



# Trabajo práctico 1

## Especificación y WP

7 de septiembre de 2024

Algoritmos y Estructuras de Datos - DC - UBA

### Grupo AJMS

| Integrante        | LU     | Correo electrónico        |
|-------------------|--------|---------------------------|
| Ferechian, Matías | 693/23 | matifere@gmail.com        |
| Nestmann, Sofía   | 366/23 | sofianestmann@gmail.com   |
| Mirasson, Javier  | 594/23 | javierestebanmn@gmail.com |
| Ramirez, Ana      | 931/23 | correodeanar@gmail.com    |



**Facultad de Ciencias Exactas y Naturales**  
Universidad de Buenos Aires

Ciudad Universitaria - (Pabellón I/Planta Baja)

Intendente Güiraldes 2610 - C1428EGA

Ciudad Autónoma de Buenos Aires - Rep. Argentina

Tel/Fax: (+54 +11) 4576-3300

<http://www.exactas.uba.ar>

# 1. Especificación

## 1.1. grandesCiudades

```
proc grandesCiudades (in ciudades : seq⟨Ciudad⟩) : seq⟨Ciudad⟩{  
  requiere {true}  
  asegura { (∀i : ℤ) (  
    (0 ≤ i < |ciudades|) ∧ ((ciudades[i] ∈ res) →L (ciudades[i]1 > 50000))  
  ) }  
}
```

## 1.2. sumaDeHabitantes

```
proc sumaDeHabitantes (in menoresDeCiudades : seq⟨Ciudad⟩, in mayoresDeCiudades : seq⟨Ciudad⟩) : seq⟨Ciudad⟩{  
  requiere { (|menoresDeCiudades| = |mayoresDeCiudades|) ∧L ((∀i, j : ℤ≥0) (  
    0 ≤ i, j < |menoresDeCiudades| ∧ menoresDeCiudades[i]0 = mayoresDeCiudades[j]0  
  ) }  
  asegura { (∀m, n : ℤ) (  
    (0 ≤ m, n < |menoresDeCiudades|) ∧ (menoresDeCiudades[n]0 = mayoresDeCiudades[m]0) ∧  
    (⟨menoresDeCiudades[n]0, menoresDeCiudades[n]1 + mayoresDeCiudades[m]1⟩ ∈ res)) }  
}
```

## 1.3. hayCamino

```
proc hayCamino (in distancias : seq⟨seq⟨ℤ⟩⟩, in desde : ℤ, in hasta : ℤ) : Bool{  
  requiere { ((∀i, j : ℤ) (  
    (0 ≤ i, j, desde, hasta < √|distancias|) ∧L ((i = j) → (distancias[i][j] = 0)) ∧ (distancias[i][j] = distancias[j][i])) }  
  asegura { res = true ↔ (∃p : seq⟨ℤ⟩) (  
    (p[0] = desde) ∧ (p[|p| - 1] = hasta) ∧ (∀k : ℤ) (  
      (0 ≤ k < |p| - 1) → distancias[p[k]][p[k+1]] > 0 ) ) }  
}
```

## 1.4. cantidadCaminosNSaltos

```
proc cantidadCaminosNSaltos (inout conexión : seq⟨seq⟨ℤ⟩⟩, in n : ℤ) : {  
  requiere { true }  
  asegura { true }  
}
```

## 1.5. caminoMinimo

```
proc hayCamino (in origen : ℤ, in destino : ℤ, in distancias : seq⟨seq⟨ℤ⟩⟩) : seq⟨ℤ⟩ {  
  requiere { true }  
  asegura { true }  
}
```

# 2. Demostraciones de correctitud