

## **Contexto problemático**

La cultura del libro ha renacido en la ciudad de Cali, es por eso que una Librería muy importante e innovadora se dispone a abrir sus puertas en la sultana del valle. Esta nueva compañía dedicada a la venta de libros de múltiples géneros y diversas lenguas se caracteriza por su estilo muy particular, innovador y eficiente a la hora de atender a sus usuarios, requiere un programa que le permita al usuario tener una grata experiencia a la hora de elegir los libros y pagarlos.

## **Paso 1: Identificación del problema**

### **Identificación de necesidades y síntomas:**

- Se necesita un programa que permita modelar la estructura de una librería.
- La solución del problema debe ser ideada usando estructuras de datos que permitan relacionarse con cada una de las necesidades del programa.
- Se debe tener en cuenta estructuras de datos para para modelar las estanterías de libros, las canastas de libros y la fila a la hora de pagar.

### **Definición del problema**

La librería requiere un programa que le permita simular cómo es una atención a un cliente, desde cómo elegir los libros, donde empacarlos y cómo pagarlos.

## **Paso 2: Recopilación de información**

Con el objetivo de modelar de la mejor forma el problema, se ha hecho una profunda búsqueda de información sobre las estructuras de datos que apliquen a este problema

## **Definiciones**

<https://wikipedia.org>

<http://lineadecodigo.com/java/usar-una-hashtable-java/>

## **Estructura de Datos**

En ciencias de la computación, una estructura de datos es una forma particular de organizar datos en una computadora para que puedan ser utilizados de manera eficiente. Diferentes tipos de estructuras de datos son adecuados para diferentes tipos de aplicaciones, y algunos son altamente especializados para tareas específicas.

### **Stack(Pila)**

Una pila (stack en inglés) es una lista ordenada o estructura de datos que permite almacenar y recuperar datos, el modo de acceso a sus elementos es de tipo LIFO (del inglés Last In, First Out, «último en entrar, primero en salir»).

### **Queue(Cola)**

Una cola es una estructura de datos, caracterizada por ser una secuencia de elementos en la que la operación de inserción push se realiza por un extremo y la operación de extracción pop por el otro. También se le llama estructura FIFO, debido a que el primer elemento en entrar será también el primero en salir.

## HashTable

Una Hashtable Java es una estructura de datos que utiliza una función hash para identificar datos mediante una llave o clave (ej. Nombre de una persona). La función hash transforma una llave a un valor índice de un arreglo de elementos.

### Paso 3: Búsqueda de soluciones creativas

Las siguientes son las alternativas que se tienen para solucionar el problema

#### Alternativa 1: Usar pilas, colas y hashtables propias

- El implementar nuestras propias colas, pilas y hashtables nos facilita de gran manera nuestra tarea ya que podemos acondicionar estas estructuras de acuerdo a las necesidades que tenemos.

#### Alternativa 2: Usar arreglos y estructuras no dinámicas

- El uso de arreglos y estructuras no conocidas, tienen un grado de implementación y entendimiento muy menor, ya que se tiene un tiempo respetable trabajando con estas estructuras.

#### Alternativa 3: Usar pilas, colas y hashtables de Java

- Java trae una librería donde se pueden encontrar estas estructuras de datos y hacer uso de estos de una manera muy eficientes.

### Paso 4: Transición de las Ideas a los Diseños Preliminares

Se descarta la alternativa número 2, ya que el trabajar solamente con arreglos y estructuras no dinámicas nos dificulta la realización del trabajo y no se podría llevar de la mejor manera las necesidades de añadir libros a la canasta, buscar en una estantería e ingresar a una cola de pago.

La revisión cuidadosa de las otras alternativas nos conduce a lo siguiente:

#### Alternativa 1: Usar pilas, colas y hashtables propias

- Estas estructuras de datos propias contarán con métodos que nos ayudaran con facilidad a cumplir con lo requerimientos pedidos para la aplicación, métodos como: poll, peek, offer, etc

#### Alternativa 2: Usar arreglos y estructuras no dinámicas

- Estas estructuras de datos de java cuentan con los mismo métodos anteriormente mencionados, son de gran ayuda porque estos métodos ya están implementados y son de gran ayuda.

### **Paso 5: Evaluación y Selección de la Mejor Solución**

- Criterio A. Precisión de la solución. La alternativa entrega una solución:
  - [2] Exacta (se prefiere una solución exacta)
  - [1] Aproximada
- Criterio B. Eficiencia. Se prefiere una solución con mejor eficiencia que las otras consideradas. La eficiencia puede ser:
  - [4] Constante
  - [3] Mayor a constante
  - [2] Logarítmica
  - [1] Lineal
- Criterio C. Completitud. Se prefiere una solución que encuentre todas las soluciones. Cuántas soluciones entrega:
  - [3] Todas
  - [2] Más de una si las hay, aunque no todas
  - [1] Sólo una o ninguna
- Criterio D. Facilidad en implementación algorítmica:
  - [2] Compatible con las operaciones aritméticas básicas de un equipo de cómputo moderno
  - [1] No compatible completamente con las operaciones aritméticas básicas de un equipo de cómputo moderno

	<b>Criterio A</b>	<b>Criterio B</b>	<b>Criterio C</b>	<b>Criterio D</b>	<b>Total</b>
<b>Alternative(1)</b>	<b>2</b>	<b>2</b>	<b>3</b>	<b>2</b>	<b>9</b>
<b>Alternative(2)</b>	<b>2</b>	<b>2</b>	<b>3</b>	<b>2</b>	<b>9</b>

La evaluación de las dos alternativas son totalmente iguales, por lo requerido se hará uso de la alternativo 1 en la que implementamos nuestras propias estructuras de datos que se adaptan de buena manera a nuestras necesidades.