

E1. TADs y especificación de problemas [35 pts]

Se quiere modelar una biblioteca en la cuál los libros se organizan en estantes. Cada estante puede alojar una cierta cantidad de libros. Al abrir la biblioteca, se indica la cantidad de estantes que tiene y cuántos libros puede alojar cada uno (cada estante puede alojar una cantidad diferente de libros). A partir de ese momento, no pueden agregarse ni eliminarse estantes.

Los libros se identifican por un número (llamado ISBN) que es único. Cuando ingresa un libro a la biblioteca, se indica en qué estante se va a guardar, y el libro se ubica en la primera posición libre de dicho estante. Cuando se retira un libro de la biblioteca, se indica el número de estante y la posición. En dicho estante, todos los libros a la derecha del que se retiró se corren a la izquierda para tapar la posición que queda libre.

Se desea conocer, para una determinada posición de un estante en particular, qué libro se encuentra guardado en esa posición.

Ejemplo:

Estado de la biblioteca (en un cierto momento):

estante 0	111	2864	6946				
estante 1							
estante 2	12423	23453	12	5467			

Luego de agregar el libro 975 al estante 0:

estante 0	111	2864	6946	975			
estante 1							
estante 2	12423	23453	12	5467			

Luego de sacar el libro de la posición 1 del estante 2:

estante 0	111	2864	6946	975			
estante 1							
estante 2	12423	12	5467				

- Indique las operaciones (procs) del TAD con todos sus parámetros.
- Describa el TAD en forma completa, indicando sus observadores, los requiere y asegura de las operaciones. Puede agregar los predicados y funciones auxiliares que necesite, con su correspondiente definición

E2. Precondición más débil [20 pts]

Dado el siguiente condicional determinar la precondición más débil que permite hacer valer la poscondición (Q) propuesta.

```

if (a mod 3 = 0)
  a := abs(a) + 1
else
  a := abs(a) * 3
endif
  
```

$$Q \equiv (\exists j : \mathbb{Z})(j \geq 0 \wedge j \times 3 = a)$$

- Escriba la WP esperada y justifique brevemente en castellano
- Derivarla formalmente a partir de los axiomas de precondición más débil. Para obtener el puntaje máximo deberá simplificarla lo más posible.

E3. Correctitud del ciclo [35 pts]

Dado el siguiente programa con su especificación

$$P_c \equiv \{n > 0 \wedge res = 0 \wedge i = 0 \wedge j = 1\}$$

```
while (i < n)
  res := res + j
  j := j * 2;
  i := i + 1;
endwhile
```

$$Q_c \equiv \{res = \sum_{k=0}^{n-1} 2^k\}$$

Contamos con el siguiente invariante, que sabemos que es incorrecto:

$$I \equiv \{0 \leq i \leq n \wedge j = 2^{2^i} \wedge res = \sum_{k=1}^i 2^k\}$$

- Indique para cada axioma del teorema del invariante si se cumple o no. Si es verdadero demuéstrelo formalmente. Si es falso muestre un contraejemplo.
- Escriba un invariante que resulte correcto, indicando las partes que cambió. No es necesario demostrar formalmente su correctitud.
- Decida si las siguientes funciones variantes son válidas o no y explique por qué.
 - $f_v = n$
 - $f_v = n - i$
 - $f_v = n - 1 - i$

E4. Conceptuales [10 pts]

Dada la siguiente especificación y un programa que la satisface, indicar si las siguientes afirmaciones son verdaderas o falsas. En caso de ser verdaderas justifique brevemente su respuesta. En caso de ser falsa, dé un contraejemplo.

- El programa seguirá siendo correcto si cambio el requiere por uno más fuerte
- El programa seguirá siendo correcto si cambio el requiere por uno más débil

```
proc todosPares (in s: seq(Z)) : Bool {
  requiere {|s| mod 2 = 0}
  asegura {res = true  $\leftrightarrow$  ( $\forall i : \mathbb{Z}$ )( $0 \leq i < |s| \rightarrow_L s[i] \bmod 2 = 0$ )}
```

```
}
res = true
i = 0
while (i < s.length())
  if (s[i] mod 2 != 0)
    res := false
  endif
  if (s[i+1] mod 2 != 0)
    res := false
  endif
```

```
  i := i+2
endwhile
```