$rep! = ok$

<i>UsuarioNoEsEditor</i>
ΞCSV
$u? : USUARIO$
$rep! : Msg$
$u? \in dom(usuarios)$
$usuarios(u?) \neq Editor$
$rep! = error$

<i>DeltaTrue</i>
ΞCSV
$n? : NOMBREARCHIVO$
$rep! : Msg$
$n? \in dom(archivoeditados)$
$rep! = error$

$Edit == EditOk \vee UsuarioInexistente \vee UsuarioNoEsEditor \vee ArchivoInexistente \vee DeltaTrue$

<div><div>DeltaOk</div><div>ΔCSV $u? : USUARIO$ $n? : NOMBREARCHIVO$ $rep! : Msg$</div><div>$u? \in dom(usuarios)$ $n? \in dom(lineabase)$ $n? \in dom(archivoeditados)$ $u? = archivoseditados(n?).2$ $head(lineabase(n?)) \neq archivoseditados(n?).1$ $lineabase' = lineabase \oplus \{n? \mapsto \langle archivoseditados(n?).1 \rangle \wedge lineabase(n?)\}$ $archivoseditados' = \{n?\} \triangleleft archivoseditados$ $usuarios' = usuarios$ $rep! = ok$</div></div>	
<div><div>DeltaFalse</div><div>ΞCSV $n? : NOMBREARCHIVO$ $rep! : Msg$</div><div>$n? \notin dom(archivoeditados)$ $rep! = error$</div></div>	
<div><div>EditorErroneo</div><div>ΞCSV $n? : u?$ $rep! : Msg$</div><div>$u? \in dom(usuarios)$ $u? \neq archivoseditados(n?).2$ $rep! = error$</div></div>	
<div><div>NoHayDiferencias</div><div>ΞCSV $n? : NOMBREARCHIVO$ $rep! : Msg$</div><div>$n? \in dom(lineabase)$ $head(lineabase(n?)) = archivoseditados(n?)$ $rep! = error$</div></div>	
elta == DeltaOk \vee UsuarioInexistente \vee ArchivoInexistente \vee DeltaFalse \vee EditorErroneo \vee NoHayDiferencias	
<div><div>DeleteOk</div><div>ΔCSV $u? : USUARIO$ $n? : NOMBREARCHIVO$ $rep! : Msg$</div><div>$u? \in dom(usuarios)$ $usuarios(u?) = Autor$ $n? \in dom(lineabase)$ $lineabase' = \{n?\} \triangleleft lineabase$ $archivoseditados = \{n?\} \triangleleft archivoseditados$ $usuarios' = usuarios$ $rep! = ok$</div></div>	

$Delete == DeleteOk \vee UsuarioInexistente \vee UsuarioNoEsAutor \vee ArchivoInexistente$

8. CSV con promocion

$[NOMBREARCHIVO, ENCABEZADOARCHIVO, CUERPOARCHIVO, USUARIO]$

$Rol ::= Lector \mid Editor \mid Autor$

$Msg ::= ok \mid error$

<i>AxiomaRoles</i>	
$Autor \Rightarrow Editor$	
$Editor \Rightarrow Lector$	
<i>Program</i>	
$program : seq(ENCABEZADOARCHIVO \times CUERPOARCHIVO)$	
$editedProgram : ENCABEZADOARCHIVO \times CUERPOARCHIVO$	
<i>ProgramInit</i>	
$program = \langle \rangle$	
$editedProgram = \emptyset$	
<i>CSV</i>	
$lineabase : NOMBREARCHIVO \rightarrow Program$	
$editedPrograms : \mathbb{P}(NOMBREARCHIVO \times USUARIO)$	
$users : USUARIO \rightarrow Rol$	
<i>CSVInv</i>	
CSV	
$\forall (n, u) \in editedPrograms \Rightarrow n \in dom(lineabase) \wedge u \in dom(users) \wedge$	
$(users(u) = editor \vee users(u) = autor)$	
<i>CSVInit</i>	
$lineabase = \emptyset$	
$editedPrograms = \emptyset$	
$users = \emptyset$	
<i>CreateProgram</i>	
$\Delta Program$	
$h? : ENCABEZADOARCHIVO$	
$c? : CUERPOARCHIVO$	
$program' = \langle (h?, c?) \rangle$	
$editedProgram' = \emptyset$	

$\text{CreateProgramOkCSV}$ ΔCSV CreateProgram $n? : \text{NOMBREARCHIVO}$ $u? : \text{USUARIO}$ $\text{rep!} : \text{Msg}$
$n? \notin \text{dom}(\text{lineabase})$ $u? \in \text{dom}(\text{usuarios})$ $\text{usuarios}(u?) = \text{Autor}$ $\text{lineabase}' = \text{lineabase} \cup \{n? \mapsto \theta \text{Program}'\}$ $\text{rep!} = \text{ok}$

ExistingProgram ΞCSV $n? : \text{NOMBREARCHIVO}$ $\text{rep!} : \text{Msg}$
$n? \in \text{dom}(\text{lineabase})$ $\text{rep!} = \text{error}$

UnknownUser ΞCSV $u? : \text{USUARIO}$ $\text{rep!} : \text{Msg}$
$u? \notin \text{dom}(\text{usuarios})$ $\text{rep!} = \text{error}$

NotAuthor ΞCSV $u? : \text{USUARIO}$ $\text{rep!} : \text{Msg}$
$u? \in \text{dom}(\text{usuarios})$ $\text{usuarios}(u?) \neq \text{Autor}$ $\text{rep!} = \text{error}$

$\text{CreateProgramCSV} == \text{CreateProgramOkCSV} \vee \text{ExistingProgram} \vee \text{UnknownUser} \vee \text{NotAuthor}$

GetOk ΞCSV $n? : \text{NOMBREARCHIVO}$ $u? : \text{USUARIO}$ $p! : \text{Program}$ $\text{rep!} : \text{Msg}$
$n? \in \text{dom}(\text{lineabase})$ $u? \in \text{dom}(\text{usuarios})$ $p! = \text{head}(\text{lineabase}(n?).\text{program})$ $\text{rep!} = \text{ok}$

<i>UnknownProgram</i>
$\exists CSV$
$n? : NOMBREARCHIVO$
$rep! : Msg$
$n? \notin dom(lineabase)$
$rep! = error$

Get == GetOk \vee *UnknownProgram* \vee *UnknownUser*

<i>EditOk</i>
$\Delta Program$
$e? : ENCABEZADOARCHIVO$
$c? : CUERPOARCHIVO$
$rep! : Msg$
$editedProgram = \emptyset$
$program' = program$
$editedProgram' = (e?, c?)$
$rep! = ok$

<i>ProgramIsEdited</i>
$\exists Program$
$rep! : Msg$
$editedProgram \neq \emptyset$
$rep! = error$

Edit == EditOk \vee *ProgramIsEdited*

<i>EditCSVok</i>
ΔCSV
<i>Edit</i>
$u? : USUARIO$
$n? : NOMBREARCHIVO$
$rep! : Msg$
$u? \in dom(users)$
$n? \in dom(lineabase)$
$usuarios(u?) = editor$
$lineabase(n?) = \theta Program$
$lineabase' = lineabase \oplus \{n? \leftrightarrow \theta Program'\}$
$editedPrograms' = \{(n?, u?)\} \cup editedPrograms$
$users' = users$
$rep! = ok$

<i>NotEditor</i>
$\exists CSV$
$u? : USUARIO$
$rep! : Msg$
$u? \in dom(usuarios)$
$usuarios(u?) \neq Editor$
$rep! = error$

EditCSV == EditCSVok \vee *UnkownUser* \vee *UnknownProgram* \vee *NotEditor*

<i>DeltaOk</i>
$\Delta Program$
$rep! : Msg$
$editedProgram \neq head(program)$ $program' = \langle editedProgram \rangle \frown program$ $editedProgram' = \emptyset$ $rep! = ok$

<i>NoDifferences</i>
$\Xi Program$
$rep! : Msg$
$editedProgram = head(program)$ $rep! = error$

$\Delta == \Delta Ok \vee NoDifferences$

<i>DeltaCSVOk</i>
ΔCSV
<i>Delta</i>
$u? : USUARIO$
$n? : NOMBREARCHIVO$
$rep! : Msg$
$u? \in dom(users)$ $n? \in \{n \mid p \in editedPrograms \wedge n = p.1\}$ $u? \in \{u \mid p \in editedPrograms \wedge n? = p.1 \wedge u = p.2\}$ $lineabase(n?) = \theta Program$ $lineabase' = lineabase \oplus \{n? \leftrightarrow \theta Program'\}$ $editedProrgams' = editedPrograms \setminus \{(n?, u?)\}$ $users' = users$ $rep! = ok$

<i>ProgramNotEdited</i>
ΞCSV
$n? : NOMBREARCHIVO$
$rep! : Msg$
$n? \notin \{n \mid p \in editedPrograms \wedge n = p.1\}$ $rep! = error$

<i>UserCantDelta</i>
ΞCSV
$u? : USUARIO$
$rep! : Msg$
$u? \notin \{u \mid p \in editedPrograms \wedge n? = p.1 \wedge u = p.2\}$ $rep! = error$

$\Delta CSV == \Delta CSVOk \vee UnkownUser \vee ProgramNotEdited \vee UserCantDelta$

<i>DeleteCSV</i>
ΔCSV
$n? : NOMBREARCHIVO$
$n? \in dom(programs)$
$programs' = \{n?\} \triangleleft programs$
$editedPrograms' = \{t \mid t \in editedPrograms \wedge t.1 \neq n?\}$
$users' = users$

9. Parcial 2021

[USUARIO, PRODUCTO]

Cantidad ::= \mathbb{N}

Msg ::= ok | error

<i>Carrito</i>
$productos : PRODUCTO \rightarrow Cantidad$

<i>CarritoInit</i>
<i>Carrito</i>
$productos = \emptyset$

<i>AgregarProductoOk</i>
$\Delta Carrito$
$p? : Producto$
$c? : Cantidad$
$rep! = Msg$
$p? \notin dom(productos)$
$productos' = productos \cup \{p? \mapsto c?\}$
$rep! = ok$

<i>ProductoExistente</i>
$\Xi Carrito$
$p? : Producto$
$rep! = Msg$
$p? \in dom(productos)$
$rep! = error$

AgregarProducto == AgregarProductoOk \vee ProductoExistente

<i>EditarProductoOk</i>
$\Delta Carrito$
$p? : Producto$
$c? : Cantidad$
$rep! = Msg$
$p? \in dom(productos)$
$productos' = productos \oplus \{p? \mapsto c?\}$
$rep! = ok$

<i>ProductoInexistente</i>
$\exists \text{Carrito}$ $p? : \text{Producto}$ $\text{rep!} = \text{Msg}$
$p? \notin \text{dom}(\text{productos})$ $\text{rep!} = \text{error}$

$\text{EditarProducto} == \text{EditarProductoOk} \vee \text{ProductoInexistente}$

<i>EliminarProductoOk</i>
$\Delta \text{Carrito}$ $p? : \text{Producto}$ $\text{rep!} = \text{Msg}$
$p? \in \text{dom}(\text{productos})$ $\text{productos}' = \{p?\} \triangleleft \text{productos}$ $\text{rep!} = \text{ok}$

$\text{EliminarProducto} == \text{EliminarProductoOk} \vee \text{ProductoInexistente}$

<i>Comercio</i>
$\text{stock} : \text{PRODUCTO} \rightarrow \text{Cantidad}$ $\text{usuarios} : \text{USUARIO} \rightarrow \text{Carrito}$

<i>ComercioInit</i>
<i>Comercio</i>
$\text{stock} = \emptyset$ $\text{usuarios} = \emptyset$

<i>ComercioInv</i>
<i>Comercio</i>
$\forall c \in \text{rango}(\text{usuarios}), p \in \text{dom}(\text{stock})$ $\sum \text{stock}(p) \geq \sum c.\text{productos}(p)$ $\forall c \in \text{rango}(\text{usuarios}), \text{dom}(c.\text{productos}) \subseteq \text{dom}(\text{stock})$

<i>CarritoAComercio</i>
$\Delta \text{Comercio}$ $\Delta \text{Carrito}$ $u? : \text{USUARIO}$ $p? : \text{PRODUCTO}$
$u? \in \text{dom}(\text{usuarios})$ $p? \in \text{dom}(\text{stock})$ $\text{usuarios}(u?) = \Theta \text{Carrito}$ $\text{usuarios}' = \text{usuarios} \oplus \{u? \rightarrow \theta \text{Carrito}'\}$ $\text{stock}' = \text{stock}$

$AgregarProductoComercioOk$ $\Delta Comercio$ $AgregarProducto$ $u? : USUARIO$ $p? : PRODUCTO$ $c? : Cantidad$ $rep! : Msg$
$u? \in dom(usuarios)$ $p? \in dom(stock)$ $c? \leq stock(p?)$ $usuarios(u?) = \theta Carrito$ $stock' = stock \oplus \{p? \rightarrow (stock(p?) - c?)\}$ $usuarios' = usuarios \oplus \{u? \rightarrow \theta Carrito'\}$ $rep! = ok$

$UsuarioInexistente$ $\Xi Comercio$ $u? : USUARIO$ $rep! : Msg$
$u? \notin dom(usuarios)$ $rep! = error$

$ProductoInexistente$ $\Xi Comercio$ $p? : PRODUCTO$ $rep! : Msg$
$p? \notin dom(stock)$ $rep! = error$

$CantidadInvalida$ $\Xi Comercio$ $p? : PRODUCTO$ $c? : Cantidad$ $rep! : Msg$
$p? \in dom(stock)$ $c? > stock(p?)$ $rep! = error$

$AgregarProductoComercio == AgregarProductoComercioOk \vee UsuarioInexistente \vee ProductoInexistente \vee$
 $CantidadInvalida$

EditarProductoComercioOk _____

$\Delta Comercio$

EditarProducto

$u? : USUARIO$

$p? : PRODUCTO$

$c? : Cantidad$

$rep! : Msg$

$u? \in dom(usuarios)$

$p? \in dom(stock)$

$c? \leq stock(p?) + ((usuarios(u?).productos(p?)) - c?)$

$usuarios(u?) = \theta Carrito$

$stock' = stock \oplus \{p? \mapsto stock(p?) + ((usuarios(u?).productos(p?)) - c?)\}$

$usuarios' = usuarios \oplus \{u? \mapsto \theta Carrito'\}$

$rep! = ok$

CantidadAActualizarInvalida _____

$\Xi Comercio$

$u? : USUARIO$

$p? : PRODUCTO$

$c? : Cantidad$

$rep! : Msg$

$p? \in dom(stock)$

$c? > stock(p?) + ((usuarios(u?).productos(p?)) - c?)$

$rep! = error$

$EditarProductoComercio == EditarProductoComercioOk \vee UsuarioInexistente \vee ProductoInexistente \vee CantidadAActualizarInvalida$

EliminarProductoComercioOk _____

$\Delta Comercio$

EliminarProducto

$u? : USUARIO$

$p? : PRODUCTO$

$rep! : Msg$

$u? \in dom(usuarios)$

$p? \in dom(stock)$

$usuarios(u?) = \theta Carrito$

$stock' = stock \oplus \{p? \mapsto stock(p?) + usuarios(u?).productos(p?)\}$

$usuarios' = usuarios \oplus \{u? \mapsto \theta Carrito'\}$

$rep! = ok$

$EliminarProductoComercio == EliminarProductoComercioOk \vee UsuarioInexistente \vee ProductoInexistente$

AgregarProductosOk _____

$\Delta Comercio$

$ps? : PRODUCTO \mapsto Cantidad$

$rep! : Msg$

$dom(batch?) \not\subseteq dom(stock)$

$stock' = stock \cup ps?$

$usuarios' = usuarios rep! = ok$

<i>ProductosExistentes</i>	
$\Xi Comercio$	
$ps? : PRODUCTO \rightarrow Cantidad$	
$rep! : Msg$	
$dom(batch?) \subseteq dom(stock)$	
$rep! = error$	

$AgregarProductos == AgregarProductosOk \vee ProductosExistentes$

<i>FinalizarCompraOk</i>	
$\Delta Comercio$	
$u? : USUARIO$	
$rep! : Msg$	
$u? \in dom(usuarios)$	
$usuarios' = \{u?\} \triangleleft usuarios$	
$stock' = stock$	
$rep! = ok$	

$FinalizarCompra == FinalizarCompraOk \vee UsuarioInexistente$