



MÉTODOS FORMALES PARA INGENIERÍA DE SOFTWARE

Trabajo Práctico N° 1

Alloy

Segundo Cuatrimestre de 2017

Ejercicios

1. Indique el tamaño, aridad, dominio y rango de las siguientes relaciones:

- a) $\text{Name} = \{(G0), (A0), (A1)\}$
- b) $\text{addressBook} = \{(G0, A0), (G0, A1), (A0, D0), (A1, D1)\}$
- c) $r1 = \{(A0, B0, C0), (A3, B0, C1), (A0, B0, D0), (A3, B1, C1), (A1, B2, C0)\}$
- d) $r2 = \{(A1, B3, C1), (A3, B1, C1), (A0, B0, C0), (A2, B2, D0)\}$
- e) $r3 = \{(A3, B1, C1)\}$

2. Considere las siguientes relaciones y la definición de operadores de conjuntos para Alloy. Para cada inciso, indique si las operaciones debajo listadas son legales y, en caso de serlo, cuál es el resultado de realizar dichas operaciones.

- a) $\text{Target} = \{(G0), (A0), (A1), (D0), (D1), (D2)\}$
 $\text{Name} = \{(G0), (A0), (A1)\}$
 $\text{Alias} = \{(A0), (A1)\}$
 $\text{Group} = \{(G0)\}$
 $\text{Addr} = \{(D0), (D1), (D2)\}$
 $\text{addressBook} = \{(G0, A0), (G0, A1), (A0, D0), (A1, D1)\}$
 - $\text{Alias} + \text{Group} =$
 - $\text{Alias} \& \text{Target} =$
 - $\text{Name} - \text{Alias} =$
 - $\text{Target} - \text{Addr} =$
 - $\text{Target in Group} =$
 - $\text{addressBook} \& \text{Group} =$
 - $\text{Alias in Name} =$
 - $\text{Target} = \text{Group} + \text{Alias} =$
- b) $A = \{(A0), (A1), (A2), (A3)\}$
 $B = \{(B0), (B1), (B2)\}$
 $C = \{(C0), (C1)\}$
 $D = \{(D0)\}$
 $r1 = \{(A0, B0, C0), (A3, B0, C1), (A0, B0, D0), (A3, B1, C1), (A1, B2, C0)\}$
 $r2 = \{(A1, B3, C1), (A3, B1, C1), (A0, B0, C0), (A2, B2, D0)\}$
 $r3 = \{(A3, B1, C1)\}$

- $(A + B) \& \text{univ} =$
- $r2 - r1 =$
- $r1 \& r2 = r2 \& r1 =$
- $A + r1 =$
- $r3 \text{ in } r2 =$
- $A - B =$

3. Considere los siguientes conjuntos de átomos para las firmas A y B:

$A = \{(A0), (A1), (A2), (A3)\}$

$B = \{(B0), (B1), (B2)\}$

Para cada inciso, determine si la relación binaria r_i es *funcional*, *inyectiva*, *ambas*, o *ninguna*.

a) $r1 = \{(A2, B0), (A1, B1), (A3, B2), (A1, B2)\}$

b) $r2 = \{(A2, B2), (A1, B1)\}$

c) $r3 = \{(A0, B2), (A3, B0), (A0, B1)\}$

d) $r4 = \{(A1, B1), (A2, B1), (A3, B0), (A0, B0)\}$

e) $r5 = \{(A2, B0), (A3, B1), (A0, B2)\}$

4. Considere los operadores de unión (+), intersección (&) y diferencia (-) de conjuntos.

a) Al aplicar dichos operadores sobre dos relaciones binarias que poseen la propiedad de ser *funcionales*, el resultado ¿preserva la característica de ser una relación *funcional* (con respecto a las relaciones originales)?. Para aquellos casos en que su respuesta sea negativa, brinde un contraejemplo.

b) Al aplicar dichos operadores sobre dos relaciones binarias que poseen la propiedad de ser *inyectivas*, el resultado ¿preserva la característica de ser una relación *inyectiva* (con respecto a las relaciones originales)?. Para aquellos casos en que su respuesta sea negativa, brinde un contraejemplo.

c) Al aplicar dichos operadores sobre dos relaciones binarias que poseen la propiedad de ser *funcionales e inyectivas*, el resultado ¿preserva la característica de ser una relación *funcional e inyectiva* (con respecto a las relaciones originales)?. Para aquellos casos en que su respuesta sea negativa, brinde un contraejemplo.

5. Considere la siguiente definición de relaciones, las cuales proveen una representación para el manejo de múltiples libretas de direcciones:

$\text{Book} = \{(B0), (B1)\}$

$\text{Target} = \{(G0), (A0), (A1), (D0), (D1), (D2)\}$

$\text{Name} = \{(G0), (A0), (A1)\}$

$\text{Alias} = \{(A0), (A1)\}$

$\text{Group} = \{(G0)\}$

$\text{Addr} = \{(D0), (D1), (D2)\}$

$\text{address} = \{(B0, G0, A0), (B1, G0, A1), (B0, A0, D0),$
 $(B1, A1, D1), (B0, A1, D2), (B1, A0, D2)\}$

$\text{myAlias} = \{(A0)\}$

$\text{myAddr} = \{(D2)\}$

Para cada uno de los siguientes incisos, determine cuál es el resultado de evaluar la expresión allí listada.

- a) `Alias -> Addr =`
- b) `Book -> Name -> Target =`
- c) `Name -> Addr + Name -> Name =`
- d) `address & (Book -> Name -> Addr) =`
- e) `Book.address =`
- f) `address.Alias =`
- g) `Book.address.myAlias =`
- h) `myAlias.(Book.address) =`
- i) `myAlias[address.myAddr] =`
- j) `Book.address[myAlias] =`
- k) `address[Book] =`
- l) `~address =`
- m) `Alias.~address =`
- n) `^(Book.address) =`
- ñ) `^(Name -> Target) =`
- o) `*(Alias.address)`
- p) `Group <: (Book.address) =`
- q) `Addr <: Target =`
- r) `address :> myAlias =`
- s) `Name :> Group =`
- t) `iden ++ address[Book] =`
- u) `Book.address ++ (^ (Book.address) - Book.address) =`

6. Para cada uno de los siguientes modelos y grupos de instancias, indique cuáles de ellas son instancias válidas del modelo. Para cada instancia inválida, indique los errores existentes. Se recomienda utilizar la herramienta Alloy para obtener una visualización gráfica del modelo.

```
a) abstract sig Target {}
   sig Addr extends Target {}
   abstract sig Name extends Target {}
   sig Alias, Group extends Name {}
```

- `Target = {(D0), (A0), (A1)}`
- `Addr = {(D0)}`
- `Name = {(A0)}`
- `Alias = {(A0), (A1)}`
- `Group = {}`

- Target = {(D0), (D1) (A0), (A1), (A2), (G0)}
 Addr = {(D0), (D1)}
 Name = {(A0), (A1), (A2), (G0)}
 Alias = {(A0), (A1), (A2)}
 Group = {(G0)}

 - Target = {(D0), (A0), (G0)}
 Addr = {(D0)}
 Name = {(A0), (A1), (G0), (G1)}
 Alias = {(A0)}
 Group = {(G0)}

 - Target = {(A0), (A1), (A2), (G0), (G1)}
 Addr = {}
 Name = {(A0), (A1), (A2), (G0), (G1)}
 Alias = {(A0), (A1), (A2)}
 Group = {(G0), (G1)}

 - Target = {(D0), (D1), (A0), (A1)}
 Addr = {(D0), (D1)}
 Name = {(A0), (A1), (D0)}
 Alias = {(A0), (A1)}
 Group = {(D0)}
- b) abstract sig Person {siblings: Person}
 sig Man extends Person {}
 sig Woman extends Person {}
 sig Married in Person {}
- Person = {(P0), (P1), (P2)}
 Man = {(P1), (P2)}
 Woman = {(P0)}
 Married = {}
 siblings = {(P0,P2), (P2,P0), (P1,P2)}

 - Person = {(P0), (P1), (P2)}
 Man = {(P1), (P2)}
 Woman = {(P0), (P1)}
 Married = {(P1)}
 siblings = {}

 - Person = {(P0), (P1)}
 Man = {(P0)}
 Woman = {(P1)}
 Married = {(P1)}
 siblings = {(P0,P0)}

- Person = {(P0), (P1)}
Man = {(P0)}
Woman = {(P1)}
Married = {}
siblings = {(P0,P2)}

- Person = {(P0), (P1), (P2)}
Man = {(P0), (P2)}
Woman = {(P1)}
Married = {(P0), (P1)}
siblings = {(P1,P2), (P2,P0), (P0,P0)}

- Person = {(P0), (P1), (P2)}
Man = {(P2)}
Woman = {(P0), (P1)}
Married = {(P0), (P1)}
siblings = {(P1,P2), (P1,P0), (P0,P1), (P2,P1)}

c) sig PrimaryColor {}
sig SecondaryColor {}
one sig Red, Yellow, Blue extends PrimaryColor {}

- PrimaryColor = {(C1), (C2), (C3), (C4)}
SecondaryColor = {(C4), (C5)}
Red = {(C1)}
Yellow = {(C2)}
Blue = {(C3)}

- PrimaryColor = {(C1), (C2), (C3), (C4)}
SecondaryColor = {(C5)}
Red = {(C1)}
Yellow = {}
Blue = {(C3)}

- PrimaryColor = {(C1), (C2), (C3)}
SecondaryColor = {}
Red = {(C1)}
Yellow = {(C3)}
Blue = {(C2)}

- PrimaryColor = {(C1), (C2), (C3), (C4)}
SecondaryColor = {(C5), (C6)}
Red = {(C2)}
Yellow = {(C3)}
Blue = {(C3)}

d) sig Name, Addr {}
 sig Book {address: Name some -> lone Addr}

- Name = {(N0), (N1), (N2)}
 Addr = {(A0), (A1)}
 Book = {(B0)}
 address = {(B0, N1, A0), (B0, N2, A1), (B0, N0, A1)}
- Name = {(N0), (N1)}
 Addr = {(A0), (A1)}
 Book = {(B0)}
 address = {(B0, N0, A0)}
- Name = {(N0), (N1), (N2)}
 Addr = {(A0)}
 Book = {(B0), (B1)}
 address = {(B1, N0, A0), (B1, N2, A0)}
- Name = {(N0), (N1), (N2)}
 Addr = {(A0), (A1)}
 Book = {(B0), (B1)}
 address = {(B0, N2, A1), (B1, N0, A0), (B1, N2, A0), (B0, N1, A0)}
- Name = {(N0), (N1), (N2)}
 Addr = {(A0), (A1)}
 Book = {(B0), (B1)}
 address = {(B0, N2, A1), (B0, N0, A0), (B0, N2, A0)}

7. Para cada uno de los siguientes modelos, brinde al menos dos instancias válidas y, en caso de existir, dos instancias inválidas.

- a) sig S1 {r1: lone T}
- b) sig S2 {r2: one T}
- c) sig S3 {r3: T -> one U}
- d) sig S4 {r4: T lone -> U}
- e) sig S5 {r5: some T}
- f) sig S6 {r6: set T}
- g) sig S7 {r7: T set -> set U}
- h) sig S8 {r8: T one -> U}

8. Explique en lenguaje coloquial el significado de cada una de las siguientes declaraciones, a partir del modelo que se ilustra a continuación. Asimismo, para cada inciso, escriba en Alloy una declaración alternativa que posea el mismo significado coloquial.

```
abstract sig Target {}
sig Addr extends Target {}
abstract sig Name extends Target {}
sig Alias, Group extends Name {}
sig Book {address: Name -> Target}
```

- $\text{all } b: \text{Book} \mid \text{some } b.\text{address}$
- $\text{all } n: \text{Name} \mid \text{some } b: \text{Book} \mid \text{none } b.\text{address}[n]$
- $\#(\text{Name}.\text{b.address}) - \text{Name} > 0$
- $\{n: \text{Name} \mid \text{no } (n.(\text{Book.address}) \ \& \ \text{Addr})\}$
- $\text{all } b: \text{Book} \mid \text{no } n: \text{Name} \mid n \text{ in } ^b.\text{address}[n]$
- $\text{all } n: \text{Name} \mid \text{no disj } t, t': \text{Target} \mid (t + t') \text{ in } n.(\text{Book.address})$
- $\text{all } b: \text{Book} \mid \text{all } n: \text{Name} \mid \text{no disj } t, t': \text{Target} \mid (t + t') \text{ in } n.(b.\text{address})$

9. Escriba el siguiente modelo en la herramienta Alloy:

```
abstract sig Person {
    children: set Person,
    siblings: set Person
}

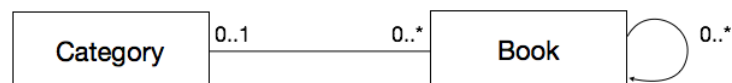
sig Man, Woman extends Person {}

sig Married in Person {
    spouse: one Married
}
```

- Analice el meta-modelo generado por la herramienta Alloy. ¿Parece correcto?
- Ejecute el siguiente comando en Alloy: `run {} for 3`. ¿Cuál es el significado intuitivo de este comando?
- Analice las primeras tres instancias generadas por Alloy en el inciso anterior. ¿Qué irregularidades detecta?
- Defina una restricción para el modelo que permita identificar los padres de una persona.
- Modifique el modelo brindado, añadiendo las restricciones indicadas en cada ítem:
 - Ninguna persona puede ser su propio padre/hijo.
 - Ninguna persona puede ser su propio ancestro.
 - Ninguna persona puede tener más de una madre, ni más de un padre.
 - Los hermanos de una persona son aquellas personas que poseen un padre en común (es decir, considere la existencia de medio-hermanos).

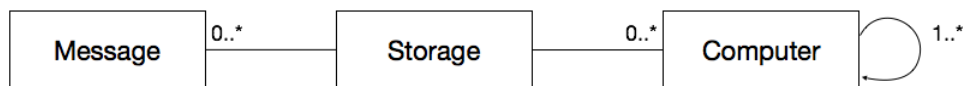
- f) A partir del modelo resultante de añadir las restricciones indicadas hasta el momento, la relación *siblings* ¿es simétrica? Defina una restricción para chequear esto en Alloy.
- g) A partir del modelo resultante de añadir las restricciones indicadas hasta el momento, la relación *siblings* ¿admite que una persona sea hermana de sí misma? Defina una restricción para chequear esto en Alloy. ¿Cómo modificaría el modelo para asegurar que esta restricción se cumpla?
- h) Defina en Alloy una restricción para determinar si dos personas son parientes de sangre. (Observación: Considere que dos personas son parientes de sangre si poseen un ancestro en común).
- i) A partir del modelo resultante de añadir las restricciones indicadas hasta el momento ¿es posible que dos parientes de sangre tengan hijos en común? Defina la restricción necesaria para chequear esto en Alloy. Modifique el modelo para garantizar que dicha restricción se cumpla.
- j) A partir del modelo resultante de añadir las restricciones indicadas hasta el momento, ¿la relación *spouse* es simétrica? ¿Es posible que una persona esté casada con más de una persona? ¿Es posible que una persona esté casada consigo misma? Defina restricciones en Alloy para chequear esto. ¿Cómo modificaría el modelo para asegurar la primera restricción y evitar la segunda y tercera?
- k) A partir del modelo resultante de añadir las restricciones indicadas hasta el momento ¿es posible que una persona esté casada con un familiar de sangre? Defina en Alloy una restricción para chequear esto. ¿Cómo modificaría el modelo para evitar dicha situación?
- l) A partir del modelo resultante de añadir las restricciones del ejercicio 9e) ¿es posible que haya personas que no posean padre ni madre? Defina una restricción que permita identificar si una persona se encuentra en tal situación, y otra restricción que permita obtener el conjunto de personas en tal condición.
- m) Defina comandos **run** para crear, en caso de ser posible, instancias del modelo que cumplan con las siguientes restricciones:
- Crear una instancia en la que las relaciones *spouse* y *siblings* no sean vacías.
 - Crear una instancia con dos parejas casadas *diferentes*.
 - Crear una instancia con a lo sumo un átomo en cada signatura de alto nivel y con un hombre casado.
 - Crear una instancia con a lo sumo dos átomos en cada signatura de alto nivel y con un hombre casado.
 - Crear una instancia donde haya a lo sumo una persona, alguna mujer y ningún hombre.

10. Considere el siguiente diagrama de clases, el cual modela una colección de libros:



Cada libro puede pertenecer a una categoría, la cual está conformada por todos los libros que pertenecen a ella. Asimismo, cada libro puede clasificar a un conjunto de libros como similares a él. Escriba un modelo en Alloy para representar este dominio, añadiendo las restricciones que considere necesarias.

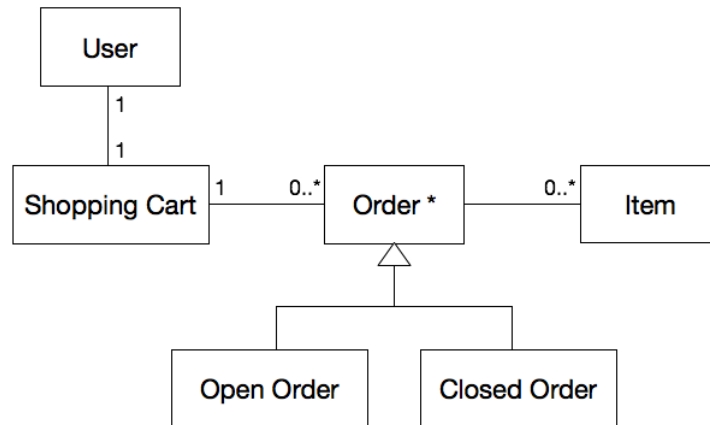
11. Considere el siguiente diagrama de clases, el cual modela la existencia de un almacenamiento de mensajes, que gestiona el envío y recepción de mensajes de computadoras que tienen acceso autorizado a él. Un mensaje posee una computadora de origen, una computadora destino, y un texto que representa el contenido del mensaje. Una computadora puede enviar al almacenamiento un mensaje cuya destinataria sea una de sus computadoras vecinas. Asimismo, una computadora puede recibir del almacenamiento un mensaje del cual sea destinataria. El almacenamiento tiene limitaciones de espacio en cuanto a la cantidad de mensajes que puede albergar (máximo 50 mensajes). De manera similar, el almacenamiento no puede gestionar pedidos de más de 5 computadoras autorizadas.



Escriba un modelo en Alloy para representar este dominio, añadiendo las restricciones que considere necesarias. Asimismo, deberá definir un conjunto de restricciones que permitan verificar que el modelo admite el siguiente comportamiento:

- Una computadora puede enviar (añadir) al almacenamiento un mensaje para sí misma o para una computadora vecina.
- Una computadora puede recibir (retirar) del almacenamiento un mensaje que la tenga como destinataria.
- Una computadora puede consultar al almacenamiento (sin retirar) todos los mensajes que la tengan como destinataria.
- Una computadora puede consultar al almacenamiento (sin retirar) todos los mensajes que la tengan como destinataria y que hayan sido enviados por una de sus computadoras vecinas.
- Una computadora puede recibir (retirar) del almacenamiento todos los mensajes que la tengan como destinataria y que hayan sido enviados por una de sus computadoras vecinas.
- Una computadora puede recibir (retirar) del almacenamiento todos los mensajes que la tengan como destinataria.
- Agregar una computadora al conjunto de computadoras autorizadas de un almacenamiento.
- Remover una computadora del conjunto de computadoras autorizadas de un almacenamiento. Dicha computadora puede ser removida únicamente si el almacenamiento no posee mensajes que la tengan como destinataria.
- Remover una computadora del conjunto de computadoras autorizadas de un almacenamiento. Al remover dicha computadora, el almacenamiento mantiene todos los mensajes que la tengan como destinataria.
- Remover una computadora del conjunto de computadoras autorizadas de un almacenamiento. Al remover dicha computadora, el almacenamiento elimina todos los mensajes que la tengan como destinataria.

12. Considere el siguiente diagrama de clases, el cual modela un sistema en el que usuarios pueden hacer uso de carritos de compras para efectuar diversas órdenes con el objetivo de comprar ítems:



Escriba un modelo en Alloy para representar este dominio, añadiendo las restricciones que considere necesarias. Asimismo, deberá definir un conjunto de restricciones que permitan verificar que el modelo admite el siguiente comportamiento:

- Añadir un ítem a una orden.
- Remover un ítem de una orden.
- Añadir una nueva orden para un usuario.
- Remover una orden de un usuario.
- Cerrar una orden de un usuario.
- Despachar el carrito de compras de un usuario.