

Algoritmos y Estructura de Datos I

Segundo cuatrimestre de 2012

30 de Agosto de 2012

TPE De Hoteles y Pasajeros v1.0

1. Tipos

```
tipo Fecha =  $\mathbb{Z}$ ;
tipo DNI =  $\mathbb{Z}$ ;
tipo Dinero =  $\mathbb{Z}$ ;
tipo Cadena = String;
tipo CheckIn = (DNI, Fecha);
tipo CheckOut = (DNI, Fecha);
tipo TipoHabitacion = Simple, Doble, Triple, Cuadruple;
tipo Accesorio = Jacuzzi, LCD, PS3, DVD, Pelotero, Inflable;
```

2. Reserva

```
tipo Reserva {
  observador documento (r: Reserva) : DNI;
  observador fechaDesde (r: Reserva) : Fecha;
  observador fechaHasta (r: Reserva) : Fecha;
  observador tipo (r: Reserva) : TipoHabitacion;
  observador confirmada (r: Reserva) : Bool;

  invariante NoAntesDeDespues :  $fechaHasta(r) > fechaDesde(r)$ ;
}
```

3. Habitación

```
tipo Habitacion {
  observador numero (h: Habitacion) :  $\mathbb{Z}$ ;
  observador tipo (h: Habitacion) : TipoHabitacion;
  observador accesorios (h: Habitacion) : [Accesorio];

  invariante sinAccesoriosRepetidos :  $sinRepetidos(accesorios(h))$ ;
  invariante accesoriosOrdenada :  $ordenada(accesorios(h))$ ;
}

problema pretencionesDePopStar (as: [Accesorio], hs: [Habitacion]) = result : [Habitacion] {
}
```

4. Hotel

```
tipo HotelBoutique {
  observador nombre (h: Hotel) : Nombre;
  observador cadena (h: Hotel) : Cadena;
  observador huespedes (h: Hotel) : [DNI];
  observador habitaciones (h: Hotel) : [Habitacion];
  observador ingresos (h: Hotel) : [(CheckIn, Habitacion)];
  observador salidas (h: Hotel) : [CheckOut];
  observador reservas (h: Hotel) : [Reserva];
  observador tarifaHabitacionXDia (h: Hotel) : [(TipoHabitacion, Dinero)];
  observador precioAccesorio (h: Hotel) : [(Accesorio, Dinero)];
  invariante habitacionesValidas :  $(\forall c \leftarrow \text{ingresos}(h)) \text{snd}(c) \in \text{habitaciones}(h)$ ;
  invariante siEstanNoSeFueron :  $\forall d \leftarrow \text{huespedes}(h) |\text{ingresosDe}(h, d)| == |\text{salidasDe}(h, d)| + 1$ ;
  invariante siSeVaEntro : ...;
  invariante estanAlMenosUnDia : ...;
  invariante noEntranDosVeces : ...;
  invariante reservasValidas :  $(\forall r \leftarrow \text{reservas}(h)) \text{existeUnaHabitacionDelTipo}(h, \text{tipo}(r))$ ;
  invariante sinTarifasRepetidas :  $\text{sinRepetidos}([\text{prm}(t) | t \leftarrow \text{tarifaHabitacionXDia}(h)])$ ;
  invariante sinPreciosRepetidos :  $\text{sinRepetidos}([\text{prm}(p) | p \leftarrow \text{precioAccesorio}(h)])$ ;
  invariante tarifasPositivas :  $(\forall t \leftarrow \text{tarifaHabitacionXDia}(h)) \text{snd}(t) > 0$ ;
  invariante preciosPositivos :  $(\forall p \leftarrow \text{precioAccesorio}(h)) \text{snd}(p) > 0$ ;
  invariante noValeAcaparar :  $(\forall r1, r2 \leftarrow \text{reservas}(h), \text{tipo}(r1) == \text{tipo}(r2) \wedge \text{documento}(r1) == \text{documento}(r2) \wedge \text{fechaDesde}(r1) == \text{fechaDesde}(r2)) r1 == r2$ ;
}

problema sobreReservado (h: Hotel, f: Fecha) = result : Bool {
}

problema registrarHuesped (h: Hotel, d: DNI, f: Fecha, a: Habitacion) {
}

problema desRegistrarHuesped (h: Hotel, d: DNI, f: Fecha) {
}

problema huespedesConPalabra (h: Hotel) = result : [DNI] {
}

problema calcularCuenta (h: Hotel, i: CheckIn, o: CheckOut, hb: Habitacion) = result : Dinero {
}

problema reservasSolapadas (h: Hotel, d: DNI) = result : Bool {
}
```

5. MinisterioDeTurismo

```
tipo MinisterioDeTurismo {  
  observador secretarias (m: MinisterioDeTurismo) : [Provincia];  
  observador registro (m: MinisterioDeTurismo, p: Provincia ) : [Hotel];  
    requiere  $p \in secretarias(m)$ ;  
  observador cadenasDeHoteles (m: MinisterioDeTurismo) : [[Hotel]];  
  invariante sinHotelesRepetidas :  $(\forall xs \leftarrow cadenasDeHoteles(m)) sinRepetidos(xs)$ ;  
  invariante sinCadenasRepetidas :  $sinRepetidos(xs)$ ;  
  invariante sinProvinciasRepetidas :  $sinRepetidos(secretarias(m))$ ;  
  invariante cadenasBienFormadas : ...;  
  invariante sinNombresRepetidosEnCadenas : ...;  
  invariante hotelesConsistentes : ...;  
}  
  
problema cadenasAmarretas (m: MinisterioDeTurismo) = result : [Cadena] {  
}  
  
problema fusionAutorizada (m: MinisterioDeTurismo, c1: Cadena, c2: Cadena) {  
}
```