

**ĐẠI HỌC ĐÀ NẴNG**  
**ĐẠI HỌC BÁCH KHOA**  
**KHOA CÔNG NGHỆ THÔNG TIN**



**ĐỒ ÁN CƠ SỞ LẬP TRÌNH**  
**ĐỀ TÀI 703. QUẢN LÝ XUẤT, NHẬP KHO**

Giảng viên hướng dẫn: **GVC. ThS. Trần Hồ Thủy Tiên**

Sinh viên thực hiện:

**Hồ Quốc Thiên Anh**

**LỚP 21TCLC\_KHDL, NHÓM 21N15A**

**Phan Minh Nhật**

**LỚP 21TCLC\_KHDL, NHÓM 21N15A**

**Nguyễn Văn Thanh Hoàng**

**LỚP 21TCLC\_KHDL, NHÓM 21N15A**

**Đà Nẵng, ngày 26 tháng 12 năm 2022**

## MỞ ĐẦU

Trong thời đại khoa học kỹ thuật tiến bộ, chất lượng cuộc sống cũng như năng suất lao động của con người càng ngày càng được nâng cao, khiến cho lượng sản phẩm cũng như hàng hóa nhân loại tạo ra gia tăng với một tốc độ chóng mặt. Từ đó, các chủ doanh nghiệp phải đối mặt với một bài toán cực kỳ hóc búa về khả năng quản lý được lượng vật tư mà doanh nghiệp của mình đã tạo ra. Nếu như không có khả năng nắm bắt được tình hình sản phẩm trong kho của mình, các doanh nghiệp sẽ phải đối diện với những thách thức cực lớn đến từ thị trường cũng như từ các đối thủ cạnh tranh khác.

Đi từ thực tế quan trọng của việc tìm ra lời giải cho vấn đề trên, nhóm chúng em đã dựa vào các kiến thức đã được học trong môn Lập trình hướng đối tượng và Cấu trúc dữ liệu để xây dựng chương trình quản lý xuất nhập kho được cài đặt bằng ngôn ngữ C++ ứng dụng kiến thức lập trình OOP cũng như dựa trên kiến thức về cấu trúc dữ liệu danh sách liên kết đã được học. Chương trình có thể phục vụ các nhu cầu cơ bản của người dùng như: tạo danh sách hàng hóa nhập/xuất kho, hiển thị danh sách, tìm kiếm vật tư,...

Nhóm chúng em mong rằng đây sẽ là cơ sở để phát triển và xây dựng các chương trình phức tạp và hoàn thiện hơn nhằm phục vụ cho nhu cầu quản lý xuất nhập kho vật tư của các chủ doanh nghiệp, xí nghiệp.

Đặc biệt, nhóm chúng em xin gửi lời cảm ơn đến cô Trần Hồ Thủy Tiên đã giúp đỡ, hướng dẫn tận tình và chỉ điểm ra những sai sót của chúng em trong suốt quá trình làm đồ án để nhóm có thể hoàn thành đề tài này.

Nhóm chúng em xin chân thành cảm ơn!

## MỤC LỤC

MỞ ĐẦU .....	1
1. TỔNG QUAN ĐỀ TÀI .....	4
1.1. Đặt vấn đề.....	4
1.2. Giới thiệu đề tài .....	4
2. PHÂN TÍCH CHỨC NĂNG HỆ THỐNG.....	5
2.1. Quản lý hàng hóa.....	5
2.2. Quản lý phiếu nhập, xuất.....	6
3. THIẾT KẾ CẤU TRÚC DỮ LIỆU VÀ THUẬT TOÁN .....	7
3.1. Phát biểu bài toán .....	7
3.2. Phân tích và ứng dụng cấu trúc dữ liệu và thuật toán trong hệ thống .....	7
3.2.1. Cấu trúc dữ liệu danh sách liên kết đôi .....	7
3.2.2. Thuật toán .....	8
3.2.3. Ứng dụng cấu trúc dữ liệu và thuật toán trong hệ thống .....	12
4. PHÂN TÍCH THIẾT KẾ HƯỚNG ĐỐI TƯỢNG .....	15
4.1. Cấu trúc hệ thống hướng đối tượng.....	15
4.1.1. Cấu trúc của các lớp hướng đối tượng .....	15
4.1.2. Quan hệ của các lớp đối tượng .....	20
4.2. Kết quả.....	20
4.2.1. Giao diện chính của chương trình .....	20
4.2.2. Kết quả thực thi chương trình.....	21
4.2.3. Nhận xét.....	29
5. KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN .....	30
5.1. Kết luận.....	30
5.2. Hướng phát triển .....	30
TÀI LIỆU THAM KHẢO .....	31

## **DANH MỤC HÌNH ẢNH**

Hình 2-1: Sơ đồ chức năng tổng quát của hệ thống .....	5
Hình 3-1: Minh họa cấu trúc dữ liệu danh sách liên kết đôi .....	7
Hình 4-1: Giao diện chính của chương trình.....	20
Hình 4-2: File dữ liệu ban đầu.....	21
Hình 4-3: Menu quản lý hàng hóa .....	21
Hình 4-4: Giao diện chức năng sắp xếp danh sách hàng hóa.....	22
Hình 4-5: Giao diện chức năng hiển thị danh sách hàng hóa.....	22
Hình 4-6: Giao diện hiển thị chức năng hiển thị lịch sử xuất nhập (1) .....	23
Hình 4-7: Giao diện chức năng hiển thị lịch sử xuất nhập (2) .....	23
Hình 4-8: Giao diện chức năng tìm kiếm hàng hóa theo 1 thuộc tính .....	24
Hình 4-9: Giao diện chức năng tìm kiếm hàng hóa theo nhiều thuộc tính.....	24
Hình 4-10: Menu quản lý phiếu.....	25
Hình 4-11: Giao diện chức năng tạo phiếu nhập.....	25
Hình 4-12: Kho sau khi tạo phiếu nhập.....	26
Hình 4-13: Giao diện chức năng tạo phiếu xuất.....	26
Hình 4-14: Kho sau khi tạo phiếu xuất.....	27
Hình 4-15: Giao diện chức năng sắp xếp danh sách phiếu.....	27
Hình 4-16: Giao diện chức năng hiển thị danh sách phiếu .....	28
Hình 4-17: Giao diện chức năng tìm kiếm phiếu theo 1 thuộc tính .....	28
Hình 4-18: Giao diện chức năng tìm kiếm phiếu theo nhiều thuộc tính .....	29

## **1. TỔNG QUAN ĐỀ TÀI**

### **1.1. Đặt vấn đề**

Hàng hóa là một phần quan trọng làm nên thành công của mỗi doanh nghiệp. Việc quản lý kho hàng hóa hiệu quả càng đóng góp trực tiếp hơn nữa vào sự thành công này. Điều này giúp cho quy trình vận hành và sản xuất của các doanh nghiệp diễn ra một cách suôn sẻ và liên tục. Để đạt được điều này yêu cầu cần có một biện pháp để quản lý được tình trạng xuất, nhập kho hàng một cách hợp lý và khoa học. Hiện nay có một số nơi vẫn áp dụng phương pháp thủ công, khiến cho việc tìm kiếm, thống kê danh mục hàng hóa trong kho gặp phải nhiều khó khăn.

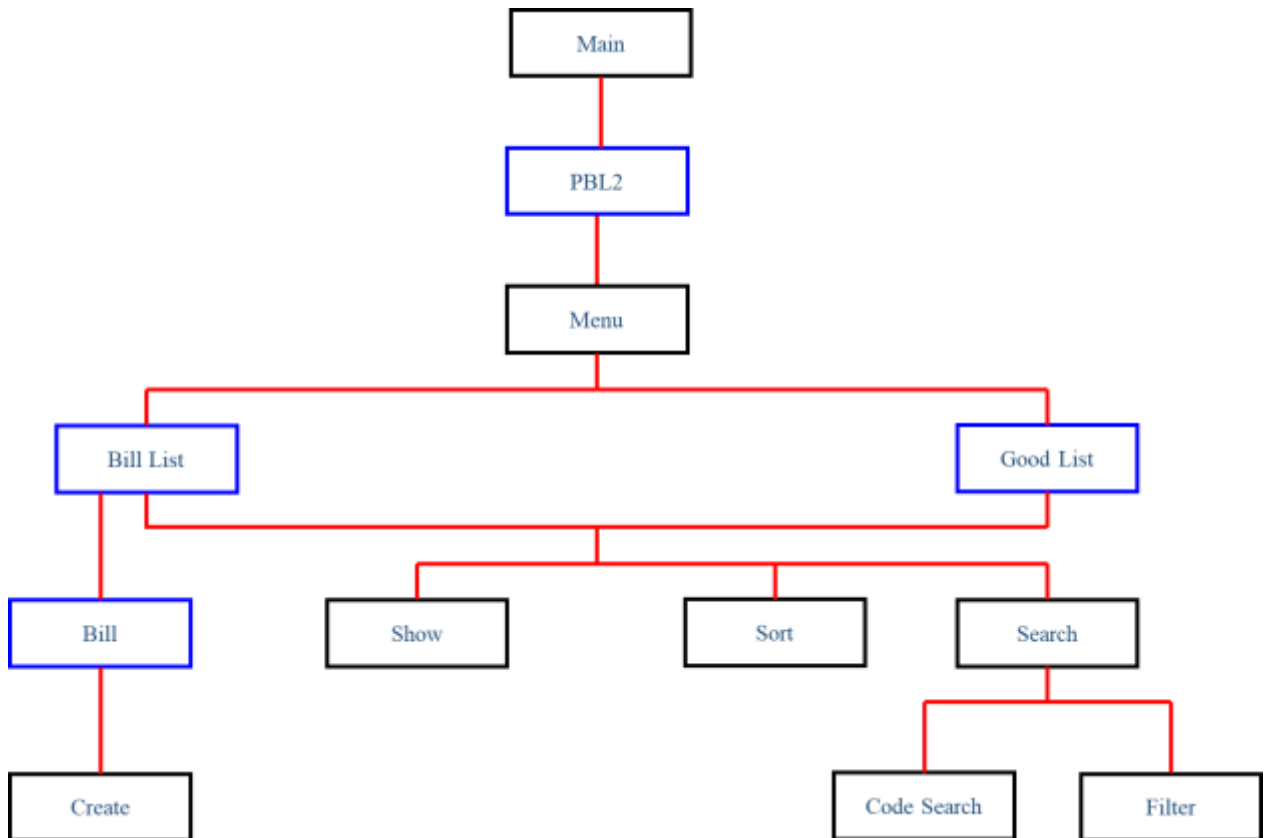
### **1.2. Giới thiệu đề tài**

Tin học hóa việc quản lý xuất nhập kho mang lại nhiều lợi ích so với quản lý thủ công. Quản lý trạng thái của kho là một bài toán quan trọng có vai trò quan trọng xuyên suốt quy trình hoạt động của doanh nghiệp, ảnh hưởng đến các quyết định sản xuất và buôn bán sản phẩm. Nhằm tối ưu hóa công việc này, nhóm chúng em xây dựng một hệ thống quản lý tình trạng xuất nhập kho đơn giản bằng ngôn ngữ C++ với các thao tác cơ bản như: làm phiếu nhập/xuất kho; tìm, thêm, xóa và tạo thông tin theo ý người dùng.

## 2. PHÂN TÍCH CHỨC NĂNG HỆ THỐNG

Hệ thống được viết thành một chương trình đơn giản với 2 tính năng tổng quát như sau:

- Chức năng 1: quản lý hàng hóa
- Chức năng 2: quản lý phiếu nhập xuất



*Hình 2-1: Sơ đồ chức năng tổng quát của hệ thống*

### 2.1. Quản lý hàng hóa

Các chức năng chính trong quản lý hàng hóa gồm:

1. Nhập, xuất vật tư: nhập/xuất hàng hóa vào/ra kho tùy theo yêu cầu của người dùng chương trình. Nếu như xuất ra hàng hóa không tồn tại trong kho sẽ báo lỗi. Hành động này sẽ tự động cập nhật cơ sở dữ liệu nếu thành công, ngược lại sẽ báo lỗi.
2. Sắp xếp danh sách vật tư: Sắp xếp lại danh sách hàng hóa trong kho dựa trên thuộc tính tùy chọn bởi người sử dụng.
3. Hiện thị danh sách vật tư: Hiện thị danh sách vật tư hiện có trong kho

4. Tìm kiếm danh sách vật tư: Tìm kiếm các vật tư dựa trên mã vật tư hoặc tập các thuộc tính bất kỳ tùy theo lựa chọn của người dùng.

5. Lịch sử nhập, xuất: Hiển thị lịch sử nhập và xuất kho trong khoảng thời gian được nhập vào bởi người dùng.

## **2.2. Quản lý phiếu nhập, xuất**

Các chức năng chính trong quản lý danh sách phiếu gồm:

1. Tạo phiếu nhập/xuất: tạo phiếu nhập/xuất kho với danh mục hàng hóa được nhập từ bàn phím. Hành động này sẽ tương đương với việc nhập/xuất hàng hóa vào/ra kho.

2. Sắp xếp danh sách phiếu: sắp xếp danh sách phiếu nhập xuất dựa trên thuộc tính được chọn bởi người dùng.

3. Hiển thị danh sách phiếu: hiển thị toàn bộ các phiếu nhập xuất đã được tạo.

4. Tìm kiếm phiếu: tìm kiếm phiếu nhập xuất trong danh sách phiếu tương ứng theo thuộc tính được chỉ định bởi người dùng.

### 3. THIẾT KẾ CẤU TRÚC DỮ LIỆU VÀ THUẬT TOÁN

#### 3.1. Phát biểu bài toán

Một kho hàng có sẵn thông tin về các loại hàng hóa hiện có. Xây dựng chương trình quản lý các lần xuất, nhập kho hàng.

Cấu trúc hàng hóa (vật tư) gồm: Mã hàng hóa, tên hàng hóa, loại hàng hóa, đơn vị tính, nhà sản xuất, số lượng, đơn giá, thành tiền.

Cấu trúc phiếu nhập xuất: mã phiếu, tên người tạo, ngày tạo, ngày nhập/xuất hàng, số mặt hàng, tổng giá tiền.

**a. Input:** Nhập thủ công hoặc đọc từ file thông tin của các hàng hóa. Nhập thông tin về phiếu nhập xuất. Người dùng lựa chọn các chức năng muốn thực hiện.

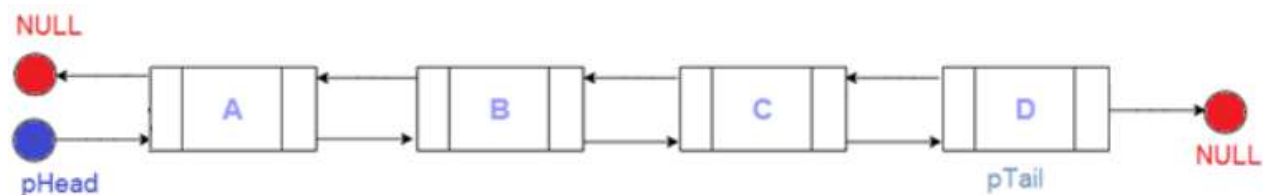
**b. Output:** Trả về kết quả thực hiện chức năng của chương trình.

#### 3.2. Phân tích và ứng dụng cấu trúc dữ liệu và thuật toán trong hệ thống

##### 3.2.1. Cấu trúc dữ liệu danh sách liên kết đôi

Danh sách liên kết kép là danh sách mà các phần tử được tổ chức theo thứ tự tuyến tính. Mỗi phần tử trong danh sách liên kết đôi có kết nối với phần tử đứng ngay trước và ngay sau nó. Dưới đây là một số khái niệm quan trọng về danh sách liên kết đôi:

- **Node:** mỗi node của danh sách liên kết lưu trữ dữ liệu và các con trỏ chứa địa chỉ của node liền trước và liền sau nó. Node còn hay được gọi là một phần tử.
- **Next:** là con trỏ chứa địa chỉ của node ngay liền sau nó.
- **Prev:** là con trỏ chứa địa chỉ của node ngay liền trước nó.
- **Head:** hay còn gọi là First, lưu địa chỉ node đầu tiên trong danh sách liên kết



Hình 3-2: Minh họa cấu trúc dữ liệu danh sách liên kết đôi



### 3.2.2. Thuật toán

#### a. Thuật toán tìm kiếm nhanh (Quick sort)

- Đây là một thuật toán được xây dựng dựa trên kỹ thuật chia để trị (Divide and Conquer)

- Mô tả thuật toán:

+ Base case: nếu danh sách chỉ còn 1 phần tử thì danh sách đó đã được sắp xếp → Không cần thao tác thêm.

+ Chia: chọn 1 phần tử làm phần tử chốt (pivot). Chia danh sách ban đầu thành 2 danh sách gồm danh sách trái (gồm các phần tử nhỏ hơn pivot) và danh sách phải (gồm các phần tử lớn hơn pivot)

+ Trị: Lặp lại đệ quy thuật toán với 2 danh sách trái và phải.

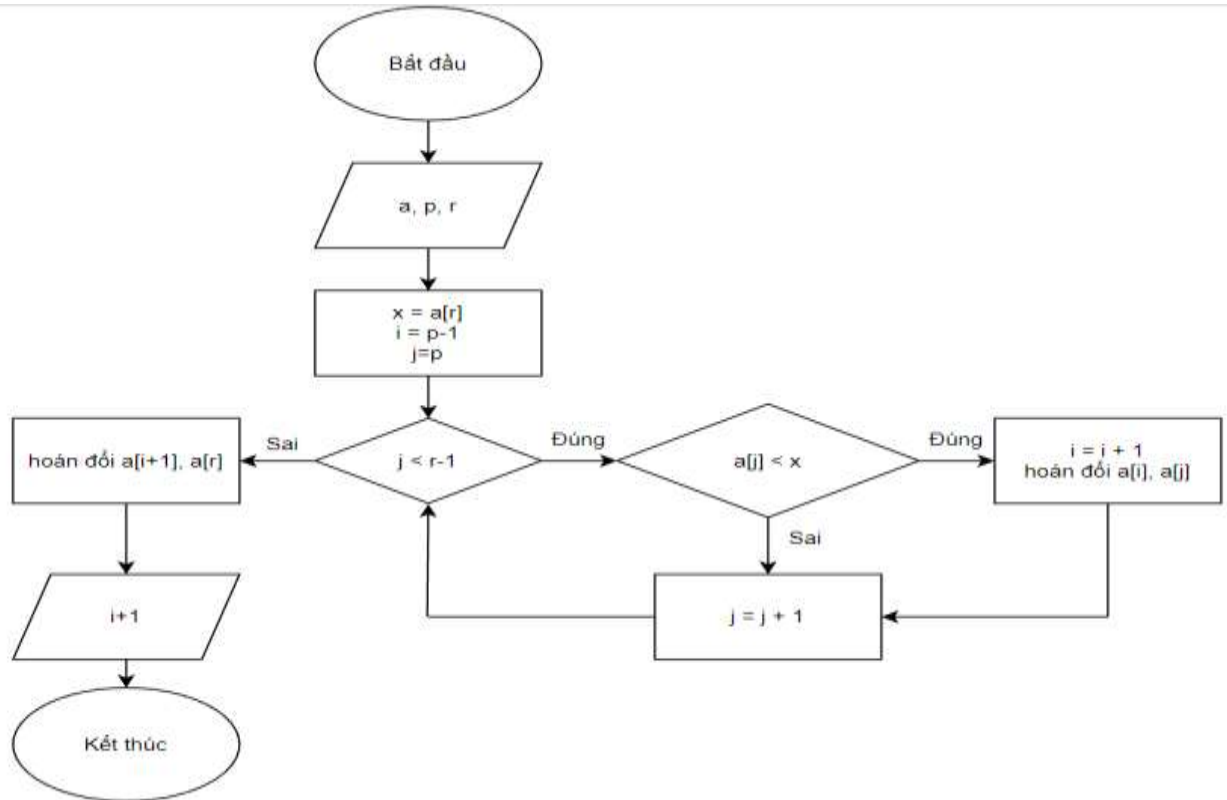
→ Từ đó ta thu được danh sách đã được sắp xếp.

- Ta có thể thấy, việc chọn pivot có vai trò quyết định đối với độ hiệu quả của thuật toán. Chúng ta nên chọn pivot sao cho nó mang giá trị gần nhất với giá trị trung bình của các phần tử trong danh sách. Ngoài ra, một số cách chọn pivot thường gặp: chọn phần tử đứng đầu, phần tử cuối, phần tử đứng giữa,...

- Các bước tiến hành thuật toán:

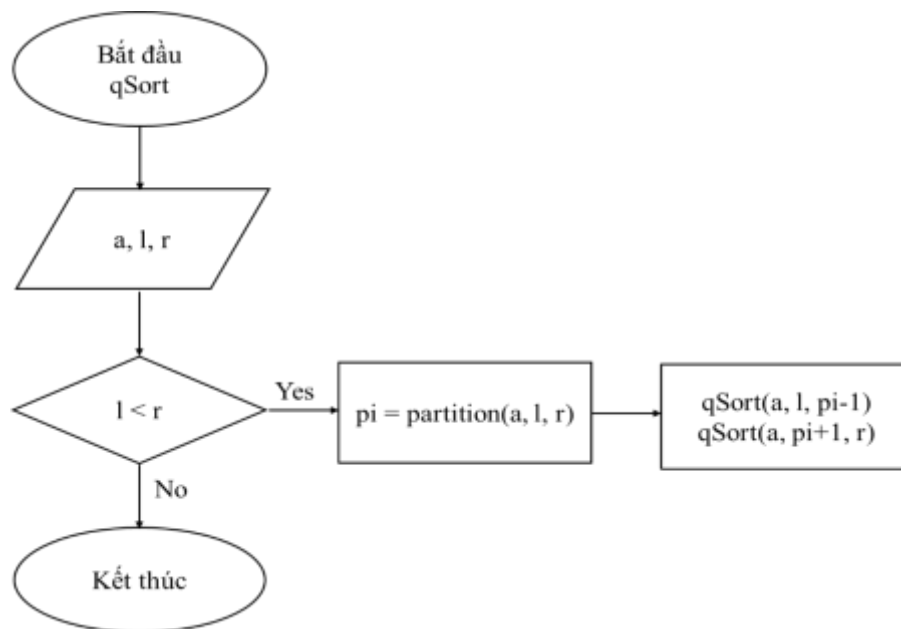
+ **Bước 1:** chọn pivot và phân đoạn

- *Bước 1.1:*  $\text{pivot} = a[r]$  //chọn pivot là số cuối danh sách
- *Bước 1.2:*  $i=p-1, j=p$  //các biến chạy kiểm tra từ đầu danh sách
- *Bước 1.3:* khi  $j < r-1$  thì kiểm tra, đúng thì tăng  $i=i+1$  và hoán đổi  $a[i], a[j]$ . Gán  $j=j+1$  rồi lặp lại bước 1.3
- *Bước 1.4:* hoán đổi  $a[i+1]$  và  $a[r]$  //hoán đổi pivot về đúng vị trí
- *Bước 1.5:* trả về vị trí hiện tại của pivot



Hình 3-1: Sơ đồ khối hàm phân hoạch (partition)

+ Bước 2: gọi đệ quy thuật toán sắp xếp nhanh với 2 danh sách con mới tạo



Hình 3-2: Sơ đồ khối thuật toán sắp xếp nhanh

- Độ phức tạp thuật toán:

Trường hợp	Xảy ra khi	Độ phức tạp
Best case	pivot gần nhất với giá trị trung bình của danh sách	$O(n \log n)$
Average case	khi các phần tử được sắp xếp lộn xộn	$O(n \log n)$
Worst case	pivot là giá trị min/max trong danh sách	$O(n^2)$

*Hình 3-3: Bảng độ phức tạp thuật toán tìm phân hoạch*

#### **b. Thuật toán tìm kiếm tuần tự (Linear search)**

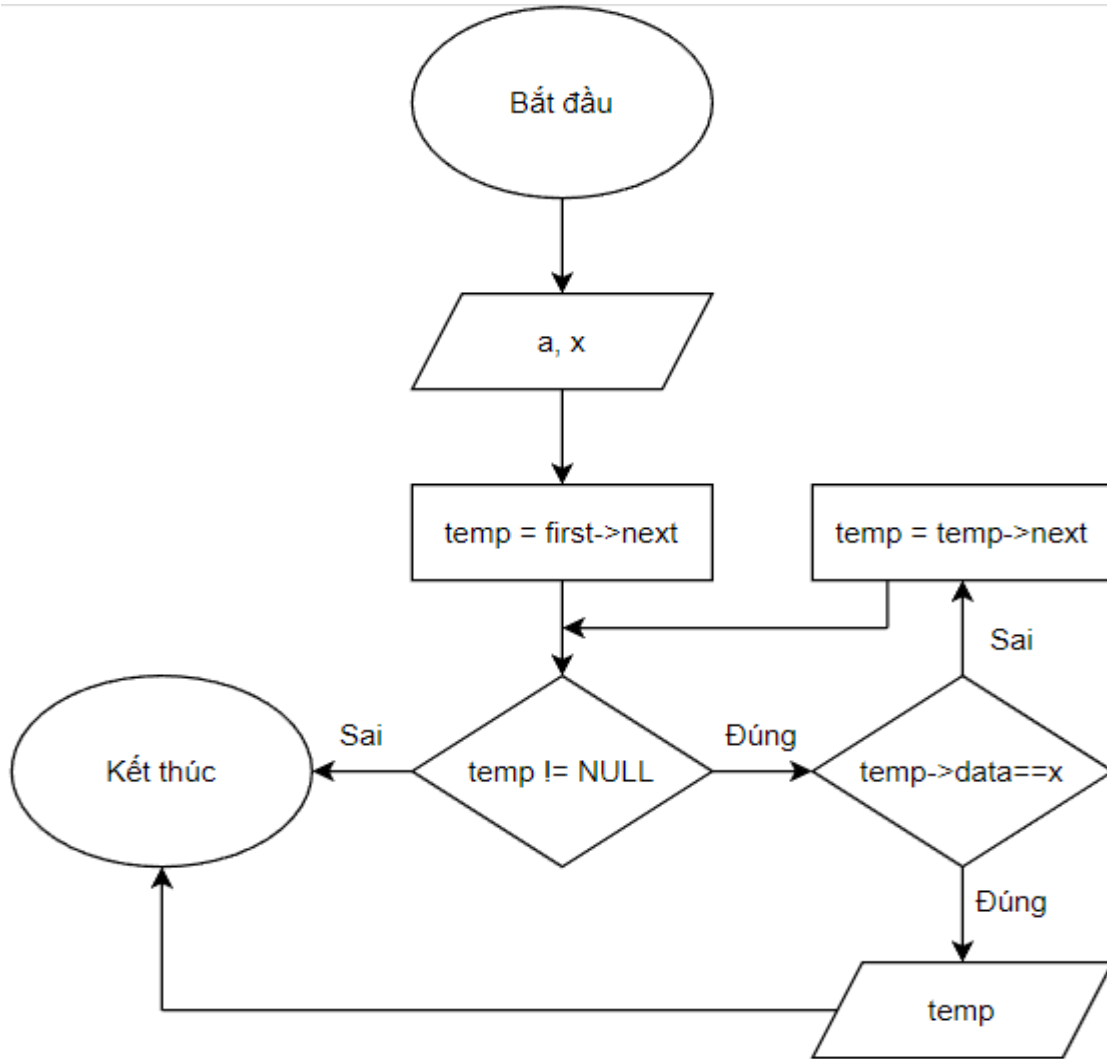
- Là phương pháp tìm kiếm một phần tử được cho trước trong danh sách bằng cách duyệt lần lượt các phần tử từ đầu danh sách cho đến khi gặp được phần tử mong muốn hoặc đã duyệt hết danh sách

- Mô tả thuật toán:

- + Bắt đầu từ phần tử đầu tiên của danh sách, duyệt từ đầu đến cuối với giá trị cần tìm là x.
- + Nếu giá trị phần tử đang duyệt bằng x thì trả về vị trí hiện tại.
- + Nếu không tìm thấy bất cứ phần tử nào khi đã duyệt hết thì trả về -1.

- Các bước tiến hành thuật toán:

- + **Bước 1:** temp = first->next//bắt đầu từ đầu danh sách.
- + **Bước 2:** temp→data==x, đúng thì in ra và kết thúc, sai thì sang bước 3.
- + **Bước 3:** temp = temp->next nếu next khác NULL thì lặp lại bước 2, nếu sai thì kết thúc.



Hình 3-4: Sơ đồ khối thuật toán tìm kiếm tuần tự

- Độ phức tạp thuật toán:

Trường hợp	Xảy ra khi	Độ phức tạp
Best case	giá trị cần tìm nằm đầu danh sách	$O(1)$
Worst case	giá trị cần tìm nằm cuối hoặc không nằm trong danh sách	$O(n)$

Hình 3-5: Độ phức tạp thuật toán tìm kiếm

### 3.2.3. Ứng dụng cấu trúc dữ liệu và thuật toán trong hệ thống

#### a. Ứng dụng cấu trúc dữ liệu

- Xây dựng một khuôn mẫu lớp **List** và khuôn mẫu lớp **Node** để xây dựng cấu trúc dữ liệu cho chương trình, và sử dụng giống như một cấu trúc dữ liệu danh sách liên kết đôi để tạo và lưu trữ một danh sách các đối tượng.

→ Từ đó, ta xây dựng các lớp **billList** và **goodList** dựa trên cấu trúc dữ liệu đã xây dựng class **goodList**: quản lý danh sách hàng hóa, đây là lớp dẫn xuất từ lớp List sử dụng template <good>

#### **Thuộc tính:**

- friend class bill
- friend class billList

#### **Phương thức:**

- constructor
- destructor
- show(): in danh sách hàng ra màn hình
- print(): in file hàng hóa ra màn hình
- nhap(), xuat(): nhập/xuất hàng hóa
  - + *Input*: thông tin hàng hóa
  - + *Output*: kết quả thực thi hành động
- addBill(), subBill(): nhập/xuất hàng hóa dựa vào phiếu nhập/xuất
  - + *Input*: thông tin phiếu
  - + *Output*: danh sách hàng hóa trong kho được cập nhật
- reduce(): giảm đi số lượng của 1 loại hàng hóa trong kho
  - + *Input*: thông tin của hàng hóa để giảm số lượng
  - + *Output*: mặt hàng được cập nhật

- search(): tương tự như search() trong billList

+ *Input*: thuộc tính tìm kiếm, giá trị của thuộc tính tìm kiếm

+ *Output*: tất cả hàng hóa mang giá trị cần tìm

- Đa năng hóa toán tử gán, nhập xuất =, >>, <<

class **billList**: quản lý các phiếu nhập xuất, đây là lớp dẫn xuất từ lớp List sử dụng template <bill>

**Phương thức:**

- constructor

- destructor

- create(): tạo danh sách phiếu

+ *Input*: thông tin của 1 phiếu

+ *Output*: danh sách phiếu mới

- show(): in danh sách phiếu ra màn hình

- print(): in file

+ *Input*: in ra file nhập/xuất

+ *Output*: file được cập nhật

- history(): hiển thị lịch sử nhập xuất

+ *Input*: khoảng thời gian muốn xem

+ *Output*: lịch sử nhập xuất trong thời gian trên

-search(): ứng dụng thuật toán linear search để tìm kiếm, tuy nhiên có một số chỉnh sửa để phù hợp với chương trình.

+ *Input*: phiếu nhập/xuất, thuộc tính cần tìm, giá trị thuộc tính cần tìm

+ *Output*: tất cả các phiếu mang giá trị cần tìm

- Đa năng hóa toán tử nhập, xuất >>, << để nhập, xuất thông tin từ database

## **b. Ứng dụng thuật toán**

- **Sắp xếp danh sách:** Nhận vào thuộc tính sắp xếp và chiều sắp xếp, từ đó thực hiện sắp xếp dựa trên cơ sở của thuật toán Quick Sort.

- **Tìm kiếm đối tượng:** Nhận vào thuộc tính tìm kiếm và giá trị của thuộc tính ấy, từ đó thực hiện việc tìm kiếm trên danh sách dựa trên cơ sở của thuật toán tìm kiếm tuần tự. Ngoài ra còn có thể tìm kiếm theo nhiều thuộc tính (ID, name, quantity, producer,...) để tìm ra chính xác đối tượng mong muốn.

+ Nội dung hàm tìm kiếm:

- Nhận vào số thuộc tính cần tìm và giá trị cần tìm
- Thực hiện duyệt từ đầu danh sách cho đến cuối
- So sánh giá trị của phần tử đang xét với giá trị cần tìm, nếu như bằng nhau thì in ra màn hình và sang phần tử kế tiếp vì kết thúc như thuật toán nguyên bản, điều này khiến độ phức tạp của thuật toán trong mọi trường hợp là  $O(n)$

→ Từ đó, ta thấy rằng hàm tìm kiếm giúp tìm ra tất cả các đối tượng cần tìm một cách chính xác, không bị giới hạn bởi số thuộc tính giúp ta tiết kiệm thời gian khi lọc danh sách, tuy nhiên thời gian thực thi của hàm sẽ lâu hơn so với bình thường.

## **4. PHÂN TÍCH THIẾT KẾ HƯỚNG ĐỐI TƯỢNG**

### **4.1. Cấu trúc hệ thống hướng đối tượng**

#### **4.1.1. Cấu trúc của các lớp hướng đối tượng**

class **pbl2**: quản lý menu, các thao tác, chức năng của hệ thống

##### ***Thuộc tính:***

- goodList khoNhap: danh sách hàng hóa trong kho
- billList phieuXuat: danh sách các phiếu xuất
- billList phieuNhap: danh sách các phiếu nhập

##### ***Phương thức:***

- constructor
- destructor
- welcome(): tạo giao diện bắt đầu chương trình
- menu(): tạo menu
- option1(), option2():
  - + *Input*: dữ liệu cần thiết nhập từ bàn phím (tên người tạo, ngày tạo, mã phiếu, số lượng hàng, thông tin từng mặt hàng)
  - + *Output*: tạo phiếu nhập/xuất tương ứng và nhập/xuất hàng hóa vào kho
- option3():
  - + *Input*: thuộc tính để sắp xếp, chiều sắp xếp
  - + *Output*: danh sách hàng hóa đã sắp xếp
- option4(), option5():
  - + *Input*: thuộc tính để sắp xếp, chiều sắp xếp
  - + *Output*: danh sách phiếu nhập/xuất đã sắp xếp
- option6(): hiển thị danh sách hàng hóa
- option7():
  - + *Input*: khoảng thời gian muốn xem lịch sử
  - + *Output*: lịch sử xuất nhập kho trong khoảng thời gian trên



- option8(), option9(): hiển thị danh sách phiếu nhập/xuất kho
- option10():
  - + *Input*: thuộc tính để tìm kiếm, giá trị của thuộc tính tìm kiếm
  - + *Output*: tất cả các hàng hóa trùng với giá trị tìm kiếm
- option11(), option12(): tương tự với option10()
  - + *Input*: thuộc tính tìm kiếm, giá trị của thuộc tính tìm kiếm
  - + *Output*: tất cả phiếu nhập/xuất trùng với giá trị tìm kiếm

class **Node**: lớp định nghĩa các phần tử trong danh sách

***Thuộc tính:***

- data (kiểu dữ liệu phụ thuộc vào kiểu dữ liệu của node)
- node \*prev: con trỏ lưu địa chỉ của node liền trước
- node \*next: con trỏ lưu địa chỉ của node liền sau

***Phương thức:***

- constructor
- destructor

**class List**: đây là lớp cơ sở để xây dựng 2 lớp danh sách hàng hóa (goodList) và danh sách phiếu (billList)

***Thuộc tính:***

- int n: số lượng phần tử đang có trong danh sách
- node \*Head: con trỏ lưu địa chỉ phần tử đầu tiên của danh sách

***Phương thức:***

- constructor
- destructor
- isEmpty(): kiểm tra danh sách trống
- numOfNode(): trả về số lượng phần tử trong danh sách
- insertHead(), insertTail(): thêm 1 phần tử vào đầu/cuối danh sách

- + *Input*: dữ liệu tương ứng với kiểu dữ liệu của danh sách
- + *Output*: danh sách được cập nhật
- insertAffter(): thêm 1 phần tử vào sau 1 phần tử khác trong danh sách
  - + *Input*: phần tử trước, giá trị cần thêm vào
  - + *Output*: danh sách được cập nhật
- del(): xóa 1 phần tử
  - + *Input*: phần tử cần xóa
  - + *Output*: danh sách được cập nhật
- sort(): sắp xếp danh sách, ứng dụng thuật toán quicksort ở trên để sắp xếp
  - + *Input*: thuộc tính sắp xếp, chiều sắp xếp
  - + *Output*: danh sách được cập nhật

class **bill**: làm việc với phiếu

***Thuộc tính:***

- bool nhapxuat: phiếu nhập/xuất
- string code: mã phiếu
- string nguoiTao: tên người tạo
- date: ngày tạo
- date: ngày nhập xuất
- string fileName
- goodList gl: danh sách hàng hóa trong phiếu
- int total: tổng tiền của các hàng hóa trong phiếu
- friend class billList

**Phương thức:**

- constructor
- destructor
- ngayNhapXuat(): trả về ngày nhập/xuất hàng
- add(): thêm hàng hóa vào phiếu
  - + *Input*: thông tin hàng hóa
  - + *Output*: phiếu được cập nhật
- print(): in file phiếu
- Đa năng hóa toán tử =, <<
- ascend(), descend(): hàm so sánh 2 phiếu
  - + *Input*: thuộc tính muốn so sánh
  - + *Output*: kết quả phép so sánh

class **good**: tương tác với hàng hóa

**Thuộc tính:**

- string code: mã hàng hóa
- string name: tên hàng hóa
- string type: loại hàng hóa
- string unit: đơn vị tính
- date dt: ngày nhập hàng
- string producer: nhà sản xuất
- int quantity: số lượng
- int unit\_price: đơn giá
- int total: tổng tiền

- friend class goodList
- friend class bill
- friend class billList

**Phương thức:**

- constructor
- destructor
- isValid(): kiểm tra tính hợp lệ của thông tin hàng hóa
  - + *Input*: thông tin hàng hóa
  - + *Output*: hợp lệ hay không
- hàm get, set datetime()
- Đa năng hóa toán tử +, -: tăng, giảm số lượng của 1 loại hàng hóa
- Đa năng hóa toán tử gán, nhập, xuất =, >>, <<
- Đa năng hóa toán tử so sánh bằng ==, !=
- ascend(), descend(): so sánh hàng hóa
  - + *Input*: thuộc tính muốn so sánh
  - + *Output*: kết quả phép so sánh

class **date**: tạo đối tượng ngày

**Thuộc tính:**

- int year, month, day

**Phương thức:**

- constructor
- destructor
- isLeap(): kiểm tra năm nhuận

- + *Input*: ngày
- + *Output*: true/false
- isValid(): kiểm tra ngày hợp lệ
  - + *Input*: ngày
  - + *Output*: true/false
- dtAsString(): trả về dạng string của 1 đối tượng date
  - + *Input*: ngày
  - + *Output*: string(ngày)

#### **4.1.2. Quan hệ của các lớp đối tượng**

- Các lớp good, goodList, bill, billList được khai báo là các lớp bạn của nhau nhằm dễ dàng truy xuất các thuộc tính bên trong.
- Lớp List được thiết kế là lớp trừu tượng, mang tính khung cấu trúc cho các lớp danh sách khác, các lớp thừa kế lớp List định nghĩa các phương thức phù hợp cho từng loại danh sách
- Lớp pbl2 là lớp nền của chương trình, cung cấp các giao diện của chương trình cũng như các phương thức gọi đến các thao tác tùy theo mong muốn của người dùng

### **4.2. Kết quả**

#### **4.2.1. Giao diện chính của chương trình**


```
+-----+
|  PBL 2: DO AN CO SO LAP TRINH  |
+-----+

+-----MENU CHUNG-----+
| 1. Quan li hang hoa.      |
| 2. Quan li phieu nhap xuat. |
| 3. Ket thuc.              |
+-----+

Lua chon cua ban la: █
```

Hình 4-1: Giao diện chính của chương trình

#### 4.2.2. Kết quả thực thi chương trình

 database.txt - Notepad

File Edit Format View Help

```
1,ban ui,dien tu,cai,20221204,panasonic,5,2000000,10000000,  
2,laptop,laptop,cai,20221204,asus,1,20000000,20000000,  
3,tu lanh,dien tu,cai,20221204,panasonic,1,20000000,20000000,  
4,tivi,dien tu,cai,20221204,samsung,1,7000000,7000000,  
5,quat,dien tu,cai,20221204,mitsubishi,6,7000000,42000000,  
6,tu lanh,dien tu,cai,20221204,LG,5,22000000,110000000,  
7,may nuoc nong,dien tu,cai,20221204,ariston,4,4000000,16000000,  
8,den,dien tu,cai,20221204,vimite,10,200000,2000000,  
9,noi com dien,dien tu,cai,20221204,panasonic,10,2000000,20000000,
```

Hình 4-2: File dữ liệu ban đầu

#### - Chức năng quản lý hàng hóa:

```
+-----+  
|  PBL 2: DO AN CO SO LAP TRINH  |  
+-----+  
  
+-----MENU HANG HOA-----+  
| 1. Sap xep danh sach hang hoa. |  
| 2. Hien thi danh sach hang hoa. |  
| 3. Hien thi lich su xuất nhập kho. |  
| 4. Tim kiem hang hoa. |  
| 5. Quay lai. |  
+-----+  
  
Lua chon cua ban la: |
```

Hình 4-3: Menu quản lý hàng hóa

```
+-----MENU HÀNG HOA-----+
| 1. Sắp xếp danh sách hàng hóa. |
| 2. Hiện thị danh sách hàng hóa. |
| 3. Hiện thị lịch sử xuất nhập kho. |
| 4. Tìm kiếm hàng hóa. |
| 5. Quay lại. |
+-----+

Lựa chọn của bạn là: 1
-----

Chọn thuộc tính sắp xếp
(1: mã hàng hóa
2: tên hàng hóa
3: loại hàng hóa
4: đơn vị tính
5: ngày nhập hàng
6: nhà sản xuất
7: số lượng
8: đơn giá
9: thành tiền): 1
Chọn thứ tự sắp xếp
(1: tăng dần
2: giảm dần): 2

Thực hiện thành công!
Nhấn phím Enter để tiếp tục.
```

Hình 4-4: Giao diện chức năng sắp xếp danh sách hàng hóa

So lượng mặt hàng: 9

STT	Mã hàng hóa	Tên hàng hóa	Loại hàng hóa	Nhà sản xuất	Ngày nhập hàng	Đơn vị tính	Số lượng	Đơn giá	Thành tiền
1	9	nồi cơm điện	điện tử	panasonic	04/12/2022	cái	10	2000000	20000000
2	8	đèn	điện tử	vinmate	04/12/2022	cái	10	200000	2000000
3	7	máy nước nóng	điện tử	ariston	04/12/2022	cái	4	4000000	16000000
4	6	tủ lạnh	điện tử	LG	04/12/2022	cái	5	22000000	110000000
5	5	quạt	điện tử	mitsubishi	04/12/2022	cái	6	7000000	42000000
6	4	tivi	điện tử	samsung	04/12/2022	cái	1	7000000	7000000
7	3	tủ lạnh	điện tử	panasonic	04/12/2022	cái	1	20000000	20000000
8	2	laptop	laptop	asus	04/12/2022	cái	1	20000000	20000000
9	1	bàn ủi	điện tử	panasonic	04/12/2022	cái	5	2000000	10000000

Thực hiện thành công!  
Nhấn phím Enter để tiếp tục.

Hình 4-5: Giao diện chức năng hiển thị danh sách hàng hóa

Nhập ngày bắt đầu:  
 Ngày: 3  
 Tháng: 12  
 Năm: 2022  
 Nhập ngày kết thúc:  
 Ngày: 5  
 Tháng: 12  
 Năm: 2022

**Lịch sử nhập:**

**Ngày 04/12/2022:**

STT	Mã hàng hóa	Tên hàng hóa	Loại hàng hóa	Nhà sản xuất	Đơn vị tính	Số lượng	Đơn giá	Thành tiền
1	7	máy nước nóng	điện tử	ariston	cái	4	4000000	16000000
2	8	đèn	điện tử	vimite	cái	10	200000	2000000
3	9	nồi cơm điện	điện tử	panasonic	cái	10	200000	2000000
4	10	laptop	điện tử	apple	cái	10	3000000	30000000

**Ngày 04/12/2022:**

STT	Mã hàng hóa	Tên hàng hóa	Loại hàng hóa	Nhà sản xuất	Đơn vị tính	Số lượng	Đơn giá	Thành tiền
1	1	tivi	điện tử	samsung	cái	1	7000000	7000000
2	2	laptop	laptop	asus	cái	1	2000000	2000000
3	3	tủ lạnh	điện tử	panasonic	cái	1	2000000	2000000

**Ngày 04/12/2022:**

STT	Mã hàng hóa	Tên hàng hóa	Loại hàng hóa	Nhà sản xuất	Đơn vị tính	Số lượng	Đơn giá	Thành tiền
1	4	tivi	điện tử	samsung	cái	1	7000000	7000000
2	1	bàn ủi	điện tử	panasonic	cái	5	200000	1000000
3	5	quạt	điện tử	mitsubishi	cái	6	700000	4200000
4	6	tủ lạnh	điện tử	LG	cái	5	2200000	11000000

Hình 4-6: Giao diện hiển thị chức năng hiển thị lịch sử xuất nhập (1)

**Lịch sử xuất:**

**Ngày 04/12/2022:**

STT	Mã hàng hóa	Tên hàng hóa	Loại hàng hóa	Nhà sản xuất	Đơn vị tính	Số lượng	Đơn giá	Thành tiền
1	10	laptop	điện tử	apple	cái	10	3000000	30000000

**Ngày 04/12/2022:**

STT	Mã hàng hóa	Tên hàng hóa	Loại hàng hóa	Nhà sản xuất	Đơn vị tính	Số lượng	Đơn giá	Thành tiền
1	1	tivi	điện tử	samsung	cái	1	7000000	7000000

Hình 4-7: Giao diện chức năng hiển thị lịch sử xuất nhập (2)



```
-----MENU HÀNG HOA-----
1. Sắp xếp danh sách hàng hóa.
2. Hiện thị danh sách hàng hóa.
3. Hiện thị lịch sử xuất nhập kho.
4. Tìm kiếm hàng hóa.
5. Quay lại.

Lựa chọn của bạn là: 4
-----
Bạn muốn tìm kiếm theo (1: code, 2: filter): 1
Nhập mã hàng hóa cần tìm: 1
Tìm thấy 1 mặt hàng!
```

STT	Mã hàng hóa	Tên hàng hóa	Loại hàng hóa	Nhà sản xuất	Ngày nhập hàng	Đơn vị tính	Số lượng	Đơn giá	Thành tiền
1	1	bàn ủi	điện tử	panasonic	04/12/2022	cái	5	2000000	10000000

Hình 4-8: Giao diện chức năng tìm kiếm hàng hóa theo 1 thuộc tính

```
-----MENU HÀNG HOA-----
1. Sắp xếp danh sách hàng hóa.
2. Hiện thị danh sách hàng hóa.
3. Hiện thị lịch sử xuất nhập kho.
4. Tìm kiếm hàng hóa.
5. Quay lại.

Lựa chọn của bạn là: 4
-----
Bạn muốn tìm kiếm theo (1: code, 2: filter): 2
Chọn các filter mà bạn muốn
(1: tên hàng hóa, 2: loại hàng hóa, 3: nhà sản xuất,
4: ngày nhập hàng, 5: số lượng, 6: đơn giá): 1 3
Nhập tên hàng hóa cần tìm: tủ lạnh
Nhập nhà sản xuất cần tìm: LG
Tìm thấy 1 mặt hàng!
```

STT	Mã hàng hóa	Tên hàng hóa	Loại hàng hóa	Nhà sản xuất	Ngày nhập hàng	Đơn vị tính	Số lượng	Đơn giá	Thành tiền
1	6	tủ lạnh	điện tử	LG	04/12/2022	cái	5	22000000	110000000

Hình 4-9: Giao diện chức năng tìm kiếm hàng hóa theo nhiều thuộc tính

- Chức năng quản lý phiếu:

```
+-----+
| PBL 2: DO AN CO SO LAP TRINH |
+-----+

+-----+-----+
|          MENU PHIEU          |
+-----+-----+
| 1. Tao phieu nhap hang hoa (nhap hang hoa). |
| 2. Tao phieu xuất hang hoa (xuất hang hoa). |
| 3. Sap xep danh sach phieu nhap.             |
| 4. Sap xep danh sach phieu xuất.             |
| 5. Hien thi danh sach phieu nhap kho.        |
| 6. Hien thi danh sach phieu xuất kho.        |
| 7. Tim kiem phieu nhap kho.                  |
| 8. Tim kiem phieu xuất kho.                  |
| 9. Quay lai.                                |
+-----+-----+

Lua chon cua ban la: █
```

Hình 4-10: Menu quản lý phiếu

```
+-----+-----+
|          MENU PHIEU          |
+-----+-----+
| 1. Tao phieu nhap hang hoa (nhap hang hoa). |
| 2. Tao phieu xuất hang hoa (xuất hang hoa). |
| 3. Sap xep danh sach phieu nhap.             |
| 4. Sap xep danh sach phieu xuất.             |
| 5. Hien thi danh sach phieu nhap kho.        |
| 6. Hien thi danh sach phieu xuất kho.        |
| 7. Tim kiem phieu nhap kho.                  |
| 8. Tim kiem phieu xuất kho.                  |
| 9. Quay lai.                                |
+-----+-----+

Lua chon cua ban la: 1
-----
Nhap ten nguoi tao: anh
Nhap ngay nhap :
    Ngay: 5
    Thang: 12
    Nam: 2022
Nhap ma phieu: id004
Nhap so luong mat hang: 1
Nhap danh sach hang hoa:
Nhap thong tin hang hoa thu 1:
    Nhap ma hang hoa: 10
    Nhap ten hang hoa: tivi
    Nhap loai hang hoa: dien tu
    Nhap don vi tinh: cai
    Nhap nha san xuất: panasonic
    Nhap so luong: 10
    Nhap don gia: 6000000

Thuc hien thanh cong!
Nhap phim Enter de tiep tục.
█
```

Hình 4-11: Giao diện chức năng tạo phiếu nhập

So lương mặt hàng: 10

STT	Mã hàng hóa	Tên hàng hóa	Loại hàng hóa	Nhà sản xuất	Ngày nhập hàng	Đơn vị tính	Số lượng	Đơn giá	Thành tiền
1	1	bàn ủi	điện tử	panasonic	04/12/2022	cai	5	2000000	10000000
2	2	laptop	laptop	asus	04/12/2022	cai	1	20000000	20000000
3	3	tủ lạnh	điện tử	panasonic	04/12/2022	cai	1	20000000	20000000
4	4	tivi	điện tử	samsung	04/12/2022	cai	1	7000000	7000000
5	5	quạt	điện tử	mitsubishi	04/12/2022	cai	6	7000000	42000000
6	6	tủ lạnh	điện tử	LG	04/12/2022	cai	5	22000000	110000000
7	7	máy nước nóng	điện tử	ariston	04/12/2022	cai	4	4000000	16000000
8	8	đèn	điện tử	vimite	04/12/2022	cai	10	200000	2000000
9	9	nồi cơm điện	điện tử	panasonic	04/12/2022	cai	10	2000000	20000000
10	10	tivi	điện tử	panasonic	05/12/2022	cai	10	6000000	60000000

Hình 4-12: Kho sau khi tạo phiếu nhập

```

+-----MENU PHIEU-----+
| 1. Tạo phiếu nhập hàng hóa (nhập hàng hóa). |
| 2. Tạo phiếu xuất hàng hóa (xuất hàng hóa). |
| 3. Sắp xếp danh sách phiếu nhập. |
| 4. Sắp xếp danh sách phiếu xuất. |
| 5. Hiện thị danh sách phiếu nhập kho. |
| 6. Hiện thị danh sách phiếu xuất kho. |
| 7. Tìm kiếm phiếu nhập kho. |
| 8. Tìm kiếm phiếu xuất kho. |
| 9. Quay lại. |
+-----+

Lựa chọn của bạn là: 2
-----
Nhập tên người tạo: anh
Nhập ngày xuất :
    Ngày: 5
    Tháng: 12
    Năm: 2022
Nhập mã phiếu: id006
Nhập số lượng mặt hàng: 1
Nhập danh sách hàng hóa:
Nhập thông tin hàng hóa thu 1:
    Nhập mã hàng hóa: 8
    Tên hàng hóa: đèn
    Loại hàng hóa: điện tử
    Nhà sản xuất: vimite
    Đơn vị tính: cai
    Ngày nhập hàng: 04/12/2022
    Số lượng: 10
    Đơn giá: 200000
    Nhập số lượng cần xuất: 10
    Nhập đơn giá cần xuất: 200000

Thực hiện thành công!
Nhấn phím Enter để tiếp tục.
  
```

Hình 4-13: Giao diện chức năng tạo phiếu xuất

So Luong mat hang: 9

STT	Ma hang hoa	Ten hang hoa	Loai hang hoa	Nha san xuat	Ngay nhap hang	Don vi tinh	So luong	Don gia	Thanh tien
1	1	ban ui	dien tu	panasonic	04/12/2022	cai	5	2000000	10000000
2	2	laptop	laptop	asus	04/12/2022	cai	1	20000000	20000000
3	3	tu lanh	dien tu	panasonic	04/12/2022	cai	1	20000000	20000000
4	4	tivi	dien tu	samsung	04/12/2022	cai	1	7000000	7000000
5	5	quat	dien tu	mitsubishi	04/12/2022	cai	6	7000000	42000000
6	6	tu lanh	dien tu	LG	04/12/2022	cai	5	22000000	110000000
7	7	may nuoc nong	dien tu	ariston	04/12/2022	cai	4	4000000	16000000
8	9	noi com dien	dien tu	panasonic	04/12/2022	cai	10	2000000	20000000
9	10	tivi	dien tu	panasonic	05/12/2022	cai	10	6000000	60000000

Hình 4-14: Kho sau khi tạo phiếu xuất

```

+-----MENU PHIEU-----+
| 1. Tao phieu nhap hang hoa (nhap hang hoa). |
| 2. Tao phieu xuat hang hoa (xuat hang hoa). |
| 3. Sap xep danh sach phieu nhap. |
| 4. Sap xep danh sach phieu xuat. |
| 5. Hien thi danh sach phieu nhap kho. |
| 6. Hien thi danh sach phieu xuat kho. |
| 7. Timkiem phieu nhap kho. |
| 8. Timkiem phieu xuat kho. |
| 9. Quay lai. |
+-----+

Lua chon cua ban la: 3
-----
Chon thuoc tinh sap xep
(1: ma phieu
2: nguoi tao
3: ngay tao
4: ngay nhap xuat
5: so mat hang
6: tong gia tien): 5
Chon thu tu sap xep
(1: tang dan
2: giam dan): 2

Thuc hien thanh cong!
Nhan phim Enter de tiep tuc.

```

Hình 4-15: Giao diện chức năng sắp xếp danh sách phiếu

```
+-----MENU PHIEU-----+
| 1. Tao phieu nhap hang hoa (nhap hang hoa).
| 2. Tao phieu xuất hang hoa (xuất hang hoa).
| 3. Sap xep danh sach phieu nhap.
| 4. Sap xep danh sach phieu xuất.
| 5. Hien thi danh sach phieu nhap kho.
| 6. Hien thi danh sach phieu xuất kho.
| 7. Tim kiem phieu nhap kho.
| 8. Tim kiem phieu xuất kho.
| 9. Quay lai.
+-----+

Lua chon cua ban la: 5
-----
So luong phieu: 4
-----
+-----+-----+-----+-----+-----+-----+-----+
| STT | Ma phieu |   Ngươi Tao   | Ngay tao | Ngay nhap hang | So mat hang | Tong gia tien |
+-----+-----+-----+-----+-----+-----+-----+
| 1 | id002 | anh | 04/12/2022 | 04/12/2022 | 4 | 169000000 |
| 2 | id003 | anh | 04/12/2022 | 04/12/2022 | 4 | 338000000 |
| 3 | id01 | anh | 04/12/2022 | 04/12/2022 | 3 | 47000000 |
| 4 | id004 | anh | 04/12/2022 | 05/12/2022 | 1 | 60000000 |
+-----+-----+-----+-----+-----+-----+-----+

Thuc hien thanh cong!
Nhan phim Enter de tiep tuc.
```

Hình 4-16: Giao diện chức năng hiển thị danh sách phiếu

```
+-----MENU PHIEU-----+
| 1. Tao phieu nhap hang hoa (nhap hang hoa).
| 2. Tao phieu xuất hang hoa (xuất hang hoa).
| 3. Sap xep danh sach phieu nhap.
| 4. Sap xep danh sach phieu xuất.
| 5. Hien thi danh sach phieu nhap kho.
| 6. Hien thi danh sach phieu xuất kho.
| 7. Tim kiem phieu nhap kho.
| 8. Tim kiem phieu xuất kho.
| 9. Quay lai.
+-----+

Lua chon cua ban la: 7
-----
Ban muon tim kiem theo (1: code, 2: filter): 1
Nhap ma phieu can tim: id002

Tim thay 1 phieu!
-----
+-----+-----+-----+-----+-----+-----+-----+
| STT | Ma phieu |   Ngươi Tao   | Ngay tao | Ngay nhap hang | So mat hang | Tong gia tien |
+-----+-----+-----+-----+-----+-----+-----+
| 1 | id002 | anh | 04/12/2022 | 04/12/2022 | 4 | 169000000 |
+-----+-----+-----+-----+-----+-----+-----+

Thuc hien thanh cong!
Nhan phim Enter de tiep tuc.
```

Hình 4-17: Giao diện chức năng tìm kiếm phiếu theo 1 thuộc tính

```
+-----MENU PHIEU-----+
| 1. Tao phieu nhap hang hoa (nhap hang hoa).
| 2. Tao phieu xuat hang hoa (xuat hang hoa).
| 3. Sap xep danh sach phieu nhap.
| 4. Sap xep danh sach phieu xuat.
| 5. Hien thi danh sach phieu nhap kho.
| 6. Hien thi danh sach phieu xuat kho.
| 7. Timkiem phieu nhap kho.
| 8. Timkiem phieu xuat kho.
| 9. Quay lai.
+-----+

Lua chon cua ban la: 7
-----
Ban muon timkiem theo (1: code, 2: filter): 2

Chon cac filter ma ban muon
(1: nguoi tao, 2: ngay tao, 3: ngay nhap hang,
4: so mat hang, 5: tong gia tien): 1 2

Nhap nguoi tao can tim: anh

Nhap ngay tao bat dau:
    Ngay: 4
    Thang: 12
    Nam: 2022
Nhap ngay tao ket thuc:
    Ngay: 4
    Thang: 12
    Nam: 2022

Tim thay 4 phieu!

+-----+-----+-----+-----+-----+-----+-----+
| STT | Ma phieu |   Ngươi Tạo   |   Ngày tạo   |   Ngày nhập hàng   |   Số mặt hàng   |   Tổng giá tiền   |
+-----+-----+-----+-----+-----+-----+-----+
| 1 | id002 | anh | 04/12/2022 | 04/12/2022 | 4 | 169000000 |
| 2 | id003 | anh | 04/12/2022 | 04/12/2022 | 4 | 338000000 |
| 3 | id01 | anh | 04/12/2022 | 04/12/2022 | 3 | 47000000 |
| 4 | id004 | anh | 04/12/2022 | 05/12/2022 | 1 | 60000000 |
+-----+-----+-----+-----+-----+-----+-----+
```

Hình 4-18: Giao diện chức năng tìm kiếm phiếu theo nhiều thuộc tính

### 4.2.3. Nhận xét

- Chương trình thực hiện được các chức năng cơ bản của công tác quản lý xuất, nhập kho
- Giao diện chưa bắt mắt, hiển thị còn khá dài dòng
- Việc thiết kế cấu trúc hướng đối tượng chưa thật sự tối ưu, việc tái sử dụng chưa hoàn thiện vì còn nhiều phần code bị lặp lại.

## **5. KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN**

### **5.1. Kết luận**

- Nhìn chung, chương trình đáp ứng được yêu cầu của đề tài. Trong quá trình làm đồ án, nhóm đã tìm hiểu và nắm được các kiến thức về danh sách liên kết, lập trình hướng đối tượng và các khái niệm liên quan như thừa kế, đa hình, khuôn mẫu. Áp dụng được các thuật toán để giải quyết các bài toán nhỏ hơn (tìm kiếm, sắp xếp). Tuy nhiên vẫn có vài khuyết điểm trong việc thiết kế cấu trúc hướng đối tượng.

### **5.2. Hướng phát triển**

- Xây dựng giao diện đồ họa đẹp đẽ, bắt mắt hơn cho người dùng
- Tối ưu cấu trúc hướng đối tượng để tăng tính hiệu quả của chương trình, tránh các tình trạng quan hệ vòng giữa các lớp
- Phát triển thêm chức năng quản lý để sát với thực tiễn

## **TÀI LIỆU THAM KHẢO**

- [1] Nguyễn Văn Hiếu, Danh sách liên kết đôi, <https://blog.luyencode.net/danh-sach-lien-ket-doi/>
- [2] GeeksforGeeks, Doubly linked list, <https://www.geeksforgeeks.org/introduction-and-insertion-in-a-doubly-linked-list/>
- [3] Lê Thị Mỹ Hạnh, Lập trình hướng đối tượng C++, 2022
- [4] GeeksforGeeks, QuickSort, <https://www.geeksforgeeks.org/quick-sort/>
- [5] GeeksforGeeks, Linear Search, <https://www.geeksforgeeks.org/linear-search/>
- [6] GeeksforGeeks, Exception Handling in C++,  
<https://www.geeksforgeeks.org/exception-handling-c/>