4. (TADs) Considere la siguiente especificación del tipo Conjunto de elementos de algún tipo T.

spec Set **of** T **where**

constructors

fun empty_set() **ret** s : Set **of** T
{- *crea un conjunto vacío.* -}

proc add (**in** e : T, **in/out** s : Set **of** T)
{- *agrega el elemento* e *al conjunto* s. -}

destroy

proc destroy (**in/out** s : Set **of** T)
{- *Libera memoria en caso que sea necesario.* -}

operations

fun is_empty_set(s : Set **of** T) **ret** b : bool
{- *Devuelve* True *si* s *es vacío.* -}

fun cardinal(s : Set **of** T) **ret** n : nat
{- *Devuelve la cantidad de elementos que tiene* s -}

fun member(e : T, s : Set **of** T) **ret** b : bool
{- *Devuelve True si el elemento* e *pertenece al conjunto* s -}

proc inters (**in/out** s : Set **of** T,**in** s0 : Set **of** T)
{- *Elimina de* s *todos los elementos que NO pertenecen a* s0 -}

{- **PRE:** not is_empty_set(s) -}
fun get(s : Set **of** T) **ret** e : T
{- *Devuelve algún elemento (cualquiera) de* s -}

proc elim (**in/out** s : Set **of** T,**in** e : T)
{- *Elimina el elemento* e *del conjunto* s *en caso que esté.* -}

proc union (**in/out** s : Set **of** T,**in** s0 : Set **of** T)
{- *Agrega a* s *todos los elementos de* s0 -}

proc diff (**in/out** s : Set **of** T,**in** s0 : Set **of** T)
{- *Elimina de* s *todos los elementos de* s0 -}

fun copy_set(s1 : Set **of** T) **ret** s2 : Set **of** T
{- *Crea un nuevo conjunto* s2 *con todos los elementos de* s1 -}

(a) Implementá los constructores del TAD Conjunto de elementos de tipo T, y las operaciones member, elim e inters, utilizando la siguiente representación:

implement Set **of** T **where**

type Set **of** T = **tuple**
elems : array[0..N-1] **of** T
size : nat
end tuple

¿Existe alguna limitación con esta representación de conjuntos? En caso afirmativo indicá si algunas de las operaciones o constructores tendrán alguna precondición adicional.

NOTA: Si necesitás alguna operación extra para implementar lo que se pide, debes implementarla también.

(b) Utilizando el tipo **abstracto** Conjunto de elementos de tipo T, implementá una función que reciba un conjunto de enteros s, un número entero i, y obtenga el entero perteneciente a s que está *más cerca* de i, es decir, un $j \in s$ tal que para todo $k \in s$, $|j - i| <= |k - i|$. Por ejemplo si el conjunto es $1, 5, 9$, y el entero 7, el resultado puede ser 5 o 9.

Voraz

...mentación, debe utilizarse el lenguaje de la materia. ...ha sido comprendida. Las explicaciones deben ser completas.

...ejercicios en **hojas separadas**.

1. Un colectivero conduce su pequeño colectivo. Muy pequeño. Solamente hay lugar para un pasajero. Más que un colectivo, parece una moto. Su recorrido o viaje va de la parada 1 hasta la parada n pasando por las paradas intermedias $2, 3, \ldots, n-1$. Hay m pasajeros esperando. Para cada pasajero i sabemos en qué parada se quiere subir (s_i), y en qué parada se va a bajar (b_i con $1 \leq s_i < b_i \leq n$). La intención del colectivero es trasladar en un viaje a la mayor cantidad de pasajeros posible. El colectivero no tiene obligación de levantar un pasajero por más que esté libre, puede preferir reservarlo para un pasajero que sube después.

   Se debe obtener el número máximo de pasajeros trasladables en un único viaje.

   Se pide lo siguiente:

   (a) Indicar de manera simple y concreta, cuál es el criterio de selección voraz para construir la solución?

   (b) Indicar qué estructuras de datos utilizarás para resolver el problema.

   (c) Explicar en palabras cómo resolverá el problema el algoritmo.

   (d) Implementar el algoritmo en el lenguaje de la materia de manera precisa.

2. El presidente de tu país te acaba de elegir como ...

Backtracking

...brevientes (con $m \leq n$).

8. Usted vive en la montaña, es invierno, y hace mucho frío. Son las 10 de la noche. Tiene una voraz estufa a leña y n troncos de distintas clases de madera. Todos los troncos son del mismo tamaño y en la estufa entra solo uno por vez. Cada tronco i es capaz de irradiar una temperatura k_i mientras se quema, y dura una cantidad t_i de minutos encendido dentro de la estufa. Se requiere encontrar el orden en que se utilizarán la menor cantidad posible de troncos a quemar entre las 22 y las 12 hs del día siguiente, asegurando que entre las 22 y las 6 la estufa irradie constantemente una temperatura no menor a $K1$; y entre las 6 y las 12 am, una temperatura no menor a $K2$.

3. Para cada uno de los siguientes algoritmos determinar **por separado** cada uno de los siguientes incisos.

   (a) ¿Qué hace? ¿Cuáles son las precondiciones necesarias para haga eso?

   (b) ¿Cómo lo hace?

   (c) El orden del algoritmo, analizando los distintos casos posibles.

   ~~(d) Proponer nombres más adecuados para los identificadores (de variables, funciones y procedimientos).~~

```
proc q(in/out a : array[1..N] of int, in x : nat)
    for j:= 1 to x do
        m:= j
        for k:= j+1 to x do
            if a[k] < a[m] then m:= k fi
        od
        swap(a,j,m)
    od
end proc
proc p(in/out a : array[1..N] of int, in i : nat)
    q(a, i-1)
    r(a, i+1)
end proc
```

```
proc r(in/out a : array[1..N] of int, in y : nat)
    for j:= y to n do
        m:= j
        while m > y ∧ a[m] < a[m-1] do
            swap(a,m,m-1)
            m:= m-1
        od
    od
end proc
```

4. Dada la especificación del tad Cola: ...

5. (Para alumnos libres) Sea T un árbol (no necesariamente binario) y supongamos que deseamos encontrar la hoja que se encuentra más cerca de la raíz. ¿Cuáles son las distintas maneras de recorrer T? ¿Cuál de ellas elegirías para encontrar esa hoja y por qué?