SOEN331: Introduction to Formal Methods for Software Engineering Assignment 4 on algebraic specifications

Tarek Ait Hamouda (40044119), Abhijit Gupta (40066502), Ethel Narra Pangan (40061530)

March 29, 2019

Spec: Location;

Sort: Location;

Imports:String, Point

Description: A location contains a description

Operations:

newLocation: String × Point → Location;

setDescription: String × Location → Location;

getDescription : Location → String;

setPoint : Point × Location → Location;

getPoint : Location → Point;

Variables:

newDesc, d: String; newPoint, p: Point;

Axioms:

[A1] getDescription(newLocation(d, p)) = d;

[A2] getDescription(setDescription(newDesc, newLocation(d, p))) = newDesc;

[A3] getDescription(setPoint(newPoint, newLocation(d,p))) = d;

[A4] getPoint(newLocation(d, p)) = p;

[A5] getPoint(setPoint(newPoint, newLocation(d,p))) = newPoint;

[A6] getPoint(setDescription(newDesc, newLocation(d,p))) = p;

[A7] setDescription(newDesc, newLocation(d,p)) = newLocation(newDesc, p);

[A8] setDescription(getDescription(newLocation(d,p)), newLocation(d,p)) = newLocation(d,p);

[A9] setPoint(newPoint, newLocation(d,p)) = newLocation(d, newPoint);

[A10] setPoint(getPoint(newLocation(d,p)), newLocation(d,p)) = newLocation(d, p);

Spec: Map(Location);

Sort: Map;

Imports: String, Point, Boolean, Location;

Description: A map contains locations

Operations:

newmap: → Map;

addlocation: Map × Location → Map;

deletelocation : Map × String → Map;

containsdescription : Map × String → Boolean;

containspoint : Map × Point → Boolean;

findlocation : Map × String → Point;

isempty : Map → Boolean;

clear : Map → Map;

Variables:

d: String; p, q: Point; loc: Location; map: Map

Axioms:

[A1] isempty(newmap) = true;

[A2] isempty(clear(map)) = true;

[A3] containsdescription(addlocation(map, loc), getDescription(loc)) = true;

[A4] addlocation(map, newLocation(d,p)) and containdescription(map, d) and not(p == findlocation(map, d) → findlocation(map, getDescription(loc)) = p;

[A5] deletelocation(addlocation(addlocation(newmap, newLocation(d, p)),newLocation(d,q)),d)

→ newmap

[A6]

[A7] findlocation(map, getDescription(loc)) = getPoint(loc)

[A8] findlocation(newmap) = undefined;

[A9] deletelocation(newmap) = undefined;

preconditions:

pre : deletelocation(map, d) = not isempty(map);

pre : findlocation(map, d) = not isempty(map);

end: