

MÉTODOS FORMALES PARA INGENIERÍA DE SOFTWARE

Segundo Cuatrimestre de 2023

Trabajo Práctico N° 1: **Conjuntos y Relaciones****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 cada inciso, indique si las operaciones debajo listadas son legales y, en caso de serlo, cuál es el resultado de realizar dichas operaciones.

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

3. Considere los siguientes conjuntos de elementos A y B:

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

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

Determine si la relación binaria r_i definida sobre A y B es *funcional*, *inyectiva*, *ambas*, o *ninguna*:

a) $r_1 = \{(A2, B0), (A1, B1), (A3, B2), (A1, B2)\}$

b) $r_2 = \{(A2, B2), (A1, B1)\}$

c) $r_3 = \{(A0, B2), (A3, B0), (A0, B1)\}$

d) $r_4 = \{(A1, B1), (A2, B1), (A3, B0), (A0, B0)\}$

e) $r_5 = \{(A2, B0), (A3, B1), (A0, B2)\}$

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. 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 visualizar gráficamente del modelo.

a)

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

■ $\text{Target} = \{(D0), (A0), (A1)\}$
 $\text{Addr} = \{(D0)\}$
 $\text{Name} = \{(A0)\}$
 $\text{Alias} = \{(A0), (A1)\}$
 $\text{Group} = \{\}$

■ $\text{Target} = \{(D0), (D1), (A0), (A1), (A2), (G0)\}$
 $\text{Addr} = \{(D0), (D1)\}$
 $\text{Name} = \{(A0), (A1), (A2), (G0)\}$
 $\text{Alias} = \{(A0), (A1), (A2)\}$
 $\text{Group} = \{(G0)\}$

■ $\text{Target} = \{(D0), (A0), (G0)\}$
 $\text{Addr} = \{(D0)\}$
 $\text{Name} = \{(A0), (A1), (G0), (G1)\}$
 $\text{Alias} = \{(A0)\}$
 $\text{Group} = \{(G0)\}$

- 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)}

```

6. Ilustre dos instancias válidas y (en caso de existir) dos instancias inválidas para el siguiente modelo de Alloy, considerando que la definición de las firmas T y U es la misma para todos los incisos, y que la definición de la firma S es la indicada en cada inciso:

```
sig T {}
```

```
sig U {}
```

a) sig S {r: lone T}

b) sig S {r: one T}

c) sig S {r: T -> one U}

d) sig S {r: T lone -> U}

e) sig S {r: some T}

f) sig S {r: set T}

g) sig S {r: T set -> set U}

h) sig S {r: T one -> U}

7. Considere el siguiente modelo en Alloy:

```
abstract sig Target {}
```

```
abstract sig Name extends Target { addressBook: some Target }
```

```
sig Alias extends Name {}
```

```
sig Group extends Name {}
```

```
sig Addr extends Target {}
```

Dada la siguiente instancia del modelo:

```
Target = {(Addr0), (Addr1), (Addr2), (Alias0), (Alias1), (Group0)}
```

```
Name = {(Alias0), (Alias1), (Group0)}
```

```
Alias = {(Alias0), (Alias1)}
```

```
Group = {(Group0)}
```

```
Addr = {(Addr0), (Addr1), (Addr2)}
```

```
addressBook = {(Alias0,Addr0), (Alias0,Addr1), (Alias0,Addr2), (Alias0,Alias0),
(Alias0,Alias1), (Alias0,Group0), (Alias1,Addr0), (Alias1,Addr1), (Alias1,Addr2),
(Alias1,Alias0), (Alias1,Alias1), (Alias1,Group0), (Group0,Addr0), (Group0,Addr1),
(Group0,Addr2), (Group0,Alias0), (Group0,Alias1), (Group0,Group0)}
```

- a) Para cada una de las siguientes operaciones, indique el resultado de evaluar dicha operación sobre la instancia arriba listada:

- Alias + Group
- Alias & Target
- Name - Alias
- Target in Group
- Alias in Name
- Target = Group + Alias
- Alias -> Addr
- Name -> Target
- Name -> Addr + Name -> Name
- addressBook & (Name -> Addr)
- addressBook.Alias
- addressBook[Name]

- `Name.addressBook`
- `~addressBook`
- `Alias.~addressBook`
- `^addressBook`
- `*addressBook`
- `*addressBook - ^addressBook`
- `^(Name -> Target)`
- `Group <: addressBook`
- `Addr <: Target`
- `Name :> Group`
- `iden ++ addressBook`

- b) Defina un comando `run` en Alloy con las restricciones necesarias para generar la instancia del modelo arriba listada.
- c) Verifique los resultados obtenidos en el inciso a) de este ejercicio utilizando el *evaluador* de Alloy para determinar el resultado de las operaciones aplicadas sobre la instancia generada con la herramienta en el inciso b).

8. Considere el siguiente modelo definido en Alloy.

```
sig Persona { lectura: set Idioma,
              escritura: set Idioma,
              oido: set Idioma,
              habla: set Idioma }
```

```
abstract sig Idioma {}
```

```
one sig Espanol, Ingles, Frances, Italiano, Portugues, Aleman, Chino extends Idioma {}
```

- a) Para cada una de las siguientes expresiones en lenguaje coloquial, escriba la expresión correspondiente en Alloy:
 - Toda persona posee al menos una de las cuatro habilidades, la cual puede corresponder a cualquier idioma.
 - Si una persona sabe escribir un idioma, entonces sabe leerlo.
 - Ninguna persona sabe hablar más de cinco idiomas.
 - Las personas que hablan un idioma saben leerlo o escribirlo.
 - Sólo dos personas saben algo de Chino (en cualquiera de sus habilidades).
- b) Escriba comandos en Alloy para generar instancias del modelo que posean las siguientes características:
 - Hay exactamente siete personas que hablan Italiano.
 - Existe alguna persona que no sabe nada de idiomas.
 - Hay al menos dos personas que hablan todos los idiomas.
 - Todas las personas escriben Portugués y Alemán.
 - Todos los idiomas son sabidos por alguien (en al menos una de sus habilidades).

OBSERVACIÓN: De ser necesario deberá agrandar el `scope`.

- c) Añada las expresiones del inciso a) como restricciones del modelo e intente generar nuevamente las instancias indicadas en el inciso b). Para aquellas que no sea posible, explique los motivos por los cuales ocurre esto.