

Caso de estudio #1

Objetivo

Aplicar los conocimientos adquiridos durante el curso en la realización de una aplicación real, analizando diferentes alternativas.

Contexto

La empresa “Grandeza y Elegancia ANte Todo” (por sus siglas *G.E.AN.T*), líder en el rubro de supermercados del país, necesita gestionar los productos de su cadena de supermercados, y nos ha encargado la construcción de un sistema software que permita hacerlo en forma eficiente.

La información que se tiene sobre un producto puede ser muy variable y extensa, pero como mínimo se tendrá:

- Nombre del producto,
- Código de identificación del producto
- Precio unitario.
- Cantidad existente del mismo en las góndolas y almacenes

y muy probablemente tengamos grandes volúmenes de datos asociados al producto (fotos, código de barras, etc.)

Nuestro sistema deberá implementar inicialmente las siguientes funcionalidades básicas:

1. Incorporar un nuevo producto a la cadena de supermercados.
2. Agregar stock a un producto existente.
3. Simular la venta de un producto (reducir el stock de un producto existente
4. Eliminar productos que ya no se venden (por no ser comercializados más).
5. Dado un código de producto, indicar las existencias del mismo en el almacén.
6. Listar todos los productos registrados, ordenados por nombre, presentando además su stock.

Como se indicó inicialmente, la empresa es una cadena de supermercados, que cuenta con una importante red de locales en diferentes ciudades del país, y en cada ciudad puede haber sucursales en diferentes barrios. Incluso, en un mismo barrio puede haber más de una sucursal.

La información de una sucursal es muy extensa, pero como mínimo se tendrá:

- Nombre de la sucursal
- Teléfono
- Dirección
- Barrio
- Ciudad

Las funcionalidades básicas a implementar deberán comportarse de la siguiente forma:

1. Incorporar un nuevo producto a una sucursal del supermercado.
2. Agregar stock a un producto existente en una sucursal.

3. Simular la venta de un producto en una sucursal (reducir el stock de un producto existente). De no haber stock suficiente para la venta en esa sucursal, deberá indicarse la lista de sucursales que tengan el stock suficiente, ordenada por cantidad de producto.
4. Eliminar productos que ya no se venden (por no ser comercializados más) en todas las sucursales del supermercado.
5. Dado un código de producto, indicar las existencias del mismo en todas las sucursales, ordenada por sucursal.
6. Listar todos los productos registrados, en una sucursal, ordenado por nombre de producto, presentando además su stock.
7. Listar todos los productos registrados, ordenados por ciudad, barrio, y nombre de producto, presentando además su stock.

Consigna

1. Diseña la combinación de estructuras de datos (utilizando árboles y listas) necesaria para representar la información. Se recomienda la realización de diagramas explicativos que ilustren las diferentes alternativas identificadas.
2. Evalúa al menos dos alternativas de combinaciones de estructuras, teniendo en cuenta el contexto del problema.
3. Desarrolla en pseudocódigo los métodos necesarios para implementar las funcionalidades necesarias de la forma más eficiente posible.
4. Analiza el orden del tiempo de ejecución de los métodos desarrollados, mostrando que es la forma más eficiente posible de acuerdo a las diferentes combinaciones de estructuras que pudieran diseñarse.

Entrega

- Informe completo de acuerdo a estándar publicado por la cátedra, con análisis de alternativas, justificación de la elección y notas de diseño (incluyendo pseudocódigo de alto nivel de las operaciones requeridas, y análisis del orden del tiempo de ejecución de las mismas)
- Proyecto NETBEANS completo con el software desarrollado, incluyendo los casos de prueba correspondientes y las pruebas de ejecución sobre operaciones de ejemplo
- Demostración pública (ante pares) del diseño y software desarrollados.