

## Programación II - TP1 2do Cuatrimestre 2019

**Fecha de presentación: jueves 3/10/19**

**Fecha de entrega por mail: jueves 17/10/19**

### Requerimientos técnicos:

- Grupos de 1 ó 2 personas.
- Se deben utilizar las herramientas de Tecnologías Java que se vieron en la materia. Al menos una vez deben usarse:
  - *StringBuilder*, cuyo uso debe basarse en la necesidad de modificar el string.
  - *Iteradores*, para recorrer las colecciones de Java y para alguna de las clases de usuario que implemente.
- Además de pasar el *junit* suministrado junto con el TP, la cátedra testeará los ejercicios con otro *junit*1 adicional, por lo que se recomienda armar un *junit* propio para probar, antes de entregar el TP.
- Se debe escribir el *IREP* de la implementación en el informe, el cual no debería extenderse más de una página.

### Presentación del trabajo

La biblioteca de la UNGS necesita hacer una reestructuración de sus libros para lo cual comprarán nuevos estantes para contener los libros. Los responsables de la institución solicitaron que los estudiantes de Computación les ayuden creando una aplicación de computadora que les permita organizar y administrar los libros en las nuevas estanterías. Luego de hacer un estudio de esta solicitud, los profesores de la carrera decidieron que los estudiantes de Programación II son los indicados para desarrollar una versión inicial del trabajo que se llamará "*Biblioteca Digital UNGS*" (de manera abreviada: *BDUNGS*) y se les encarga hacer el diseño e implementación.

La biblioteca se construye con una cantidad predeterminada de estantes de iguales dimensiones, en donde se ubicarán los libros. Los libros se clasifican según su categoría y estarán ubicados en estantes destinados a dicha categoría.

### A continuación, se especifican los detalles para su diseño.

- La biblioteca se crea con una cantidad predeterminada de estantes disponibles con un ancho fijo para cada uno de ellos. Este ancho se mide en cm.
- Los estantes están rotulados con su *categoría*. Cada estante tiene un ancho en cm.
- Los libros se registran por su *código*, *ISBN*, su *nombre* y *categoría*. Se conoce también el espesor, o *ancho*, de cada uno, a fines de poder ubicarlos en los estantes. Se puede tener más de un ejemplar del mismo libro.
- Los estantes se identifican dentro de la biblioteca con un número de orden.

### **Los requerimientos que fueron solicitados por la biblioteca son los siguientes:**

- a. **Ingresar un nuevo libro.** Ingresar un libro a la biblioteca requiere ubicarlo en un estante con su misma categoría, donde se encuentre lugar disponible.
- b. **Rotular un estante.** Se asignan rótulos a los estantes con su categoría (sería su rótulo). Se van agregando nuevos estantes rotulados (vacíos) a medida que se necesiten. Si el estante ya tenía rótulo y está vacío se puede re-rotular.
- c. **Eliminar un libro.** Se necesitará eliminar libros de la biblioteca de manera definitiva.
- d. **Reacomodar los libros de una categoría.** Se necesitará poder reacomodar los libros ubicados en los estantes de una categoría. Esto se realiza traspasando los libros de los estantes con mayor espacio libre a los estantes con menor espacio libre. El proceso debe informar cuántos estantes se liberaron. Si un libro fue prestado se reserva el lugar que ocupa como si estuviera presente. Ver Anexo.
- e. **Ver los libros de una categoría.** Se requiere conocer todos los libros de una categoría determinada, incluyendo la cantidad de ejemplares que se encuentra de cada libro.
- f. **Conocer espacio libre de un estante** identificado por su número de ubicación.

### **Modelar el TAD BDUNGS**

Escribir el Irep.

Justificar los TAD o clases soporte. Es obligatorio que al menos haya otro TAD que no sea una clase predefinida de Java, y con las funcionalidades suficiente para que no sea una simple “cáscara” que tan solo enmascara una estructura\*.

### **Implementar el TAD BDUNGS**

Debe ser consistente con el Irep elegido.

Debe respetar la interfaz sugerida por la cátedra.

Debe cumplir satisfactoriamente el junit de la cátedra.

### **Presentación del Informe (no más de dos páginas!):**

- Escribir el IREP
- Justificar adecuadamente la relación entre los requerimientos u operaciones pedidas y las estructuras de datos seleccionadas.

### **Consideraciones importantes para la implementación:**

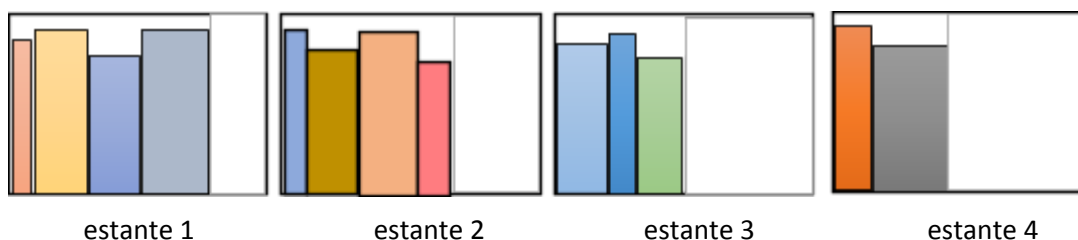
- Se deben utilizar las colecciones de datos adecuadas para que se pueda acceder a los estantes de una categoría de la manera más rápida o eficiente posible.
- Si es necesario implementarán con redundancia de datos para cumplir con los requerimientos.

- Devolver el espacio libre de un estante, una vez conocida su ubicación dentro de la biblioteca, **debe ser  $O(1)$** .
- El rotular un estante crea uno nuevo si no existe, pero si ya existe y está vacío, le cambia el rótulo y si no está vacío no hace nada. (lanza una excepción).

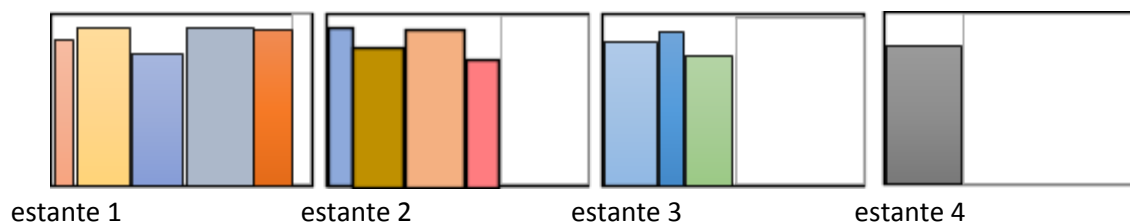
## ANEXO

### Se pide reacomodar los estantes de categoría X

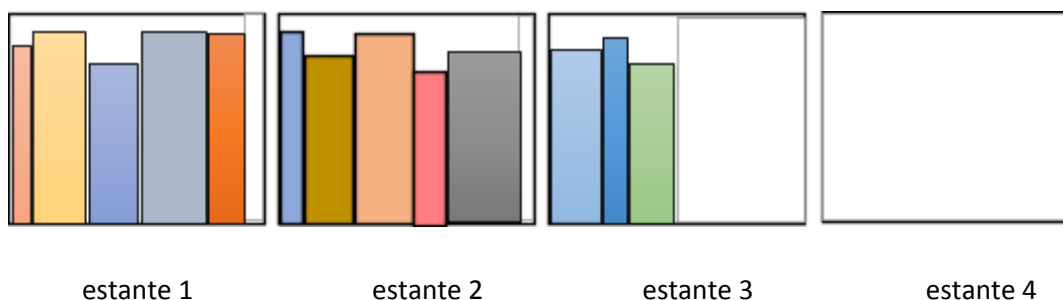
Se considera como ejemplo los 4 estantes que corresponden a la categoría X y se pasan los libros, uno por uno, del estante más vacío hacia el menos vacío.



Se pasan los libros del estante 4 al 1, comenzando con uno de ellos, por ejemplo el naranja.



El libro naranja pasa al estante 1. Fue posible porque su espesor es menor que su espacio libre.



El libro gris del estante 4 se reubica. No tiene espacio en el estante 1. Se continúa con el estante siguiente, y encuentra lugar en el estante 2. El estante 4 quedó vacío.

Se continuaría con el estante 3 pero ninguno de sus libros pudo ser reacomodado. Se considera que ya se hizo lo máximo posible y termina el proceso de reacomodar y devuelve 1, ya que un estante quedó vacío.

Se pide que el siguiente caso de uso (código cliente) funcione con el TAD BDUNGS

```
public static void main(String[] args) {  
    BDUNGS bd = new BDUNGS(10, 100);  
    bd.rotularEstante("Computacion", 1);  
    bd.rotularEstante("Matematica", 2);  
    bd.ingresarLibro("9789684443457", "Computacion", "Estructuras de datos", 5);  
    bd.ingresarLibro("9788415552222", "Computacion", "Estructuras de datos en Java", 7);  
    bd.ingresarLibro("9389557783457", "Matematica", "Análisis de Funciones", 4);  
    System.out.println(bd);  
    bd.eliminarLibro("9389557783457");  
    bd.rotularEstante("Análisis Matemático", 2);  
    System.out.println(bd);  
}
```