

Práctica 2 de Intro a la programación

Nad Garraz

Actualizado: 07/04/2024

1 Formato de especificaciones semiformales:

Nota: Contrato \rightarrow especificaciones. Contrato entre 2 partes.

- Contrato: el programador escribe un programa P tal que si el usuario suministra datos que hacen verdadera la precondition, entonces P termina en una cantidad finita de pasos retornando un valor que hace verdadera la postcondición.
 - El programa P es **correcto** para la especificación dada por la precondition y la postcondición exactamente cuando se cumple el contrato.
 - Si el usuario no cumple la precondition y P se cuelga o no cumple la postcondición...
 - ¿El usuario tiene derecho a quejarse? **Creería que no.**
 - ¿Se cumple el contrato? **No, se debe cumplir precondition para que se cumpla el contrato.**
 - Si el usuario cumple la precondition y P se cuelga o no cumple la postcondición...
 - ¿El usuario tiene derecho a quejarse? **Diría que sí.**
 - ¿Se cumple el contrato? **No, porque debería terminar el programa cumpliendo la postcondición.**
-

Ejercicio 1.

a)

| entrada x | salida res |
|-------------|--------------|
| 0 | 2 |
| 2 | 4 |
| -2 | -4 |

b) requiere: $\{x \text{ es positivo}\}$

asegura: $\{resultado \text{ es la raíz cuadrada de } x\}$

| entrada x | salida res |
|-------------|--------------|
| 1 | 1 |
| 4 | -2 |
| 25 | 5 |

Consultar por el tema de la definición de raíz cuadrada en este ejercicio es justamente la definicion que está en el inciso (b.). Es medio circular el asunto. Supongo que está bien definida de ahora en más. Y que no vale que la $res < 0$

c)

| entrada x | salida res |
|-------------|--------------|
| -1 | 1 |
| 1.4 | 1 |
| 1.6 | 2 |
| 1.5 | 2 1 |

Consultar por el tema del redondeo. Al estar igual distancia el 1 y 2, se cumple la poscondición? Puedo tener ese tipo de ambigüedad?

Al parecer ya sea 1 o 2, funcionaría como respuesta. Dado que cumplen el asegura.

d)

| entrada s | salida res |
|---------------------------|----------------------------------|
| $\langle 2, 4, 9 \rangle$ | $\langle \sqrt{2}, 2, 3 \rangle$ |
| $\langle 25 \rangle$ | $\langle 5 \rangle$ |

e)

| entrada s | salida res |
|---------------------------|----------------------------------|
| $\langle 2, 4, 9 \rangle$ | $\langle 2, \sqrt{2}, 3 \rangle$ |
| | $\langle 2, 3, \sqrt{2} \rangle$ |
| | $\langle 3, 2, \sqrt{2} \rangle$ |
| | $\langle 3, \sqrt{2}, 2 \rangle$ |
| | $\langle \sqrt{2}, 2, 3 \rangle$ |
| | $\langle \sqrt{2}, 3, 2 \rangle$ |
| $\langle 25 \rangle$ | $\langle 5 \rangle$ |

f)

| entrada s | salida res |
|---------------------------|--|
| $\langle 2, 4, 9 \rangle$ | $\langle 2, 2, 2 \rangle$ |
| | $\langle \sqrt{2}, \sqrt{2}, \sqrt{2} \rangle$ |
| | $\langle 3, 3, 3 \rangle$ |
| | $\langle 2, 9, 2 \rangle$ |
| | $\langle 3, 2, 4 \rangle$ |
| | $\langle \sqrt{2}, 2, 9 \rangle$ |
| | $\langle \sqrt{2}, 9, 4 \rangle$ |
| $\langle 25 \rangle$ | $\langle 5 \rangle$ |

g)

| entrada s | salida res |
|----------------------------|------------------------------|
| $\langle -2, 4, 9 \rangle$ | $\langle 2, 3 \rangle$ |
| | $\langle 3, 2 \rangle$ |
| | $\langle 2, 3, 2, 3 \rangle$ |
| $\langle 25 \rangle$ | $\langle 5 \rangle$ |

h)

| entrada s | salida res |
|---------------------------|---|
| $\langle 2, 4, 9 \rangle$ | $\langle \sqrt{2}, 2, 3 \rangle$ $\langle \sqrt{2}, 2, 3, e^{i2\pi} \rangle$ |
| $\langle 25 \rangle$ | $\langle 5 \rangle$ |

i)

| entrada s | salida res |
|-------------------------------|----------------------------------|
| $\langle 2, 4, 9 \rangle$ | $\langle \sqrt{2}, 2, 3 \rangle$ |
| $\langle 25 \rangle$ | $\langle 5 \rangle$ |
| $\langle \sqrt{3}, 4 \rangle$ | $\langle 100, 2 \rangle$ |

Ejercicio 2.

1. En los problemas `raicesCuadradas` que utilizan el problema `raizCuadrada`, ¿Se puede eliminar el requiere: "*Todos los elementos de s son positivos*"? Justificar.

No. Porque necesitás *devolver* un número \mathbb{R} .

2. ¿Qué consecuencia tiene la diferencia de *asegura* en el resultado entre los problemas `raicesCuadradasUno` y `raicesCuadradasDos`?

Se ve en la respuesta dada al ejercicio. Se puede cambiar la posición de los resultados en la $seq(\mathbb{R})$

3. En base a la respuesta del ítem anterior, ¿Un algoritmo que satisface la especificación de `raicesCuadradasUno`, también satisface la especificación de `raicesCuadradasDos`? ¿Y al revés?

Sí, el conjunto solución de `raicesCuadradasUno`, es más pequeño, está contenido en la solución de `raicesCuadradasDos`, es decir:

Las condiciones de los *asegura* del `res(raicesCuadradasUno)` son **más fuertes** que las de `res(raicesCuadradasDos)`.

$$\begin{aligned} \{seq(res(raicesCuadradasUno))\} &\subseteq \{seq(res(raicesCuadradasDos))\} \\ res(raicesCuadradasUno) &\rightarrow res(raicesCuadradasDos) \\ \text{fuerte} &\rightarrow \text{débil} \end{aligned}$$

Las condiciones fuertes dan en general un conjunto de soluciones más chico. Las condiciones débiles son más relajadas, por lo que permiten que se meta mucha basura no deseada en el conjunto solución.

4. Explicar en palabras las diferencias entre los problemas `raicesCuadradasCinco` y `raicesCuadradasSeis`. ¿Cómo influye el *asegura* de longitud máxima? ¿Es $\langle \sqrt{3}, \sqrt{9} \rangle$ una salida válida para ambos problemas, dado $s = \langle 3, 9, 11, 15, 18 \rangle$? ¿Es $\langle \sqrt{3}, \sqrt{9}, \sqrt{11}, \sqrt{13} \rangle$ una salida válida para el problema `raicesCuadradasCinco` dado $s = \langle 3, 9, 11 \rangle$ ambos problemas,

Los *asegura*son más fuertes en `raicesCuadradasSeis`, al evitar poner elementos que nada tengan que ver con s limitando la longitud de res . $asegura_{fuerte} \rightarrow asegurado_{débil}$, por lo que la solución: $\langle \sqrt{3}, \sqrt{9} \rangle$ es solución de ambos, sin embargo $\langle \sqrt{3}, \sqrt{9}, \sqrt{11}, \sqrt{13} \rangle$ no es solución del problema con los *asegura*fuertes, porque no cumple que sea solución de $s = \langle 3, 9, 11 \rangle$ violando el *asegura*sobre la longitud $res \leq s$

5. ¿Cómo me cambia en el problema `raicesCuadradasCuatro` agregar un *asegura*que diga que res tiene misma longitud que s ?

Eso me obligaría a tener la respuesta de problema `raicesCuadradasDos`. Si bien no puedo agregar basura, puedo intercambiar los índices de res

6. ¿Si los problemas `raicesCuadradasDos` y `raicesCuadradasTres` tienen el mismo *res* para la misma entrada (una secuencia específica de números), quiere decir que son el mismo problema?

No, el aseguradel `raicesCuadradasDos` es más fuerte que el del `raicesCuadradasTres`.

7. ¿Qué ocurre si eliminamos los `requiere` "no hay repetidos"? ¿Es $\langle 2, 2, 1 \rangle$ una salida válida para el problema `raicesCuadradasDos` dado $s = \langle 4, 1, 1 \rangle$?

El problema quedaría así:

`requiere`: {Todos los elementos de s son positivos}

`asegura`: { res tiene la misma cantidad de elementos que s .}

`asegura`: {Los elementos de res son la salida de aplicar el problema `raizCuadrada()` a todos los elementos de la secuencia s }

Sí, es una respuesta válida, dado que $\langle 2, 2, 1 \rangle$ tiene igual longitud que $\langle 4, 1, 1 \rangle$ y los valores son las raíces cuadradas de los elementos de s .

¿Elementos de s con igual valor pero en índices distintos, son el mismo elemento?

¿Es como en álgebra? $\langle 1, 1, 2 \rangle \stackrel{?}{=} \langle 1, 2, 2 \rangle$

Ejercicio 3. Responder las preguntas dada la siguiente especificación para el problema de ordenar una secuencia de enteros (es decir, dada una secuencia de enteros, devolver los mismos elementos ordenados de menor a mayor):

```
problema ordenar (s: seq<Z>): seq<Z> {  
  requiere: {True}  
  asegura: {resultado es una secuencia de la cual cada elemento es  
            estrictamente mayor que el anterior}  
}
```

a) Dado $s = \langle 4, 3, 5 \rangle$ como secuencia de entrada ¿Es válida $res = \langle 3, 4, 5 \rangle$. \rightarrow sí.

b) **Hacer!**

c) **Hacer!**

d) **Hacer!**

e) **Hacer!**

f) **Hacer!**

Ejercicio 4. Se desea especificar el problema de duplicar todos los valores de la secuencia y se cuenta con la siguiente especificación:

```
problema duplicarTodos (s: seq<Z>): seq<Z> {  
  requiere: {True}  
  asegura: {resultado tiene la misma cantidad de elementos que s}  
}
```

a) ¿Qué problemas tiene la especificación dada? Dar ejemplos de valores para *resultado* que satisfagan la especificación pero no sean respuestas correctas.

Hacer!

b) Indicar cuál/es de los siguientes **asegura** debería/n ser agregado/s a la especificación. Justificar en cada caso por qué deberían o no ser agregados.

- **asegura**: {Para cada valor x que pertenece a s , hay algún valor en *resultado* que es la salida de `duplicar(x)`.}
- **asegura**: {En cada posición de *resultado*, el valor es mayor al valor en esa misma posición de s .}
- **asegura**: {En cada posición de *resultado*, el valor es igual a la salida de aplicar `duplicar()` al valor en esa misma posición de s .}
- **asegura**: {Todos los elementos de *resultado* son números pares.}

Hacer!

c)

a es válida b no cumple el $< \dots$ se corrige con \leq o achicar los requiere, por ejemplo, no tener elementos repetidos.

f) problema ordenar ($s: \text{seq}(\mathbb{Z}): \text{seq}(\mathbb{Z})$)

requiere: True asegura: resultado es una secuencia con todos los mismos elementos que s asegura: $\#(A,s) = \#(A.\text{resultado})$ asegura: resultado tiene igual long que s asegura: ordenados de menor a mayor

5) problema cantidadColectivosLinea($\text{linea}: \mathbb{Z}, \text{colectivos}: \text{seq}(\mathbb{Z}), \text{colectivosHoy}: \text{seq}(\mathbb{Z}) : \mathbb{Z}$)
requiere: $\text{linea} \in \text{colectivos}$ asegura: respuesta $\#(\text{linea}, \text{colectivos})$

1d 1e 2.2 2.3 1h 1i 2.4 2.7 3 a,b,c,d,e,f 5 a