

**BỘ GIÁO DỤC VÀ ĐÀO TẠO**  
**TRƯỜNG ĐẠI HỌC ĐẠI NAM**



## **BÀI TẬP LỚN**

**TÊN HỌC PHẦN: CẤU TRÚC DỮ LIỆU VÀ GIẢI THUẬT**  
**ĐỀ TÀI: XÂY DỰNG BÀI TOÁN QUẢN LÝ SẢN PHẨM SỬ**  
**DỤNG CÂY NHỊ PHÂN**

**Giáo viên hướng dẫn: Đẩu Hải Phong**

**Sinh viên thực hiện:**

- 1. Bùi Quang Tuấn**
- 2. Nguyễn Thanh Tùng**
- 3. Phong Ngọc Anh**

**Hà Nội, 2024**

BỘ GIÁO DỤC VÀ ĐÀO TẠO  
TRƯỜNG ĐẠI HỌC ĐẠI NAM



# BÀI TẬP LỚN

TÊN HỌC PHẦN: CẤU TRÚC DỮ LIỆU VÀ GIẢI THUẬT  
ĐỀ TÀI: XÂY DỰNG BÀI TOÁN QUẢN LÝ SẢN PHẨM SỬ  
DỤNG CÂY NHỊ PHÂN

STT	Mã Sinh Viên	Họ và Tên	Ngày Sinh	Điểm	
				Bảng Số	Bảng Chữ
1	1771020718	Bùi Quang Tuấn	28/04/2005		
2	1771020729	Nguyễn Thanh Tùng	01/09/2004		
3	1771020056	Phong Ngọc Anh	23/10/2005		

CÁN BỘ CHẤM THI

Hà Nội, 2024

## LỜI NÓI ĐẦU

Trong thời đại công nghệ thông tin phát triển như hiện nay, việc hiểu và áp dụng các cấu trúc dữ liệu và giải thuật là vô cùng cần thiết và quan trọng. Môn học "Cấu trúc dữ liệu và Giải thuật" không chỉ cung cấp những kiến thức cơ bản mà còn trang bị cho chúng ta những công cụ cần thiết để giải quyết các bài toán phức tạp trong thực tế.

Trong phạm vi của bài tập lớn này, chúng tôi xin giới thiệu về "Cây nhị phân" - một trong những cấu trúc dữ liệu cơ bản nhưng có tính ứng dụng cao. Cây nhị phân là một cấu trúc dữ liệu dạng cây trong đó mỗi nút có tối đa hai con. Đặc biệt, cây nhị phân sản phẩm là một ứng dụng cụ thể của cây nhị phân, nơi mỗi nút đại diện cho một sản phẩm trong một hệ thống quản lý hàng hóa, quản lý cửa hàng trực tuyến hoặc các ứng dụng thương mại điện tử khác. Trong phần bài báo cáo này chúng tôi sẽ giới thiệu về các chương trình mà nhóm chúng tôi đã xây dựng để quản lý các sản phẩm bao gồm như: Nhà sản phẩm, Tên sản phẩm, Số lượng , Đơn vị ...

## MỤC LỤC

CHƯƠNG 1. GIỚI THIỆU ĐỀ TÀI.....	6
1.1. Giới thiệu đề tài .....	6
1.2. Mục đích của đề tài .....	6
CHƯƠNG 2. PHÂN TÍCH ĐỀ TÀI .....	7
2.1. Dữ liệu lưu trữ .....	7
2.2. Cấu trúc dữ liệu .....	7
2.3. Các chức năng .....	8
2.4. Thiết kế menu .....	13
CHƯƠNG 3. PHÂN TÍCH CHƯƠNG TRÌNH.....	15
3.1. Khai báo thư viện .....	15
3.2. Khởi tạo danh sách sản phẩm .....	15
3.3. Kiểm tra rỗng .....	16
3.4. Thêm phần tử vào danh sách sản phẩm .....	16
3.5. In danh sách sản phẩm .....	18
3.6. Đọc dữ liệu từ file vào danh sách sản phẩm .....	18
3.8. In danh sách sản phẩm theo chiều tăng của giá cả .....	21
3.9. In danh sách sản phẩm theo chiều giảm của giá cả .....	22
3.10 In ra tổng giá trị của mỗi loại sản phẩm .....	23
3.11 Tính và in ra số loại mặt hàng trong danh sách. ....	24
3.12 Tính và in ra tổng số lượng của tất cả các loại sản phẩm .....	25
3.14 Tìm và in ra thông tin của những sản phẩm thuộc hãng x .....	27
3.15 Tìm và in ra những sản phẩm có đơn vị tính khối lượng là x .....	28
3.16 In ra thông tin của những loại sản phẩm có số lượng trên x .....	29

3.17 In ra thông tin của những loại sản phẩm có số lượng dưới $x$ .....	30
3.18 In ra thông tin của sản phẩm có giá cao nhất .....	31
3.19 In ra thông tin của sản phẩm có giá thấp nhất .....	32
3.20 In ra những sản phẩm có giá trên mức $x$ .....	33
3.21 In ra những sản phẩm có giá dưới mức $x$ .....	34
3.22 Tính tổng số lượng của những loại sản phẩm có giá trên mức $x$ .....	35
3.23 Tính tổng số lượng của những loại sản phẩm có giá dưới mức $x$ .....	36
3.24 Xoá sản phẩm có giá dưới mức $x$ khỏi danh sách .....	37
3.25 Xoá sản phẩm có giá trên $x$ khỏi danh sách .....	39
3.26 Tính số lượng nhà cung cấp trong danh sách .....	41
3.27 Thống kê các loại sản phẩm theo nhà cung cấp .....	43
3.28 Xoá sản phẩm có giá cao nhất .....	45
3.29 Xoá sản phẩm có giá thấp nhất .....	45
3.30 Xoá sản phẩm có giá bằng $x$ .....	46
3.31 Thoát khỏi chương trình .....	50
CHƯƠNG 4. MÃ NGUỒN CHƯƠNG TRÌNH .....	51
KẾT LUẬN .....	74
DANH MỤC TÀI LIỆU THAM KHẢO .....	75

## CHƯƠNG 1. GIỚI THIỆU ĐỀ TÀI

### 1.1. Giới thiệu đề tài

Đề tài “Xây dựng bài toán quản lý sản phẩm sử dụng cây nhị phân” là một bài toán thực tế trong việc quản lý lưu trữ thông tin của các loại sản phẩm và phân phối chúng dựa trên quy luật về giá. Đề tài này tập trung vào việc xây dựng một hệ thống phân tầng giúp lưu trữ và tìm kiếm các mặt hàng trở lên đơn giản bằng cách sử dụng cơ chế của cây nhị phân.

Cây nhị phân (Binary Tree) là một cấu trúc dữ liệu cây mà mỗi nút có nhiều nhất hai nút con, được gọi là con trái (left child) và con phải (right child). Trong đề tài này, cây nhị phân được sử dụng để lưu trữ các thông tin của sản phẩm giúp cho việc tìm kiếm và so sánh trở nên đơn giản hơn.

### 1.2. Mục đích của đề tài

Việc xây dựng đề tài này giúp tạo ra một chương trình giúp cho việc lưu trữ và thay đổi thông tin của các sản phẩm trở nên đơn giản thông qua 30 chức năng được hiển thị ra cho người dùng lựa chọn giúp họ thao tác và làm việc với danh sách sản phẩm được tạo ra và lưu trữ bằng cây nhị phân.

Ngoài ra việc thực hiện đề tài cũng nhằm khai thác và tìm hiểu thêm nhiều ứng dụng và cách dùng của cây nhị phân thông qua nguyên lý hoạt động và cấu tạo của nó. Không chỉ có duyệt và tìm kiếm, mà còn có xóa, sắp xếp, sàng lọc,...

## CHƯƠNG 2. PHÂN TÍCH ĐỀ TÀI

### 2.1. Dữ liệu lưu trữ

- Dữ liệu cần quản lý và lưu trữ về sản phẩm gồm:
  - Tên nhà cung cấp – kiểu: string
  - Tên sản phẩm – kiểu: string
  - Số lượng – kiểu: int
  - Đơn giá – kiểu: float
  - Đơn vị Sản phẩm – kiểu: string
- Bộ dữ liệu được lưu trữ trong tệp văn bản SanPham.txt lần lượt gồm các thông tin (Tên nhà cung cấp, Tên sản phẩm, Số lượng, Đơn giá, Đơn vị sản phẩm) của 2 sản phẩm như sau:

NhaCungCap1

TenSanPham1

100

10.5

kg

NhaCungCap2

TenSanPham2

150

20.75

lit

### 2.2. Cấu trúc dữ liệu

```
struct SanPham
```

```
{
```

```

string NhaCungCap;

string TenSanPham;

int SoLuong;

float DonGia;

string DVT;

};

```

### 2.3. Các chức năng

- Khởi tạo danh sách sản phẩm
  - Khởi tạo cho nó một danh sách để lưu trữ dữ liệu sản phẩm
- Kiểm tra danh sách
  - Kiểm tra danh sách có dữ liệu chưa:
    - + Nếu có thì trả về True
    - + Chưa có thì trả về False
- Thêm sản phẩm vào danh sách
  - Cho người dùng nhập vào:
    - + Nhà cung cấp
    - + Tên sản phẩm
    - + Số lượng
    - + Đơn giá
    - + Đơn vị
- In danh sách sản phẩm
  - In ra thông tin của sản phẩm gồm:
    - + Nhà cung cấp
    - + Tên sản phẩm
    - + Số lượng
    - + Đơn giá
    - + Đơn vị
  - Nếu không có dữ liệu in ra , trả về “ Danh sách Rỗng”



- Đọc dữ liệu từ file vào danh sách
  - Lấy dữ liệu từ file SanPham.txt và thêm vào danh sách :
- Ghi dữ liệu từ danh sách vào file
  - Lấy hết dữ liệu mà đã thêm vào danh sách Sản phẩm và ghi vào danh sách có tên file là Sp.txt:
- In danh sách sản phẩm theo chiều tăng giá cả
  - In ra màn hình với sản phẩm giá theo chiều tăng :
- In danh sách sản phẩm theo chiều giảm giá cả
  - In ra màn hình với sản phẩm giá theo chiều giảm:
- In và tính tổng giá trị của mỗi loại sản phẩm
  - Hiện thị ra màn hình với tổng giá của mỗi loại sản-Hiện thị gồm:
    - + Nhà cung cấp
    - + Tên sản phẩm
    - + Số lượng
    - + Đơn giá
    - + Đơn vị
    - + Tổng giá
- Tính và in ra số loại mặt hàng trong danh sách.
  - Hiện thị ra màn hình số lượng mặt hàng đã thêm vào trong danh sách.
- Tính và in ra tổng số lượng của tất cả các loại sản phẩm
  - Hiện thị tổng số lượng của sản phẩm.
- Tìm và in ra thông tin của sản phẩm có tên là x
  - Nhập tên sản phẩm muốn tìm :
    - + Nếu tìm thấy:
      - \* In ra sản phẩm gồm: Tên nhà cung cấp, Tên sản phẩm, Số lượng, Đơn giá, Đơn vị
    - + Nếu không tìm thấy:
      - \* Hiện thị ra màn hình: Không tìm thấy
- Tìm và in ra thông tin của những sản phẩm thuộc hãng x

- Nhập tên nhà cung cấp muốn tìm :
  - + Nếu tìm thấy:
    - \* In ra sản phẩm gồm: Tên nhà cung cấp, Tên sản phẩm, Số lượng, Đơn giá, Đơn vị
  - + Nếu không tìm thấy:
    - \* Hiện thị ra màn hình: Không tìm thấy
- Tìm và in ra những sản phẩm có đơn vị tính khối lượng là x
  - Nhập Đơn vị mà cần tìm:
    - + Nếu thấy thì hiện thị hết tất cả sản phẩm ra:
    - + Nếu không tìm thấy sẽ thông báo “ Không tìm thấy sản phẩm nào có khối lượng nào như vậy”
- In ra thông tin của những loại sản phẩm có số lượng trên x
  - Nhập số lượng :
    - + Nếu có số lượng nào lớn hơn thì hiện thị ra màn hình.
    - + Nếu không tìm thấy số lượng nào lớn hơn nó thì hiện thị ra màn hình thông báo “ không có số lượng nào trên x”
- In ra thông tin của những loại sản phẩm có số lượng dưới x
  - Nhập số lượng :
    - + Nếu có số lượng nào lớn hơn thì hiện thị ra màn hình.
    - + Nếu không tìm thấy số lượng nào nhỏ hơn nó thì hiện thị ra màn hình thông báo “ không có số lượng nào nhỏ hơn x”
- In ra thông tin của sản phẩm có giá cao nhất
  - Hiện thị ra màn hình sản phẩm có giá sản phẩm cao nhất.
  - Nếu danh sách rỗng thì thông báo người dùng là “ Danh sách rỗng !”
- In ra thông tin của sản phẩm có giá thấp nhất
  - Hiện thị ra màn hình sản phẩm có giá sản phẩm thấp nhất.
  - Nếu danh sách rỗng thì thông báo người dùng là “ Danh sách rỗng !”
- In ra những sản phẩm có giá trên mức x

- Nhập giá sản phẩm :
  - + Nếu có giá sản phẩm nào lớn hơn thì hiện thị ra màn hình.
  - + Nếu không tìm thấy giá sản phẩm nào lớn hơn nó thì hiện thị ra màn hình thông báo “ không có số lượng nào trên x”
- In ra những sản phẩm có giá dưới mức x
  - Nhập giá sản phẩm :
    - + Nếu có giá sản phẩm nào dưới mức thì hiện thị ra màn hình.
    - + Nếu không tìm thấy giá sản phẩm nào dưới mức nó thì hiện thị ra màn hình thông báo “ không có số lượng nào trên x”
- Tính tổng số lượng của những loại sản phẩm có giá trên mức x
  - Nhập mức giá :
    - + Nếu có giá sản phẩm nào trên mức thì hiện thị ra tổng số lượng.
    - + Nếu không tìm thấy giá sản phẩm nào trên mức nó thì hiện thị ra màn hình thông báo “ không tìm thấy giá trị nào trên x”
- Tính tổng số lượng của những loại sản phẩm có giá dưới mức x
  - Nhập mức giá :
    - + Nếu có giá sản phẩm nào dưới mức thì hiện thị ra tổng số lượng.
    - + Nếu không tìm thấy giá sản phẩm nào dưới mức nó thì hiện thị ra màn hình thông báo “ không tìm thấy giá trị nào dưới x”
- Xóa sản phẩm có giá dưới mức x khỏi danh sách
  - Nhập mức giá :
    - + Nếu có giá sản phẩm nào dưới mức thì xóa đi tất cả sản phẩm nào dưới mức giá. Sau khi xóa xong sẽ hiện ra màn hình thông báo người dùng “ Đã xóa thành công”
    - + Nếu không tìm thấy giá sản phẩm nào dưới mức hiện thị ra màn hình thông báo “ không có sản phẩm nào có giá dưới x để xóa ”
- Xóa sản phẩm có giá trên x khỏi danh sách
  - Nhập mức giá :

+ Nếu có giá sản phẩm nào dưới mức thì xóa đi tất cả sản phẩm nào trên mức giá. Sau khi xóa xong sẽ hiện ra màn hình thông báo người dùng “ Đã xóa thành công”

+ Nếu không tìm thấy giá sản phẩm nào trên mức hiện thị ra màn hình thông báo “ không có sản phẩm nào có giá dưới x để xóa ”

■ Tính số lượng nhà cung cấp trong danh sách

- Nếu danh sách không rỗng thì hiện thị ra màn hình tổng số lượng nhà cung cấp có trong danh sách.

- Nếu danh sách rỗng thì hiện thị ra màn hình thông báo người dùng “ Danh sách rỗng!”

■ In ra thông tin các loại sản phẩm của từng nhà cung cấp

- Nếu danh sách không rỗng thì hiện thị ra màn hình thông tin nhà cung cấp, tên sản phẩm, đơn giá, số lượng, đơn vị.

- Nếu danh sách rỗng thì hiện thị ra màn hình thông báo người dùng “ Danh sách rỗng!”

■ Xoá sản phẩm có giá cao nhất

- Kiểm tra nếu danh sách không rỗng thì tìm giá cao nhất và xóa nó khỏi danh sách. Sau khi xóa xong sẽ hiện thị ra màn hình “đã xóa”

-Nếu danh sách rỗng thì hiện thị ra màn hình thông báo người dùng “ Danh sách rỗng!”

■ Xoá sản phẩm có giá thấp nhất

- Kiểm tra nếu danh sách không rỗng thì tìm giá thấp nhất và xóa nó khỏi danh sách. Sau khi xóa xong sẽ hiện thị ra màn hình “đã xóa”

-Nếu danh sách rỗng thì hiện thị ra màn hình thông báo người dùng “ Danh sách rỗng!”

■ Xoá sản phẩm có giá bằng x

- Nhập mức giá :

+ Nếu tìm thấy mức giá trong danh sách thì nó sẽ xóa nó khỏi danh sách. Sau khi xóa xong sẽ thông báo người dùng “ Đã xóa”

+ Nếu tìm không tìm thấy hiện ra màn hình thông báo người dùng “ không tìm thấy thông mức giá nào bằng x”

■ Thoát khỏi chương trình

- Thoát khỏi chương trình và tắt chương trình:

## 2.4. Thiết kế menu

<-----Các chức năng----->

1. Khởi tạo danh sách sản phẩm
2. Kiểm tra danh sách
3. Thêm sản phẩm vào danh sách
4. In danh sách sản phẩm
5. Đọc dữ liệu từ file vào danh sách
6. Ghi dữ liệu từ danh sách vào file
7. In danh sách sản phẩm theo chiều tăng giá cả
8. In danh sách sản phẩm theo chiều giảm giá cả
9. In và tính tổng giá trị của mỗi loại sản phẩm
10. Tính và in ra số loại mặt hàng trong danh sách.
11. Tính và in ra tổng số lượng của tất cả các loại sản phẩm
12. Tìm và in ra thông tin của sản phẩm có tên là x
13. Tìm và in ra thông tin của những sản phẩm thuộc hãng x
14. Tìm và in ra những sản phẩm có đơn vị tính khối lượng là x

15. In ra thông tin của những loại sản phẩm có số lượng trên x
16. In ra thông tin của những loại sản phẩm có số lượng dưới x
17. In ra thông tin của sản phẩm có giá cao nhất
18. In ra thông tin của sản phẩm có giá thấp nhất
19. In ra những sản phẩm có giá trên mức x
20. In ra những sản phẩm có giá dưới mức x
21. Tính tổng số lượng của những loại sản phẩm có giá trên mức x
22. Tính tổng số lượng của những loại sản phẩm có giá dưới mức x
23. Xóa sản phẩm có giá dưới mức x khỏi danh sách
24. Xóa sản phẩm có giá trên x khỏi danh sách
25. Tính số lượng nhà cung cấp trong danh sách
26. In ra thông tin các loại sản phẩm của từng nhà cung cấp
27. Xóa sản phẩm có giá cao nhất
28. Xóa sản phẩm có giá thấp nhất
29. Xóa sản phẩm có giá bằng x
30. Thoát khỏi chương trình

## CHƯƠNG 3. PHÂN TÍCH CHƯƠNG TRÌNH

### 3.1. Khai báo thư viện

Các thư viện được dùng trong Chương trình quản lý Sản phẩm là:

```
#include<iostream> // Thư viện mặc định phải có khi dùng cpp
#include<fstream> // Thư viện này giúp ta đọc file viết file
#include<iomanip> // Thư viện này giúp lấy các số thập phân sau dấu phẩy
#include<string> // Thư viện giúp xử lý chuỗi ký tự trong mảng
#include<cstdlib> // Thư viện này là thư viện chuẩn với nhiều chức năng
khác nhau nhưng trong báo cáo này nó dùng để xóa đi các phần thừa khi in
ra màn hình
```

### 3.2. Khởi tạo danh sách sản phẩm

#### Code Khởi tạo danh sách sản phẩm

```
void Init(Tree &T)
{
    T=NULL;
}
```

#### Hiện thị ra màn hình

```
Da khoi tao danh sach san pham!
<-----Cac Chuc Nang----->
1. Khoi tao danh sach san pham
2. Kiem tra danh sach
3. Them san pham vao danh sach
4. In danh sach san pham
5. Doc thong tin cac san pham tu file vao danh sach
6. Ghi du lieu tu danh sach vao file
7. In danh sach san pham theo chieu tang gia
8. In danh sach san pham theo chieu giam gia
9. In ra tong gia tri cua moi loai san pham
10. Tinh va in ra so loai mat hang trong danh sach
11. Tinh va in ra tong so luong cua tat ca cac loai san pham
12. Tim va in ra thong tin cua san pham co ten x
13. Tim va in ra thong tin cua nhung loai san pham thuoc hang x
14. Tim va in ra thong tin nhung loai san pham co don vi tinh khi loai luong la x
15. In ra thong tin nhung loai san pham co so luong tren x
16. In ra thong tin nhung loai san pham co so luong duoi x
17. In ra thong tin san pham co gia cao nhất
18. In ra thong tin san pham co gia thap nhất
19. In ra thong tin nhung san pham co gia tren muc x
20. In ra thong tin nhung san pham co gia duoi muc x
21. Tong so luong nhung loai san pham co gia tren muc x
22. Tong so luong nhung loai san pham co gia duoi muc x
23. Xoa nhung loai san pham co gia duoi muc x
24. Xoa nhung loai san pham co gia tren muc x
25. Tinh so luong nha cung cap
26. Thong ke so cac loai san pham theo tung nha cung cap
27. Xoa san pham co gia cao nhất
28. Xoa san pham co gia thap nhất
29. Xoa san pham co gia bang x
30. Thoat khoi chuong trinh
Moi chon chuc nang (1-30)
==>
```

### 3.3. Kiểm tra rỗng

#### Code Kiểm tra có rỗng không

```
bool isEmpty(Tree T)
{
    if (T==NULL)
    {
        return true;
    }
    else
    {
        return false;
    }
}
```

#### Hiện thị ra màn hình

```
Danh sach khong rong!

<-----Cac Chuc Nang----->
1. Khoi tao danh sach san pham
2. Kiem tra danh sach
3. Them san pham vao danh sach
4. In danh sach san pham
5. Doc thong tin cac san pham tu file vao danh sach
6. Ghi du lieu tu danh sach vao file
7. In danh sach san pham theo chieu tang gia
8. In danh sach san pham theo chieu giam gia
9. In ra tong gia tri cua moi loai san pham
10. Tinh va in ra so loai mat hang trong danh sach
11. Tinh va in ra tong so luong cua tat ca cac loai san pham
12. Tim va in ra thong tin cua san pham co ten x
13. Tim va in ra thong tin cua nhung loai san pham thuoc hang x
14. Tim va in ra thong tin nhung loai san pham co don vi tinh khioi luong la x
15. In ra thong tin nhung loai san pham co so luong tren x
16. In ra thong tin nhung loai san pham co so luong duoi x
17. In ra thong tin san pham co gia cao nhat
18. In ra thong tin san pham co gia thap nhat
19. In ra thong tin nhung san pham co gia tren muc x
20. In ra thong tin nhung san pham co gia duoi muc x
21. Tong so luong nhung loai san pham co gia tren muc x
22. Tong so luong nhung loai san pham co gia duoi muc x
23. Xoa nhung loai san pham co gia duoi muc x
24. Xoa nhung loai san pham co gia tren muc x
25. Tinh so luong nha cung cap
26. Thong ke so cac loai san pham theo tung nha cung cap
27. Xoa san pham co gia cao nhat
28. Xoa san pham co gia thap nhat
29. Xoa san pham co gia bang x
30. Thoat khoi chuong trinh
Moi chon chuc nang (1-30)
==>
```

### 3.4. Thêm phần tử vào danh sách sản phẩm

- Cho người dùng nhập vào:



- + Nhà cung cấp
- + Tên sản phẩm
- + Số lượng
- + Đơn giá
- + Đơn vị

#### Code Thêm phần tử vào danh sách

```
void AddNode(Tree &T, Item x)
{
    if (T==NULL)
    {
        T= new TNode;
        T->Key=x;
        T->Left=NULL;
        T->Right=NULL;
    }
    else
    {
        if (T->Key.DonGia>x.DonGia)
        {
            AddNode(T->Left,x);
        }
        else
        {
            AddNode(T->Right,x);
        }
    }
}
```

#### Hiện thị ra màn hình

```
Nhap vao nhung thong tin cua san pham can them:
Ten nha cung cap: code
Ten san pham: i love you code
So luong: 1
Don gia: 0
Don Vi: iu
```

### 3.5. In danh sách sản phẩm

#### In danh tất cả các sản phẩm

```
void InDanhSach(Tree T)
{
    if (T!=NULL)
    {
        cout<<T->Key.NhaCungCap<<"\t\t"
        <<T->Key.TenSanPham<<"\t\t"
        <<T->Key.SoLuong<<"\t\t"
        <<T->Key.DonGia<<"\t\t"
        <<T->Key.DVT<<endl;
        InDanhSach(T->Left);
        InDanhSach(T->Right);
    }
}
```

#### Hiện thị ra màn hình

NhaCC	TenSp	SL	Gia	DonVi
code	ilove code	1	0	iu

### 3.6. Đọc dữ liệu từ file vào danh sách sản phẩm

- Lấy dữ liệu từ file SanPham.txt để thêm vào cây nhị phân

#### Đọc dữ liệu từ file vào

```

void ReadFileToDSSP(Tree &T)
{
    ifstream f1;
    f1.open("SanPham.txt");
    if (!f1)
    {
        cout<<"Mo tep loi!\n";
    }
    else
    {
        Item x;
        while (!f1.eof())
        {
            f1.ignore();
            getline(f1,x.NhaCungCap);
            getline(f1,x.TenSanPham);
            f1>>x.SoLuong;
            f1>>x.DonGia;
            f1>>x.DVT;
            AddNode(T,x);
        }
        cout<<"Da doc xong!\n";
    }
    f1.close();
}

```

**Hiện thị ra màn hình**

```

Da doc xong!

<-----Cac Chuc Nang----->
1. Khoi tao danh sach san pham
2. Kiem tra danh sach
3. Them san pham vao danh sach
4. In danh sach san pham
5. Doc thong tin cac san pham tu file vao danh sach
6. Ghi du lieu tu danh sach vao file
7. In danh sach san pham theo chieu tang gia
8. In danh sach san pham theo chieu giam gia
9. In ra tong gia tri cua moi loai san pham
10. Tinh va in ra so loai mat hang trong danh sach
11. Tinh va in ra tong so luong cua tat ca cac loai san pham
12. Tim va in ra thong tin cua san pham co ten x
13. Tim va in ra thong tin cua nhung loai san pham thuoc hang x
14. Tim va in ra thong tin nhung loai san pham co don vi tinh khoi luong la x
15. In ra thong tin nhung loai san pham co so luong tren x
16. In ra thong tin nhung loai san pham co so luong duoi x
17. In ra thong tin san pham co gia cao nhat
18. In ra thong tin san pham co gia thap nhat
19. In ra thong tin nhung san pham co gia tren muc x
20. In ra thong tin nhung san pham co gia duoi muc x
21. Tong so luong nhung loai san pham co gia tren muc x
22. Tong so luong nhung loai san pham co gia duoi muc x
23. Xoa nhung loai san pham co gia duoi muc x
24. Xoa nhung loai san pham co gia tren muc x
25. Tinh so luong nha cung cap
26. Thong ke so cac loai san pham theo tung nha cung cap
27. Xoa san pham co gia cao nhat
28. Xoa san pham co gia thap nhat
29. Xoa san pham co gia bang x
30. Thoat khoi chuong trinh
Moi chon chuc nang (1-30)
==>

```

### 3.7. Ghi dữ liệu từ danh sách vào file

- Đọc dữ liệu từ cây nhị phân ghi vào file Sp.txt

#### Ghi dữ liệu vào file

```

void WriteToFile(ofstream &file, Tree T){
    if(T!=NULL){
        file<<T->Key.NhaCungCap<<"\n"
        <<T->Key.TenSanPham<<"\n"
        <<T->Key.SoLuong<<"\n"
        <<T->Key.DonGia<<"\n"
        <<T->Key.DVT<<endl;
        WriteToFile(file, T->Left);
        WriteToFile(file, T->Right);
    }
}

void WriteFileDSSP(Tree T){

```

```

    ofstream file;
    file.open("Sp.txt");
    if(!file){
        cout<<"File không tồn tại! ";
    }else{
        file<<"\n";
        WriteToFile(file,T);
        file.close();
        cout<<"Đã ghi xong\n";
    }
}

```

### Hiện thị ra màn hình

```

Đã ghi xong
<-----Cac Chuc Nang----->
1. Khởi tạo danh sách sản phẩm
2. Kiểm tra danh sách
3. Thêm sản phẩm vào danh sách
4. In danh sách sản phẩm
5. Đọc thông tin các sản phẩm từ file vào danh sách
6. Ghi dữ liệu từ danh sách vào file
7. In danh sách sản phẩm theo chiều tăng giá
8. In danh sách sản phẩm theo chiều giảm giá
9. In ra tổng giá trị của mọi loại sản phẩm
10. Tính và in ra số loại mặt hàng trong danh sách
11. Tính và in ra tổng số lượng của tất cả các loại sản phẩm
12. Tìm và in ra thông tin của sản phẩm có tên x
13. Tìm và in ra thông tin của những loại sản phẩm thuộc hàng x
14. Tìm và in ra thông tin những loại sản phẩm có đơn vị tính khối lượng là x
15. In ra thông tin những loại sản phẩm có số lượng trên x
16. In ra thông tin những loại sản phẩm có số lượng dưới x
17. In ra thông tin sản phẩm có giá cao nhất
18. In ra thông tin sản phẩm có giá thấp nhất
19. In ra thông tin những sản phẩm có giá trên mức x
20. In ra thông tin những sản phẩm có giá dưới mức x
21. Tổng số lượng những loại sản phẩm có giá trên mức x
22. Tổng số lượng những loại sản phẩm có giá dưới mức x
23. Xóa những loại sản phẩm có giá dưới mức x
24. Xóa những loại sản phẩm có giá trên mức x
25. Tính số lượng nhà cung cấp
26. Thống kê số các loại sản phẩm theo từng nhà cung cấp
27. Xóa sản phẩm có giá cao nhất
28. Xóa sản phẩm có giá thấp nhất
29. Xóa sản phẩm có giá bằng x
30. Thoát khỏi chương trình
Mọi chọn chức năng (1-30)
==>

```

### 3.8. In danh sách sản phẩm theo chiều tăng của giá cả

```
void LNR(Tree T)
```

```

{
    if (T!=NULL)
    {
        LNR(T->Left);
        cout<<T->Key.NhaCungCap<<"\t\t"
        <<T->Key.TenSanPham<<"\t\t"
        <<T->Key.SoLuong<<"\t\t"
        <<T->Key.DonGia<<"\t\t"
        <<T->Key.DVT<<endl;
        LNR(T->Right);
    }
}

```

### Hiện thị ra màn hình

NhaCC	TenSp	SL	Gia	DonVi
NhaCungCap1	TenSanPham1	100	10.5	kg
NhaCungCap2	TenSanPham2	150	20.75	lit
NhaCungCap3	TenSanPham3	200	30	kg
NhaCungCap4	TenSanPham4	250	40.25	lit
NhaCungCap5	TenSanPham5	300	50.5	kg
NhaCungCap6	TenSanPham6	350	60.75	lit
NhaCungCap7	TenSanPham7	400	70	kg
NhaCungCap8	TenSanPham8	450	80.25	lit
NhaCungCap9	TenSanPham9	500	90.5	kg
NhaCungCap10	TenSanPham10	550	100.75	lit

```

<-----Cac Chuc Nang----->
1. Khoi tao danh sach san pham
2. Kiem tra danh sach
3. Them san pham vao danh sach
4. In danh sach san pham
5. Doc thong tin cac san pham tu file vao danh sach
6. Ghi du lieu tu danh sach vao file
7. In danh sach san pham theo chieu tang gia
8. In danh sach san pham theo chieu giam gia
9. In ra tong gia tri cua moi loai san pham
10. Tinh va in ra so loai mat hang trong danh sach

```

### 3.9. In danh sách sản phẩm theo chiều giảm của giá cả

#### In danh sách sản phẩm theo chiều giảm

```

void RNL(Tree T)
{
    if (T!=NULL)
    {

```

```

    RNL(T->Right);
    cout<<T->Key.NhaCungCap<<"\t\t"
    <<T->Key.TenSanPham<<"\t\t"
    <<T->Key.SoLuong<<"\t\t"
    <<T->Key.DonGia<<"\t\t"
    <<T->Key.DVT<<endl;
    RNL(T->Left);
}
}

```

### Hiện thị ra màn hình

NhaCC	TenSp	SL	Gia	DonVi
NhaCungCap10	TenSanPham10	550	100.75	lit
NhaCungCap9	TenSanPham9	500	90.5	kg
NhaCungCap8	TenSanPham8	450	80.25	lit
NhaCungCap7	TenSanPham7	400	70	kg
NhaCungCap6	TenSanPham6	350	60.75	lit
NhaCungCap5	TenSanPham5	300	50.5	kg
NhaCungCap4	TenSanPham4	250	40.25	lit
NhaCungCap3	TenSanPham3	200	30	kg
NhaCungCap2	TenSanPham2	150	20.75	lit
NhaCungCap1	TenSanPham1	100	10.5	kg

<-----Cac Chuc Nang----->  
 1. Khoi tao danh sach san pham  
 2. Kiem tra danh sach  
 3. Them san pham vao danh sach

### 3.10 In ra tổng giá trị của mỗi loại sản phẩm

#### Tổng Giá trị của mỗi loại sản phẩm

```

void InGiaTri(Tree T)
{
    if (T!=NULL)
    {
        cout<<T->Key.NhaCungCap<<"\t\t"
        <<T->Key.TenSanPham<<"\t\t"
        <<T->Key.SoLuong<<"\t\t"
        <<T->Key.DonGia<<"\t\t"
    }
}

```

```

    <<T->Key.DonGia*T->Key.SoLuong<<endl;
    InGiaTri(T->Left);
    InGiaTri(T->Right);
}
}

```

#### Hiện thị ra màn hình

NhaCC	TenSp	SL	Gia	TongGia
NhaCungCap1	TenSanPham1	100	10.5	1050
NhaCungCap2	TenSanPham2	150	20.75	3112.5
NhaCungCap3	TenSanPham3	200	30	6000
NhaCungCap4	TenSanPham4	250	40.25	10062.5
NhaCungCap5	TenSanPham5	300	50.5	15150
NhaCungCap6	TenSanPham6	350	60.75	21262.5
NhaCungCap7	TenSanPham7	400	70	28000
NhaCungCap8	TenSanPham8	450	80.25	36112.5
NhaCungCap9	TenSanPham9	500	90.5	45250
NhaCungCap10	TenSanPham10	550	100.75	55412.5

### 3.11 Tính và in ra số loại mặt hàng trong danh sách.

#### Code

```

int DemSoSanPham(Tree T)
{
    if (T != NULL)
    {
        return 1 + DemSoSanPham(T->Left) + DemSoSanPham(T->Right);
    }
    else
    {
        return 0;
    }
}

```

#### Hiện thị ra màn hình



```

Tong so luong san pham: 10

<-----Cac Chuc Nang----->
1. Khoi tao danh sach san pham
2. Kiem tra danh sach
3. Them san pham vao danh sach
4. In danh sach san pham
5. Doc thong tin cac san pham tu file vao danh sach
6. Ghi du lieu tu danh sach vao file
7. In danh sach san pham theo chieu tang gia
8. In danh sach san pham theo chieu giam gia
9. In ra tong gia tri cua moi loai san pham
10. Tinh va in ra so loai mat hang trong danh sach
11. Tinh va in ra tong so luong cua tat ca cac loai san pham
12. Tim va in ra thong tin cua san pham co ten x
13. Tim va in ra thong tin cua nhung loai san pham thuoc hang x
14. Tim va in ra thong tin nhung loai san pham co don vi tinh khoi luong la x
15. In ra thong tin nhung loai san pham co so luong tren x
16. In ra thong tin nhung loai san pham co so luong duoi x
17. In ra thong tin san pham co gia cao nhất
18. In ra thong tin san pham co gia thap nhất
19. In ra thong tin nhung san pham co gia tren muc x
20. In ra thong tin nhung san pham co gia duoi muc x
21. Tong so luong nhung loai san pham co gia tren muc x
22. Tong so luong nhung loai san pham co gia duoi muc x
23. Xoa nhung loai san pham co gia duoi muc x
24. Xoa nhung loai san pham co gia tren muc x
25. Tinh so luong nha cung cap
26. Thong ke so cac loai san pham theo tung nha cung cap
27. Xoa san pham co gia cao nhất
28. Xoa san pham co gia thap nhất
29. Xoa san pham co gia bang x
30. Thoat khoi chuong trinh
Moi chon chuc nang (1-30)
==>

```

### 3.12 Tính và in ra tổng số lượng của tất cả các loại sản phẩm

- Tính và in ra tổng tất cả số lượng sản phẩm, có trong cấu trúc cây nhị phân

#### Tổng số lượng sản phẩm

```

float Total_Sp(Tree T){
    if(T!=NULL){
        return(T->Key.SoLuong +Total_Sp(T->Left) +Total_Sp(T->Right));
    }else{
        return 0;
    }
}

```

#### Hiện thị ra màn hình

```

Tong Sl sp: 3250

<-----Cac Chuc Nang----->
1. Khoi tao danh sach san pham
2. Kiem tra danh sach
3. Them san pham vao danh sach
4. In danh sach san pham
5. Doc thong tin cac san pham tu file vao danh sach
6. Ghi du lieu tu danh sach vao file
7. In danh sach san pham theo chieu tang gia
8. In danh sach san pham theo chieu giam gia
9. In ra tong gia tri cua moi loai san pham
10. Tinh va in ra so loai mat hang trong danh sach
11. Tinh va in ra tong so luong cua tat ca cac loai san pham
12. Tim va in ra thong tin cua san pham co ten x
13. Tim va in ra thong tin cua nhung loai san pham thuoc hang x
14. Tim va in ra thong tin nhung loai san pham co don vi tinh khoi luong la x
15. In ra thong tin nhung loai san pham co so luong tren x
16. In ra thong tin nhung loai san pham co so luong duoi x
17. In ra thong tin san pham co gia cao nhat
18. In ra thong tin san pham co gia thap nhat
19. In ra thong tin nhung san pham co gia tren muc x
20. In ra thong tin nhung san pham co gia duoi muc x
21. Tong so luong nhung loai san pham co gia tren muc x
22. Tong so luong nhung loai san pham co gia duoi muc x
23. Xoa nhung loai san pham co gia duoi muc x
24. Xoa nhung loai san pham co gia tren muc x
25. Tinh so luong nha cung cap
26. Thong ke so cac loai san pham theo tung nha cung cap
27. Xoa san pham co gia cao nhat
28. Xoa san pham co gia thap nhat
29. Xoa san pham co gia bang x
30. Thoat khoi chuong trinh
Moi chon chuc nang (1-30)
==>

```

### 3.13 Tìm và in ra thông tin của sản phẩm có tên là x

- Nhập tên sản phẩm cần tìm

#### Code

```

void TimSanPham(Tree T, string x, bool &kthay)
{
    if (T != NULL)
    {
        TimSanPham(T->Left, x, kthay);
        if (T->Key.TenSanPham == x)
        {
            cout << T->Key.NhaCungCap << "\t\t"
            << T->Key.TenSanPham << "\t\t"
            << T->Key.SoLuong << "\t\t"
            << T->Key.DonGia << "\t\t"
            << T->Key.DVT << endl;
            kthay = true;
        }
    }
}

```

```

        TimSanPham(T->Right, x, kthay);
    }
}

```

### Hiện thị ra màn hình

```

Nhap ten san pham can tim: TenSanPham2
NhaCungCap2          TenSanPham2          150          20.75          lit
<-----Cac Chuc Nang----->
1. Khoi tao danh sach san pham
2. Kiem tra danh sach
3. Them san pham vao danh sach
4. In danh sach san pham
5. Doc thong tin cac san pham tu file vao danh sach
6. Ghi du lieu tu danh sach vao file
7. In danh sach san pham theo chieu tang gia
8. In danh sach san pham theo chieu giam gia
9. In ra tong gia tri cua moi loai san pham
10. Tinh va in ra so loai mat hang trong danh sach
11. Tinh va in ra tong so luong cua tat ca cac loai san pham
12. Tim va in ra thong tin cua san pham co ten x
13. Tim va in ra thong tin cua nhung loai san pham thuoc hang x
14. Tim va in ra thong tin nhung loai san pham co don vi tinh khioi luong la x
15. In ra thong tin nhung loai san pham co so luong tren x
16. In ra thong tin nhung loai san pham co so luong duoi x
17. In ra thong tin san pham co gia cao nhat
18. In ra thong tin san pham co gia thap nhat
19. In ra thong tin nhung san pham co gia tren muc x
20. In ra thong tin nhung san pham co gia duoi muc x
21. Tong so luong nhung loai san pham co gia tren muc x
22. Tong so luong nhung loai san pham co gia duoi muc x
23. Xoa nhung loai san pham co gia duoi muc x
24. Xoa nhung loai san pham co gia tren muc x
25. Tinh so luong nha cung cap
26. Thong ke so cac loai san pham theo tung nha cung cap
27. Xoa san pham co gia cao nhat
28. Xoa san pham co gia thap nhat
29. Xoa san pham co gia bang x
30. Thoat khoi chuong trinh
Moi chon chuc nang (1-30)

```

### 3.14 Tìm và in ra thông tin của những sản phẩm thuộc hãng x

- Nhập Tên Nhà Cung Cấp muốn tìm:

#### Code

```

void TimSanPhamHangX(Tree T, string x, bool &kthay)
{
    if (T != NULL)
    {
        TimSanPhamHangX(T->Left, x, kthay);
        if (T->Key.NhaCungCap == x)
        {
            cout << T->Key.NhaCungCap << "\t\t" << T->Key.TenSanPham

```

```

<< "\t\t" << T->Key.Soluong << "\t\t" << T->Key.DonGia << "\t\t" <<
T->Key.DVT << endl;

    kthay = true;

}

TimSanPhamHangX(T->Right, x, kthay);

}

}

```

### Hiện thị ra màn hình

```

Nhap ten hang can tim: NhaCungCap1
NhaCungCap1          TenSanPham1          100          10.5          kg

<-----Cac Chuc Nang----->
1. Khoi tao danh sach san pham
2. Kiem tra danh sach
3. Them san pham vao danh sach
4. In danh sach san pham
5. Doc thong tin cac san pham tu file vao danh sach
6. Ghi du lieu tu danh sach vao file
7. In danh sach san pham theo chieu tang gia
8. In danh sach san pham theo chieu giam gia
9. In ra tong gia tri cua moi loai san pham
10. Tinh va in ra so loai mat hang trong danh sach
11. Tinh va in ra tong so luong cua tat ca cac loai san pham
12. Tim va in ra thong tin cua san pham co ten x
13. Tim va in ra thong tin cua nhung loai san pham thuoc hang x
14. Tim va in ra thong tin nhung loai san pham co don vi tinh khoi luong la x
15. In ra thong tin nhung loai san pham co so luong tren x
16. In ra thong tin nhung loai san pham co so luong duoi x
17. In ra thong tin san pham co gia cao nhất
18. In ra thong tin san pham co gia thap nhất
19. In ra thong tin nhung san pham co gia tren muc x
20. In ra thong tin nhung san pham co gia duoi muc x
21. Tong so luong nhung loai san pham co gia tren muc x
22. Tong so luong nhung loai san pham co gia duoi muc x
23. Xoa nhung loai san pham co gia duoi muc x
24. Xoa nhung loai san pham co gia tren muc x
25. Tinh so luong nha cung cap
26. Thong ke so cac loai san pham theo tung nha cung cap
27. Xoa san pham co gia cao nhất
28. Xoa san pham co gia thap nhất
29. Xoa san pham co gia bang x
30. Thoat khoi chuong trinh
Moi chon chuc nang (1-30)

```

### 3.15 Tìm và in ra những sản phẩm có đơn vị tính khối lượng là x

- Nhập đơn vị muốn tìm trong cây nhị phân

#### Code

```

void TimSanPhamDVT(Tree T, string x, bool &kthay)
{
    if (T != NULL)
    {
        TimSanPhamDVT(T->Left, x, kthay);

        if (T->Key.DVT == x)

```

```

    {
        cout << T->Key.NhaCungCap << "\t\t" << T->Key.TenSanPham
        << "\t\t" << T->Key.SoLuong << "\t\t" << T->Key.DonGia << "\t\t" <<
        T->Key.DVT << endl;
        kthay = true;
    }
    TimSanPhamDVT(T->Right, x, kthay);
}
}

```

### Hiện thị ra màn hình

```

Nhap san pham co khoi luong can tim: kg
NhaCungCap1      TenSanPham1      100      10.5      kg
NhaCungCap3      TenSanPham3      200      30      kg
NhaCungCap5      TenSanPham5      300      50.5      kg
NhaCungCap7      TenSanPham7      400      70      kg
NhaCungCap9      TenSanPham9      500      90.5      kg

<-----Cac Chuc Nang----->
1. Khoi tao danh sach san pham
2. Kiem tra danh sach
3. Them san pham vao danh sach
4. In danh sach san pham

```

### 3.16 In ra thông tin của những loại sản phẩm có số lượng trên x

- Hiện thị tất cả ra màn hình với số lượng trên x

#### Code

```

void Sl_up_x(Tree T, int x, bool& i){
    if(T!=NULL){
        Sl_up_x(T->Left, x, i);
        Sl_up_x(T->Right, x, i);
        if(x<T->Key.SoLuong){
            cout<<T->Key.NhaCungCap<<"\t\t"
            <<T->Key.TenSanPham<<"\t\t"
            <<T->Key.SoLuong<<"\t\t"
            <<T->Key.DonGia<<"\t\t"
            <<T->Key.DVT<<endl;
        }
    }
}

```

```

        i=true;
    }
}
}

```

### Hiện thị ra màn hình

```

So luong tren x:240
NhaCungCap10      TenSanPham10      550      100.75      lit
NhaCungCap9       TenSanPham9       500      90.5       kg
NhaCungCap8       TenSanPham8       450      80.25      lit
NhaCungCap7       TenSanPham7       400      70        kg
NhaCungCap6       TenSanPham6       350      60.75      lit
NhaCungCap5       TenSanPham5       300      50.5       kg
NhaCungCap4       TenSanPham4       250      40.25      lit

<-----Cac Chuc Nang----->
1. Khoi tao danh sach san pham
2. Kiem tra danh sach
3. Them san pham vao danh sach
4. In danh sach san pham
5. Doc thong tin cac san pham tu file vao danh sach
6. Ghi du lieu tu danh sach vao file
7. In danh sach san pham theo chieu tang gia
8. In danh sach san pham theo chieu giam gia
9. In ra tong gia tri cua moi loai san pham
10. Tinh va in ra so loai mat hang trong danh sach
11. Tinh va in ra tong so luong cua tat ca cac loai san pham
12. Tim va in ra thong tin cua san pham co ten x
13. Tim va in ra thong tin cua nhung loai san pham thuoc hang x
14. Tim va in ra thong tin nhung loai san pham co don vi tinh khoi luong la x
15. In ra thong tin nhung loai san pham co so luong tren x
16. In ra thong tin nhung loai san pham co so luong duoi x
17. In ra thong tin san pham co gia cao nhat
18. In ra thong tin san pham co gia thap nhat
19. In ra thong tin nhung san pham co gia tren muc x
20. In ra thong tin nhung san pham co gia duoi muc x
21. Tong so luong nhung loai san pham co gia tren muc x
22. Tong so luong nhung loai san pham co gia duoi muc x
23. Xoa nhung loai san pham co gia duoi muc x
24. Xoa nhung loai san pham co gia tren muc x
25. Tinh so luong nha cung cap

```

### 3.17 In ra thông tin của những loại sản phẩm có số lượng dưới x

#### Code

```

void SpSoLuongDuoix(Tree T, int x, bool &kthay)
{
    if (T != NULL)
    {
        SpSoLuongDuoix(T->Left, x, kthay);
        if (T->Key.SoLuong < x)

```

```

    {
        cout << T->Key.NhaCungCap << "\t\t"
        << T->Key.TenSanPham << "\t\t"
        << T->Key.SoLuong << "\t\t"
        << T->Key.DonGia << "\t\t" << T->Key.DVT << endl;
        kthay = true;
    }
    SpSoLuongDuoix(T->Right, x, kthay);
}
}

```

### Hiện thị ra màn hình

```

Nhap san pham co so luong duoi x: 300
NhaCungCap1      TenSanPham1      100      10.5      kg
NhaCungCap2      TenSanPham2      150      20.75     lit
NhaCungCap3      TenSanPham3      200      30        kg
NhaCungCap4      TenSanPham4      250      40.25     lit

<-----Cac Chuc Nang----->
1. Khoi tao danh sach san pham
2. Kiem tra danh sach
3. Them san pham vao danh sach
4. In danh sach san pham
5. Doc thong tin cac san pham tu file vao danh sach
6. Ghi du lieu tu danh sach vao file
7. In danh sach san pham theo chieu tang gia
8. In danh sach san pham theo chieu giam gia
9. In ra tong gia tri cua moi loai san pham
10. Tinh va in ra so loai mat hang trong danh sach
11. Tinh va in ra tong so luong cua tat ca cac loai san pham
12. Tim va in ra thong tin cua san pham co ten x
13. Tim va in ra thong tin cua nhung loai san pham thuoc hang x
14. Tim va in ra thong tin nhung loai san pham co don vi tinh khoi luong la x
15. In ra thong tin nhung loai san pham co so luong tren x
16. In ra thong tin nhung loai san pham co so luong duoi x

```

### 3.18 In ra thông tin của sản phẩm có giá cao nhất

#### Code

```

Item Max_value(Tree T){
    if (T->Right == NULL)
    {
        return T->Key;
    }
}

```



```

    }
    else
    {
        return Max_value(T->Right);
    }
}

```

### Hiện thị ra màn hình

```

Loai san pham co gia cao nhat:
NhaCungCap10      TenSanPham10      550      100.75      lit
<-----Cac Chuc Nang----->
1. Khoi tao danh sach san pham
2. Kiem tra danh sach
3. Them san pham vao danh sach
4. In danh sach san pham
5. Doc thong tin cac san pham tu file vao danh sach
6. Ghi du lieu tu danh sach vao file
7. In danh sach san pham theo chieu tang gia
8. In danh sach san pham theo chieu giam gia
9. In ra tong gia tri cua moi loai san pham
10. Tinh va in ra so loai mat hang trong danh sach
11. Tinh va in ra tong so luong cua tat ca cac loai san pham
12. Tim va in ra thong tin cua san pham co ten x
13. Tim va in ra thong tin cua nhung loai san pham thuoc hang x
14. Tim va in ra thong tin nhung loai san pham co don vi tinh khioi luong la x
15. In ra thong tin nhung loai san pham co so luong tren x
16. In ra thong tin nhung loai san pham co so luong duoi x
17. In ra thong tin san pham co gia cao nhat
18. In ra thong tin san pham co gia thap nhat
19. In ra thong tin nhung san pham co gia tren muc x
20. In ra thong tin nhung san pham co gia duoi muc x
21. Tong so luong nhung loai san pham co gia tren muc x
22. Tong so luong nhung loai san pham co gia duoi muc x
23. Xoa nhung loai san pham co gia duoi muc x
24. Xoa nhung loai san pham co gia tren muc x
25. Tinh so luong nha cung cap
26. Thong ke so cac loai san pham theo tung nha cung cap
27. Xoa san pham co gia cao nhat
28. Xoa san pham co gia thap nhat
29. Xoa san pham co gia bang x
30. Thoat khoi chuong trinh

```

### 3.19 In ra thông tin của sản phẩm có giá thấp nhất

#### Code

```

Item Min_value(Tree T)
{
    if (T->Left == NULL)
    {
        return T->Key;
    }
    else

```



```

{
    return Min_value(T->Left);
}
}

```

### Hiện thị ra màn hình

```

Loai san pham co gia thap nhat:
NhaCungCap1      TenSanPham1      100      10.5      kg

<-----Cac Chuc Nang----->
1. Khoi tao danh sach san pham
2. Kiem tra danh sach
3. Them san pham vao danh sach
4. In danh sach san pham
5. Doc thong tin cac san pham tu file vao danh sach
6. Ghi du lieu tu danh sach vao file
7. In danh sach san pham theo chieu tang gia
8. In danh sach san pham theo chieu giam gia
9. In ra tong gia tri cua moi loai san pham
10. Tinh va in ra so loai mat hang trong danh sach
11. Tinh va in ra tong so luong cua tat ca cac loai san pham
12. Tim va in ra thong tin cua san pham co ten x
13. Tim va in ra thong tin cua nhung loai san pham thuoc hang x
14. Tim va in ra thong tin nhung loai san pham co don vi tinh khoi luong la x
15. In ra thong tin nhung loai san pham co so luong tren x
16. In ra thong tin nhung loai san pham co so luong duoi x
17. In ra thong tin san pham co gia cao nhat
18. In ra thong tin san pham co gia thap nhat
19. In ra thong tin nhung san pham co gia tren muc x
20. In ra thong tin nhung san pham co gia duoi muc x
21. Tong so luong nhung loai san pham co gia tren muc x
22. Tong so luong nhung loai san pham co gia duoi muc x
23. Xoa nhung loai san pham co gia duoi muc x
24. Xoa nhung loai san pham co gia tren muc x
25. Tinh so luong nha cung cap

```

### 3.20 In ra những sản phẩm có giá trên mức x

#### Code

```

void Price_up_x(Tree T, int x, bool& i){
    if(T!=NULL){
        if(x<T->Key.DonGia){
            cout<<T->Key.NhaCungCap<<"\t\t"
            <<T->Key.TenSanPham<<"\t\t"
            <<T->Key.SoLuong<<"\t\t"
            <<T->Key.DonGia<<"\t\t"
            <<T->Key.DVT<<endl;
            Price_up_x(T->Left,x,i);
        }
    }
}

```

```

        Price_up_x(T->Right,x,i);
        i=true;
    }
    else
    {
        Price_up_x(T->Right,x,i);
    }
}
}

```

### Hiện thị ra màn hình

```

Gia tren x:34
NhaCungCap4      TenSanPham4      250      40.25      lit
NhaCungCap5      TenSanPham5      300      50.5       kg
NhaCungCap6      TenSanPham6      350      60.75      lit
NhaCungCap7      TenSanPham7      400      70         kg
NhaCungCap8      TenSanPham8      450      80.25      lit
NhaCungCap9      TenSanPham9      500      90.5       kg
NhaCungCap10     TenSanPham10     550      100.75     lit

<-----Cac Chuc Nang----->
1. Khoi tao danh sach san pham
2. Kiem tra danh sach
3. Them san pham vao danh sach
4. In danh sach san pham
5. Doc thong tin cac san pham tu file vao danh sach
6. Ghi du lieu tu danh sach vao file
7. In danh sach san pham theo chieu tang gia
8. In danh sach san pham theo chieu giam gia
9. In ra tong gia tri cua moi loai san pham
10. Tinh va in ra so loai mat hang trong danh sach
11. Tinh va in ra tong so luong cua tat ca cac loai san pham
12. Tim va in ra thong tin cua san pham co ten x
13. Tim va in ra thong tin cua nhung loai san pham thuoc hang x
14. Tim va in ra thong tin nhung loai san pham co don vi tinh khoi luong la x
15. In ra thong tin nhung loai san pham co so luong tren x

```

### 3.21 In ra những sản phẩm có giá dưới mức x

#### Code

```

void InDuoi_x(Tree T, int x){
    if(T!=NULL){
        if(x>T->Key.DonGia){
            cout<<T->Key.NhaCungCap<<"\t\t"

```

```

    <<T->Key.TenSanPham<<"\t\t"
    <<T->Key.SoLuong<<"\t\t"
    <<T->Key.DonGia<<"\t\t"
    <<T->Key.DVT<<endl;
    InDuoai_x(T->Left,x);
    InDuoai_x(T->Right,x);
}
else
{
    InDuoai_x(T->Left,x);
}
}
}

```

### Hiện thị ra màn hình

```

Gia tren x:30
NhaCungCap1      TenSanPham1      100      10.5      kg
NhaCungCap2      TenSanPham2      150      20.75     lit

<-----Cac Chuc Nang----->
1. Khoi tao danh sach san pham
2. Kiem tra danh sach
3. Them san pham vao danh sach
4. In danh sach san pham
5. Doc thong tin cac san pham tu file vao danh sach
6. Ghi du lieu tu danh sach vao file
7. In danh sach san pham theo chieu tang gia
8. In danh sach san pham theo chieu giam gia
9. In ra tong gia tri cua moi loai san pham
10. Tinh va in ra so loai mat hang trong danh sach
11. Tinh va in ra tong so luong cua tat ca cac loai san pham
12. Tim va in ra thong tin cua san pham co ten x
13. Tim va in ra thong tin cua nhung loai san pham thuoc hang x
14. Tim va in ra thong tin nhung loai san pham co don vi tinh khi loai luong la x
15. In ra thong tin nhung loai san pham co so luong tren x
16. In ra thong tin nhung loai san pham co so luong duoi x
17. In ra thong tin san pham co gia cao nhat
18. In ra thong tin san pham co gia thap nhat
19. In ra thong tin nhung san pham co gia tren muc x
20. In ra thong tin nhung san pham co gia duoi muc x
21. Tong so luong nhung loai san pham co gia tren muc x
22. Tong so luong nhung loai san pham co gia duoi muc x
23. Xoa nhung loai san pham co gia duoi muc x
24. Xoa nhung loai san pham co gia tren muc x

```

### 3.22 Tính tổng số lượng của những loại sản phẩm có giá trên mức x

**Code**

```

int Sl_PriceX(Tree T, float x){
    if(T!=NULL){
        if(x<T->Key.DonGia){
            return T->Key.Soluong + Sl_PriceX(T->Left,x) +
Sl_PriceX(T->Right,x);
        }
        else
        {
            return Sl_PriceX(T->Right,x);
        }
    }
}

```

**Hiện thị ra màn hình**

```

Nhap muc gia: 23
Tong Sl la: 3000

<-----Cac Chuc Nang----->
1. Khoi tao danh sach san pham
2. Kiem tra danh sach
3. Them san pham vao danh sach
4. In danh sach san pham
5. Doc thong tin cac san pham tu file vao danh sach
6. Ghi du lieu tu danh sach vao file
7. In danh sach san pham theo chieu tang gia
8. In danh sach san pham theo chieu giam gia
9. In ra tong gia tri cua moi loai san pham
10. Tinh va in ra so loai mat hang trong danh sach
11. Tinh va in ra tong so luong cua tat ca cac loai san pham
12. Tim va in ra thong tin cua san pham co ten x
13. Tim va in ra thong tin cua nhung loai san pham thuoc hang x
14. Tim va in ra thong tin nhung loai san pham co don vi tinh khoi luong la x
15. In ra thong tin nhung loai san pham co so luong tren x
16. In ra thong tin nhung loai san pham co so luong duoi x
17. In ra thong tin san pham co gia cao nhat
18. In ra thong tin san pham co gia thap nhat
19. In ra thong tin nhung san pham co gia tren muc x
20. In ra thong tin nhung san pham co gia duoi muc x
21. Tong so luong nhung loai san pham co gia tren muc x
22. Tong so luong nhung loai san pham co gia duoi muc x

```

**3.23 Tính tổng số lượng của những loại sản phẩm có giá dưới mức x****Code**

```

int Total_SL_downX(Tree T,float x){
    if(T!=NULL){
        if(x>T->Key.DonGia){
            return T->Key.SoLuong + Total_SL_downX(T->Left,x) +
Total_SL_downX(T->Right,x);
        }
        else
        {
            return Total_SL_downX(T->Left,x);
        }
    }
}

```

#### Hiện thị ra màn hình

```

Nhap muc gia: 50
Tong SL la: 700

<-----Cac Chuc Nang----->
1. Khoi tao danh sach san pham

```

### 3.24 Xóa sản phẩm có giá dưới mức x khỏi danh sách

- Nhập giá muốn xóa tất cả giá dưới mức x trong danh sách cây nhị phân

#### Code

```

TNode* TimKiem(TNode* T) // tim node nho nhat
{
    while (T->Left != NULL)
        T = T->Left;
    return T;
}

TNode* XoaNode(Tree &T, float DonGia)
{

```

```

    if (T == NULL)
        return T;
    if (DonGia < T->Key.DonGia)
    {
        T->Left = XoaNode(T->Left, DonGia);
    }
    else if (DonGia > T->Key.DonGia)
    {
        T->Right = XoaNode(T->Right, DonGia);
    }
    else
    {
        if (T->Left == NULL)
        {
            TNode* temp = T->Right;
            delete T;
            return temp;
        }
        else if (T->Right == NULL)
        {
            TNode* temp = T->Left;
            delete T;
            return temp;
        }
        TNode* temp = TimKiem(T->Right);
        T->Key = temp->Key;
        T->Right = XoaNode(T->Right, temp->Key.DonGia);
    }
    return T;
}

void XoaSpGiaDuoix(Tree &T, float x)

```

```

{
    if (T == NULL)
        return;
    XoaSpGiaDuoix(T->Left, x);
    XoaSpGiaDuoix(T->Right, x);
    if (T != NULL && T->Key.DonGia < x)
    {
        T = XoaNode(T, T->Key.DonGia);
        XoaSpGiaDuoix(T, x);
    }
}

```

### Hiện thị ra màn hình

```

Nhap muc gia it hon de xoa: 23
Da xoa thanh cong!

<-----Cac Chuc Nang----->
1. Khoi tao danh sach san pham
2. Kiem tra danh sach
3. Them san pham vao danh sach
4. In danh sach san pham
5. Doc thong tin cac san pham tu file vao danh sach
6. Ghi du lieu tu danh sach vao file
7. In danh sach san pham theo chieu tang gia
8. In danh sach san pham theo chieu giam gia
9. In ra tong gia tri cua moi loai san pham
10. Tinh va in ra so loai mat hang trong danh sach
11. Tinh va in ra tong so luong cua tat ca cac loai san pham
12. Tim va in ra thong tin cua san pham co ten x
13. Tim va in ra thong tin cua nhung loai san pham thuoc hang x
14. Tim va in ra thong tin nhung loai san pham co don vi tinh khoi luong la x
15. In ra thong tin nhung loai san pham co so luong tren x

```

### 3.25 Xóa sản phẩm có giá trên x khỏi danh sách

#### Code

```

void DeletePrice(Tree &T, float x){
    if(T!=NULL){
        if(T->Key.DonGia>x){
            DeletePrice(T->Left,x);
            DeletePrice(T->Right,x);
            Tree Node=T;

```

```

        if(T->Left==NULL&&T->Right==NULL){
            T=NULL;
        }else if(T->Left==NULL){
            T=T->Right;
        }else if(T->Right==NULL){
            T=T->Left;
        }else{
            Tree PNode=T;
            Tree CNode=T->Left;
            while (CNode!=NULL)
            {
                T=CNode;
                CNode=CNode->Left;
            }
            if(PNode->Left==CNode){
                PNode->Left=CNode->Right;
            }else{
                PNode->Right=CNode->Right;
            }
            Node=CNode;
        }
        delete Node;
    }
    else
    {
        DeletePrice(T->Right,x);
    }
}
}

```

**Hiện thị ra màn hình**



```

Delete Gia tren X: 50
Da xoa xog!
<-----Cac Chuc Nang----->
1. Khoi tao danh sach san pham
2. Kiem tra danh sach
3. Them san pham vao danh sach
4. In danh sach san pham
5. Doc thong tin cac san pham tu file vao danh sach
6. Ghi du lieu tu danh sach vao file
7. In danh sach san pham theo chieu tang gia
8. In danh sach san pham theo chieu giam gia
9. In ra tong gia tri cua moi loai san pham
10. Tinh va in ra so loai mat hang trong danh sach
11. Tinh va in ra tong so luong cua tat ca cac loai san pham
12. Tim va in ra thong tin cua san pham co ten x
13. Tim va in ra thong tin cua nhung loai san pham thuoc hang x
14. Tim va in ra thong tin nhung loai san pham co don vi tinh khioi luong la x
15. In ra thong tin nhung loai san pham co so luong tren x
16. In ra thong tin nhung loai san pham co so luong duoi x
17. In ra thong tin san pham co gia cao nhat
18. In ra thong tin san pham co gia thap nhat
19. In ra thong tin nhung san pham co gia tren muc x
20. In ra thong tin nhung san pham co gia duoi muc x
21. Tong so luong nhung loai san pham co gia tren muc x
22. Tong so luong nhung loai san pham co gia duoi muc x
23. Xoa nhung loai san pham co gia duoi muc x

```

### 3.26 Tính số lượng nhà cung cấp trong danh sách

#### Code

```

int Sl_NCC(Tree T, string name[]){
    if (!isEmpty(T))
    {
        int i=0;
        if (name[0]=="")
        {
            name[0]=T->Key.NhaCungCap;
        }
        else
        {
            while (name[i]!="")
            {
                if (name[i]==T->Key.NhaCungCap)
                {
                    break;
                }
            }
        }
    }
}

```

```

    }
    else{
        if (name[i+1]=="")
        {
            name[i+1]=T->Key.NhaCungCap;
            break;
        }
        i++;
    }
}

}
Sl_NCC(T->Left,name);
Sl_NCC(T->Right,name);
}
}

```

### Hiện thị ra màn hình

```

So luong nha cung cap trong danh sach la: 2

<-----Cac Chuc Nang----->
1. Khoi tao danh sach san pham
2. Kiem tra danh sach
3. Them san pham vao danh sach
4. In danh sach san pham
5. Doc thong tin cac san pham tu file vao danh sach
6. Ghi du lieu tu danh sach vao file
7. In danh sach san pham theo chieu tang gia
8. In danh sach san pham theo chieu giam gia
9. In ra tong gia tri cua moi loai san pham
10. Tinh va in ra so loai mat hang trong danh sach
11. Tinh va in ra tong so luong cua tat ca cac loai san pham
12. Tim va in ra thong tin cua san pham co ten x
13. Tim va in ra thong tin cua nhung loai san pham thuoc hang x
14. Tim va in ra thong tin nhung loai san pham co don vi tinh khoi luong la x
15. In ra thong tin nhung loai san pham co so luong tren x
16. In ra thong tin nhung loai san pham co so luong duoi x
17. In ra thong tin san pham co gia cao nhat
18. In ra thong tin san pham co gia thap nhat
19. In ra thong tin nhung san pham co gia tren muc x
20. In ra thong tin nhung san pham co gia duoi muc x
21. Tong so luong nhung loai san pham co gia tren muc x
22. Tong so luong nhung loai san pham co gia duoi muc x
23. Xoa nhung loai san pham co gia duoi muc x
24. Xoa nhung loai san pham co gia tren muc x
25. Tinh so luong nha cung cap
26. Thong ke so cac loai san pham theo tung nha cung cap
27. Xoa san pham co gia cao nhat

```

### 3.27 Thống kê các loại sản phẩm theo nhà cung cấp

#### Code

```
int Sl_NCC(Tree T, string name[]){
    if (!isEmpty(T))
    {
        int i=0;
        if (name[0]=="")
        {
            name[0]=T->Key.NhaCungCap;
        }
        else
        {
            while (name[i]!="")
            {
                if (name[i]==T->Key.NhaCungCap)
                {
                    break;
                }
                else{
                    if (name[i+1]=="")
                    {
                        name[i+1]=T->Key.NhaCungCap;
                        break;
                    }
                    i++;
                }
            }
        }

        Sl_NCC(T->Left,name);
    }
}
```

```

        Sl_NCC(T->Right,name);
    }

}

void TimSanPhamHangX1(Tree T, string x)
{
    if (T != NULL)
    {
        TimSanPhamHangX1(T->Left,x);
        if (T->Key.NhaCungCap == x)
        {
            cout << "\t\t" << T->Key.TenSanPham << "\t\t" << T-
>Key.SoLuong << "\t\t" << T->Key.DonGia << "\t\t" << T->Key.DVT <<
endl;
        }
        TimSanPhamHangX1(T->Right,x);
    }
}

```

### Hiện thị ra màn hình

```

Phan phoi cac loai san pham theo nha cung cap:
NhaCungCap5      TenSanPham5      300      50.5      kg
NhaCungCap6      TenSanPham6      350      60.75     lit
NhaCungCap7      TenSanPham7      400      70        kg
NhaCungCap8      TenSanPham8      450      80.25     lit
NhaCungCap9      TenSanPham9      500      90.5      kg

<-----Cac Chuc Nang----->
1. Khoi tao danh sach san pham
2. Kiem tra danh sach
3. Them san pham vao danh sach
4. In danh sach san pham
5. Doc thong tin cac san pham tu file vao danh sach
6. Ghi du lieu tu danh sach vao file
7. In danh sach san pham theo chieu tang gia
8. In danh sach san pham theo chieu giam gia
9. In ra tong gia tri cua moi loai san pham
10. Tinh va in ra so loai mat hang trong danh sach
11. Tinh va in ra tong so luong cua tat ca cac loai san pham
12. Tim va in ra thong tin cua san pham co ten x
13. Tim va in ra thong tin cua nhung loai san pham thuoc hang x
14. Tim va in ra thong tin nhung loai san pham co don vi tinh khoi luong la x
15. In ra thong tin nhung loai san pham co so luong tren x
16. In ra thong tin nhung loai san pham co so luong duoi x

```

### 3.28 Xoá sản phẩm có giá cao nhất

#### Code

```
void DeletePriceTall(Tree& T){

    if(T->Right!=NULL){
        DeletePriceTall(T->Right);
    }else{
        Tree NewNode=T;
        T=T->Left;
        delete NewNode;
    }
}
```

#### Hiện thị ra màn hình

```
Da xoa!
<-----Cac Chuc Nang----->
1. Khoi tao danh sach san pham
2. Kiem tra danh sach
3. Them san pham vao danh sach
4. In danh sach san pham
5. Doc thong tin cac san pham tu file vao danh sach
6. Ghi du lieu tu danh sach vao file
7. In danh sach san pham theo chieu tang gia
8. In danh sach san pham theo chieu giam gia
9. In ra tong gia tri cua moi loai san pham
10. Tinh va in ra so loai mat hang trong danh sach
11. Tinh va in ra tong so luong cua tat ca cac loai san pham
12. Tim va in ra thong tin cua san pham co ten x
13. Tim va in ra thong tin cua nhung loai san pham thuoc hang x
14. Tim va in ra thong tin nhung loai san pham co don vi tinh khoi luong la x
15. In ra thong tin nhung loai san pham co so luong tren x
16. In ra thong tin nhung loai san pham co so luong duoi x
```

### 3.29 Xoá sản phẩm có giá thấp nhất

#### Code

```
void XoaSpGiaMin(Tree& T){
    if(T!=NULL){
        Item MinDV=Min_value(T);
```

```

        T=XoaNode(T, MinDV.DonGia);
    }

}

```

### Hiện thị ra màn hình

```

Da xoa!
<-----Cac Chuc Nang----->
1. Khoi tao danh sach san pham
2. Kiem tra danh sach
3. Them san pham vao danh sach
4. In danh sach san pham
5. Doc thong tin cac san pham tu file vao danh sach
6. Ghi du lieu tu danh sach vao file
7. In danh sach san pham theo chieu tang gia
8. In danh sach san pham theo chieu giam gia
9. In ra tong gia tri cua moi loai san pham
10. Tinh va in ra so loai mat hang trong danh sach
11. Tinh va in ra tong so luong cua tat ca cac loai san pham
12. Tim va in ra thong tin cua san pham co ten x
13. Tim va in ra thong tin cua nhung loai san pham thuoc hang x
14. Tim va in ra thong tin nhung loai san pham co don vi tinh khoi luong la x
15. In ra thong tin nhung loai san pham co so luong tren x
16. In ra thong tin nhung loai san pham co so luong duoi x

```

### 3.30 Xoá sản phẩm có giá bằng x

- Nhập giá bằng x:

#### Code

```

void NodeTheMang(Tree &X, Tree &Y)
{
    if (Y->Left != NULL)
    {
        NodeTheMang(X, Y->Left);
    }
    else
    {
        X->Key = Y->Key;
        X = Y;
        Y = Y->Right;
    }
}

```

```

    }
}
void XoaNodex(Tree &T, float x)
{
    if (T == NULL)
    {
        return;
    }
    else
    {
        if (x < T->Key.DonGia)
        {
            XoaNodex(T->Left, x);
        }
        else if (x > T->Key.DonGia)
        {
            XoaNodex(T->Right, x);
        }
        else
        {
            TNode *X = T;
            if (T->Left == NULL)
            {
                T=T->Right;
            }
            else if (T->Right == NULL)
            {
                T=T->Left;
            }
            else
            {

```

```

        NodeTheMang(X, T->Right);
    void NodeTheMang(Tree &X, Tree &Y)
{
    if (Y->Left != NULL)
    {
        NodeTheMang(X, Y->Left);
    }
    else
    {
        X->Key = Y->Key;
        X = Y;
        Y = Y->Right;
    }
}

void XoaNodeX(Tree &T, float x)
{
    if (T == NULL)
    {
        return;
    }
    else
    {
        if (x < T->Key.DonGia)
        {
            XoaNodeX(T->Left, x);
        }
        else if (x > T->Key.DonGia)
        {
            XoaNodeX(T->Right, x);
        }
        else
        {
            TNode *X = T;
            if (T->Left == NULL)
            {
                T=T->Right;
            }
            else if (T->Right == NULL)
            {
                T=T->Left;
            }
            else

```



```

        {
            NodeTheMang(X, T->Right);
        }
        delete X;
    }
}
}
bool kiemtragia(Tree T, float x){
    if(T!=NULL)
    {
        if (T->Key.DonGia==x)
        {
            return true;
        }
        else if (T->Key.DonGia<x)
        {
            kiemtragia(T->Right,x);
        }
        else
        {
            kiemtragia(T->Right,x);
        }
    }
}
}

```

### Hiện thị ra màn hình

```

Nhap gia san pham can xoa: 70
Da xoa!
<-----Cac Chuc Nang-----
1. Khoi tao danh sach san pham
2. Kiem tra danh sach
3. Them san pham vao danh sach
4. In danh sach san pham
5. Doc thong tin cac san pham tu file vao danh sach
6. Ghi du lieu tu danh sach vao file
7. In danh sach san pham theo chieu tang gia
8. In danh sach san pham theo chieu giam gia
9. In ra tong gia tri cua moi loai san pham
10. Tinh gia tri trung binh cua moi loai san pham

```

### 3.31 Thoát khỏi chương trình

- Nhập chức năng 30 chương trình sẽ tự tắt và thoát khỏi chương trình

#### Code

```
case 30:
    {
        cout<<"Tam biet!";
        break;
    }
}
```

#### Hiện thị ra màn hình

```
Tam biet!
PS D:\CODE\DNU_CPP\KIEN_TRUC_DU_LIEU_VS_GIAI_THUAT\BTL> |
```

## CHƯƠNG 4. MÃ NGUỒN CHƯƠNG TRÌNH

*Toàn bộ code của chương trình sử dụng cấu trúc cây nhị phân.*

```
#include<iostream>
#include<fstream>
#include<iomanip>
#include<string>
#include<cstdlib>
using namespace std;
struct SanPham
{
    string NhaCungCap;
    string TenSanPham;
    int SoLuong;
    float DonGia;
    string DVT;
};
typedef SanPham Item;
struct TNode
{
    Item Key;
    TNode *Left;
    TNode *Right;
};
typedef TNode *Tree;
//1
void Init(Tree &T)
{
    T=NULL;
}
//2
bool isEmpty(Tree T)
{
    if (T==NULL)
    {
        return true;
    }
    else
    {
        return false;
    }
}
```

```

//3
void AddNode(Tree &T, Item x)
{
    if (T==NULL)
    {
        T= new TNode;
        T->Key=x;
        T->Left=NULL;
        T->Right=NULL;
    }
    else
    {
        if (T->Key.DonGia>x.DonGia)
        {
            AddNode(T->Left,x);
        }
        else
        {
            AddNode(T->Right,x);
        }
    }
}

//4
void InDanhSach(Tree T)
{
    if (T!=NULL)
    {
        cout<<T->Key.NhaCungCap<<"\t\t"<<T->Key.TenSanPham<<"\t\t"<<T->Key.Soluong<<"\t\t"<<T->Key.DonGia<<"\t\t"<<T->Key.DVT<<endl;
        InDanhSach(T->Left);
        InDanhSach(T->Right);
    }
}

//5
void ReadFileToDSSP(Tree &T)
{
    ifstream f1;
    f1.open("SanPham.txt");
    if (!f1)
    {
        cout<<"Mo tep loi!\n";
    }
    else

```

```

{
    Item x;
    while (!f1.eof())
    {
        f1.ignore();
        getline(f1,x.NhaCungCap);
        getline(f1,x.TenSanPham);
        f1>>x.Soluong;
        f1>>x.DonGia;
        f1>>x.DVT;
        AddNode(T,x);
    }
    cout<<"Da doc xong!\n";
}
f1.close();
}
//6 - Nguyễn Thanh Tùng
void WriteToFile(ofstream &file,Tree T){
    if(T!=NULL){
        file<<T->Key.NhaCungCap<<"\n"<<T->Key.TenSanPham<<"\n"<<T->Key.Soluong<<"\n"<<T->Key.DonGia<<"\n"<<T->Key.DVT<<endl;
        WriteToFile(file,T->Left);
        WriteToFile(file,T->Right);
    }
}
void WriteFileDSSP(Tree T){
    //fstream file("Sp.txt",ios::out); /// au tao tao ra file de chua no.
    ofstream file;
    file.open("Sp.txt");
    if(!file){
        cout<<"File khong ton tai! ";
    }else{
        file<<"\n";
        WriteToFile(file,T);
        file.close();
        cout<<"Da ghi xog\n";
    }
}
// 7
void LNR(Tree T)
{
    if (T!=NULL)
    {

```

```

        LNR(T->Left);
        cout<<T->Key.NhaCungCap<<"\t\t"
        <<T->Key.TenSanPham<<"\t\t"
        <<T->Key.Soluong<<"\t\t"<<T->Key.DonGia<<"\t\t"<<T->Key.DVT<<endl;
        LNR(T->Right);
    }
}
// 8
void RNL(Tree T)
{
    if (T!=NULL)
    {
        RNL(T->Right);
        cout<<T->Key.NhaCungCap<<"\t\t"<<T->Key.TenSanPham<<"\t\t"<<T->
        >Key.Soluong<<"\t\t"<<T->Key.DonGia<<"\t\t"<<T->Key.DVT<<endl;
        RNL(T->Left);
    }
}
// 9
void InGiaTri(Tree T)
{
    if (T!=NULL)
    {
        cout<<T->Key.NhaCungCap<<"\t\t"<<T->Key.TenSanPham<<"\t\t"<<T->
        >Key.Soluong<<"\t\t"<<T->Key.DonGia<<"\t\t"<<T->Key.DonGia*T->Key.Soluong<<endl;
        InGiaTri(T->Left);
        InGiaTri(T->Right);
    }
}
// 10
int DemSoSanPham(Tree T)
{
    if (T != NULL)
    {
        return 1 + DemSoSanPham(T->Left) + DemSoSanPham(T->Right);
    }
    else
    {
        return 0;
    }
}
// 11
float Total_Sp(Tree T){

```

```

        if(T!=NULL){
            return(T->Key.Soluong +Total_Sp(T->Left) +Total_Sp(T->Right));
        }else{
            return 0;
        }
    }
}
// 12
void TimSanPham(Tree T, string x, bool &kthay)
{
    if (T != NULL)
    {
        TimSanPham(T->Left, x, kthay);
        if (T->Key.TenSanPham == x)
        {
            cout << T->Key.NhaCungCap << "\t\t" << T->Key.TenSanPham << "\t\t" << T->Key.Soluong << "\t\t" << T->Key.DonGia << "\t\t" << T->Key.DVT << endl;
            kthay = true;
        }
        TimSanPham(T->Right, x, kthay);
    }
}

// 13
void TimSanPhamHangX(Tree T, string x, bool &kthay)
{
    if (T != NULL)
    {
        TimSanPhamHangX(T->Left, x, kthay);
        if (T->Key.NhaCungCap == x)
        {
            cout << T->Key.NhaCungCap << "\t\t" << T->Key.TenSanPham << "\t\t" << T->Key.Soluong << "\t\t" << T->Key.DonGia << "\t\t" << T->Key.DVT << endl;
            kthay = true;
        }
        TimSanPhamHangX(T->Right, x, kthay);
    }
}

// 14
void TimSanPhamDVT(Tree T, string x, bool &kthay)
{
    if (T != NULL)
    {
        TimSanPhamDVT(T->Left, x, kthay);
    }
}

```

```

        if (T->Key.DVT == x)
        {
            cout << T->Key.NhaCungCap << "\t\t" << T->Key.TenSanPham << "\t\t" << T->Key.Soluong << "\t\t" << T->Key.DonGia << "\t\t" << T->Key.DVT << endl;
            kthay = true;
        }
        TimSanPhamDVT(T->Right, x, kthay);
    }
}

// 15
void Sl_up_x(Tree T, int x, bool& i){
    if(T!=NULL){
        Sl_up_x(T->Left, x, i);
        Sl_up_x(T->Right, x, i);
        if(x<T->Key.Soluong){
            cout<<T->Key.NhaCungCap<<"\t\t"<<T->Key.TenSanPham<<"\t\t"<<T->Key.Soluong<<"\t\t"<<T->Key.DonGia<<"\t\t"<<T->Key.DVT<<endl;
            // cout<<"sl: "<<T->Key.Soluong<<endl;
            i=true;
        }
    }
}

// 16
void SpSoluongDuoix(Tree T, int x, bool &kthay)
{
    if (T != NULL)
    {
        SpSoluongDuoix(T->Left, x, kthay);
        if (T->Key.Soluong < x)
        {
            cout << T->Key.NhaCungCap << "\t\t" << T->Key.TenSanPham << "\t\t" << T->Key.Soluong << "\t\t" << T->Key.DonGia << "\t\t" << T->Key.DVT << endl;
            kthay = true;
        }
        SpSoluongDuoix(T->Right, x, kthay);
    }
}

// 17
Item Max_value(Tree T){
    if (T->Right == NULL)
    {
        return T->Key;
    }
}

```



```

    }
    else
    {
        return Max_value(T->Right);
    }
}
// 18
Item Min_value(Tree T)
{
    if (T->Left == NULL)
    {
        return T->Key;
    }
    else
    {
        return Min_value(T->Left);
    }
}
// 19
void Price_up_x(Tree T, int x, bool& i){
    if(T!=NULL){
        if(x<T->Key.DonGia){
            cout<<T->Key.NhaCungCap<<"\t\t"<<T->Key.TenSanPham<<"\t\t"<<T->Key.Soluong<<"\t\t"<<T->Key.DonGia<<"\t\t"<<T->Key.DVT<<endl;
            Price_up_x(T->Left,x,i);
            Price_up_x(T->Right,x,i);
            i=true;
        }
        else
        {
            Price_up_x(T->Right,x,i);
        }
    }
}
// 20
void InDuoix(Tree T, int x){
    if(T!=NULL){
        if(x>T->Key.DonGia){
            cout<<T->Key.NhaCungCap<<"\t\t"<<T->Key.TenSanPham<<"\t\t"<<T->Key.Soluong<<"\t\t"<<T->Key.DonGia<<"\t\t"<<T->Key.DVT<<endl;
            InDuoix(T->Left,x);
            InDuoix(T->Right,x);
        }
    }
}

```

```

    }
    else
    {
        InDuoai_x(T->Left,x);
    }
}

// 21
int Sl_PriceX(Tree T, float x){
    if(T!=NULL){
        if(x<T->Key.DonGia){
            return T->Key.Soluong + Sl_PriceX(T->Left,x) + Sl_PriceX(T->Right,x);
        }
        else
        {
            return Sl_PriceX(T->Right,x);
        }
    }
}

// 22
int Total_SL_downX(Tree T,float x){
    if(T!=NULL){
        if(x>T->Key.DonGia){
            return T->Key.Soluong + Total_SL_downX(T->Left,x) + Total_SL_downX(T->Right,x);
        }
        else
        {
            return Total_SL_downX(T->Left,x);
        }
    }
}

// 23
TNode* TimKiem(TNode* T) // tim node nho nhat
{
    while (T->Left != NULL)
        T = T->Left;
    return T;
}

```

```

TNode* XoaNode(Tree &T, float DonGia)           // dung de xoa node k bi loi. case
12,13
{
    if (T == NULL)
        return T;
    if (DonGia < T->Key.DonGia)
    {
        T->Left = XoaNode(T->Left, DonGia);
    }
    else if (DonGia > T->Key.DonGia)
    {
        T->Right = XoaNode(T->Right, DonGia);
    }
    else
    {
        if (T->Left == NULL)
        {
            TNode* temp = T->Right;
            delete T;
            return temp;
        }
        else if (T->Right == NULL)
        {
            TNode* temp = T->Left;
            delete T;
            return temp;
        }
        TNode* temp = TimKiem(T->Right);
        T->Key = temp->Key;
        T->Right = XoaNode(T->Right, temp->Key.DonGia);
    }
    return T;
}

void XoaSpGiaDuoix(Tree &T, float x)
{
    if (T == NULL)
        return;
    XoaSpGiaDuoix(T->Left, x);
    XoaSpGiaDuoix(T->Right, x);
    if (T != NULL && T->Key.DonGia < x)
    {
        T = XoaNode(T, T->Key.DonGia);
        XoaSpGiaDuoix(T, x);
    }
}

```

```

    }
}
// 24
void DeletePrice(Tree &T, float x){
    if(T!=NULL){
        if(T->Key.DonGia>x){
            DeletePrice(T->Left,x);
            DeletePrice(T->Right,x);
            Tree Node=T;
            if(T->Left==NULL&&T->Right==NULL){
                T=NULL;
            }else if(T->Left==NULL){
                T=T->Right;
            }else if(T->Right==NULL){
                T=T->Left;
            }else{
                Tree PNode=T;
                Tree CNode=T->Left;
                while (CNode!=NULL)
                {
                    T=CNode;
                    CNode=CNode->Left;
                }
                if(PNode->Left==CNode){
                    PNode->Left=CNode->Right;
                }else{
                    PNode->Right=CNode->Right;
                }
                Node=CNode;
            }
            delete Node;
        }
        else
        {
            DeletePrice(T->Right,x);
        }
    }
}
// 25-26
void Sl_NCC(Tree T, string name[]){
    if (!isEmpty(T))
    {

```

```

    int i=0;
    if (name[0]=="")
    {
        name[0]=T->Key.NhaCungCap;
    }
    else
    {
        while (name[i]!="")
        {
            if (name[i]==T->Key.NhaCungCap)
            {
                break;
            }
            else{
                if (name[i+1]=="")
                {
                    name[i+1]=T->Key.NhaCungCap;
                    break;
                }
                i++;
            }
        }

        }
    SL_NCC(T->Left,name);
    SL_NCC(T->Right,name);
}

}

void TimSanPhamHangX1(Tree T, string x)
{
    if (T != NULL)
    {
        TimSanPhamHangX1(T->Left,x);
        if (T->Key.NhaCungCap == x)
        {
            cout <<"\t\t" << T->Key.TenSanPham << "\t\t" << T->Key.SoLuong << "\t\t" <<
T->Key.DonGia << "\t\t" << T->Key.DVT << endl;
        }
        TimSanPhamHangX1(T->Right,x);
    }
}

// 27

```

```

void DeletePriceTall(Tree& T){

    if(T->Right!=NULL){
        DeletePriceTall(T->Right);
    }else{
        Tree NewNode=T;
        T=T->Left;
        delete NewNode;
    }
}
// 28
void XoaSpGiaMin(Tree& T){
    if(T!=NULL){
        Item MinDV=Min_value(T);
        T=XoaNode(T, MinDV.DonGia);
    }
}
//29
void NodeTheMang(Tree &X, Tree &Y)
{
    if (Y->Left != NULL)
    {
        NodeTheMang(X, Y->Left);
    }
    else
    {
        X->Key = Y->Key;
        X = Y;
        Y = Y->Right;
    }
}

void XoaNodex(Tree &T, float x)
{
    if (T == NULL)
    {
        return;
    }
    else
    {
        if (x < T->Key.DonGia)
        {
            XoaNodex(T->Left, x);

```

```

    }
    else if (x > T->Key.DonGia)
    {
        XoaNodeX(T->Right, x);
    }
    else
    {
        TNode *X = T;
        if (T->Left == NULL)
        {
            T=T->Right;
        }
        else if (T->Right == NULL)
        {
            T=T->Left;
        }
        else
        {
            NodeTheMang(X, T->Right);
        }
        delete X;
    }
}
}
bool kiemtragia(Tree T,float x){
    if(T!=NULL)
    {
        if (T->Key.DonGia==x)
        {
            return true;
        }
        else if (T->Key.DonGia<x)
        {
            kiemtragia(T->Right,x);
        }
        else
        {
            kiemtragia(T->Left,x);
        }
    }
}
// Menu

```

```

void Menu(Tree T)
{
    int LuaChon;
    do
    {
        cout<<"\n<-----Cac Chuc Nang----->\n";

        cout<<"1. Khoi tao danh sach san pham\n";
        cout<<"2. Kiem tra danh sach\n";
        cout<<"3. Them san pham vao danh sach\n";
        cout<<"4. In danh sach san pham\n";
        cout<<"5. Doc thong tin cac san pham tu file vao danh sach\n";
        cout<<"6. Ghi du lieu tu danh sach vao file\n";
        cout<<"7. In danh sach san pham theo chieu tang gia\n";
        cout<<"8. In danh sach san pham theo chieu giam gia\n";
        cout<<"9. In ra tong gia tri cua moi loai san pham\n";
        cout<<"10. Tinh va in ra so loai mat hang trong danh sach\n";
        cout<<"11. Tinh va in ra tong so luong cua tat ca cac loai san pham\n";
        cout<<"12. Tim va in ra thong tin cua san pham co ten x\n";
        cout<<"13. Tim va in ra thong tin cua nhung loai san pham thuoc hang x\n";
        cout<<"14. Tim va in ra thong tin nhung loai san pham co don vi tinh khoi luong
        la x\n";

        cout<<"15. In ra thong tin nhung loai san pham co so luong tren x\n";
        cout<<"16. In ra thong tin nhung loai san pham co so luong duoi x\n";
        cout<<"17. In ra thong tin san pham co gia cao nhat\n";
        cout<<"18. In ra thong tin san pham co gia thap nhat\n";
        cout<<"19. In ra thong tin nhung san pham co gia tren muc x\n";
        cout<<"20. In ra thong tin nhung san pham co gia duoi muc x\n";
        cout<<"21. Tong so luong nhung loai san pham co gia tren muc x\n";
        cout<<"22. Tong so luong nhung loai san pham co gia duoi muc x\n";
        cout<<"23. Xoa nhung loai san pham co gia duoi muc x\n";
        cout<<"24. Xoa nhung loai san pham co gia tren muc x\n";
        cout<<"25. Tinh so luong nha cung cap\n";
        cout<<"26. Thong ke so cac loai san pham theo tung nha cung cap\n";
        cout<<"27. Xoa san pham co gia cao nhat\n";
        cout<<"28. Xoa san pham co gia thap nhat\n";
        cout<<"29. Xoa san pham co gia bang x\n";
        cout<<"30. Thoat khoi chuong trinh\n";
        cout<<"Moi chon chuc nang (1-30)\n==> ";
        cin>>LuaChon;
        system("cls");
        switch(LuaChon)
        {

```



```

case 1:
{
    Init(T);
    cout<<"Da khoi tao danh sach san pham!\n";
    break;
}
case 2:
{
    if(isEmpty(T)){
        cout<<"Danh sach rong!\n";
    }
    else
    {
        cout<<"Danh sach khong rong!\n";
    }
    break;
}
case 3:
{
    Item x;
    cout<<"Nhap vao nhung thong tin cua san pham can them:\n";
    cout<<"Ten nha cung cap: ";
    cin.ignore();
    getline(cin,x.NhaCungCap);
    cout<<"Ten san pham: ";
    getline(cin,x.TenSanPham);
    cout<<"So luong: ";
    cin>>x.SoLuong;
    cout<<"Don gia: ";
    cin>>x.DonGia;
    cout<<"Don Vi: ";
    cin>>x.DVT;
    AddNode(T,x);
    break;
}
case 4:
{
    if (!isEmpty(T))
    {
        cout<<"\nNhaCC\t\t\t"<<"TenSp\t\t\t"<<"Sl\t\t"<<"Gia"<<"\t\t"<<"
DonVi"<<endl;

        InDanhSach(T);
    }
}

```

```

    }
    else
    {
        cout<<"Danh sach rong!\n";
    }

    break;
}
case 5:
{
    ReadFileToDSSP(T);
    break;
}
case 6: // doc tu danh sach vao file
{
    WriteFileDSSP(T);
    break;
}
case 7:
{
    if (!isEmpty(T))
    {
        cout<<"\nNhaCC\t\t\t"<<"TenSp\t\t\t"<<"Sl\t\t"<<"Gia"<<"\t\t"<<"
DonVi"<<endl;

        LNR(T);
    }
    else
    {
        cout<<"Danh sach rong!\n";
    }

    break;
}
case 8:
{
    if (!isEmpty(T))
    {
        cout<<"\nNhaCC\t\t\t"<<"TenSp\t\t\t"<<"Sl\t\t"<<"Gia"<<"\t\t"<<"
DonVi"<<endl;

        RNL(T);
    }
    else
    {

```

```

        cout<<"Danh sach rong!\n";
    }

    break;
}
case 9: // tinh tong gia sp
{
    cout<<"\nNhaCC\t\t\t"<<"TenSp\t\t\t"<<"Sl\t\t"<<"Gia"<<"\t\t"<<"TongG
ia"<<endl;

    InGiaTri(T);
    break;
}
case 10:
{
    cout << "Tong so luong san pham: " << DemSoSanPham(T) << endl;
    break;
}
case 11: // tinh tong sl sp
{
    cout<<"Tong Sl sp: "<<Total_Sp(T)<<endl;
    break;
}
case 12:
{
    string x;
    bool kthay = false;
    cout << "Nhap ten san pham can tim: ";
    cin.ignore();
    getline(cin, x);
    TimSanPham(T, x, kthay);
    if (!kthay)
    {
        cout << "Khong tim thay ten san pham!" << endl;
    }
    break;
}
case 13:
{
    string x;
    bool kthay = false;
    cout << "Nhap ten hang can tim: ";
    cin.ignore();
    getline(cin, x);

```

```

TimSanPhamHangX(T, x, kthay);
if (!kthay)
{
    cout << "Khong tim thay ten hang!" << endl;
}
break;
}
case 14:
{
    string x;
    bool kthay = false;
    cout << "Nhap san pham co khoi luong can tim: ";
    cin.ignore();
    getline(cin, x);
    TimSanPhamDVT(T, x, kthay);
    if (!kthay)
    {
        cout << "Khong tim thay ten san pham co khoi luong nhu vay!" << endl;
    }
    break;
}
case 15:// in ra so luong tren x
{
    int x;
    bool i=false;
    cout<<"So luong tren x:";cin>>x;
    Sl_up_x(T,x,i);
    if(!i){
        cout<<"Khong co sl nao tren: "<<x;
    }
    break;
}
case 16:
{
    int x;
    bool kthay = false;
    cout << "Nhap san pham co so luong duoi x: ";
    cin >> x;
    SpSoLuongDuoix(T, x, kthay);
    if (!kthay)
    {
        cout << "Khong tim thay ten san pham co so luong it nhu vay!" << endl;
    }
}

```

```

        break;
    }
    case 17:
    {
        if (!isEmpty(T))
        {
            Item x=Max_value(T);
            cout<<"Loai san pham co gia cao nhat:\n";
            cout<<x.NhaCungCap<<"\t\t"<<x.TenSanPham<<"\t\t"<<x.Soluong<<"\t\t"
            <<x.DonGia<<"\t\t"<<x.DVT<<endl;
        }
        else
        {
            cout<<"Danh sach rong!\n";
        }

        break;
    }
    case 18://in ra thong tin co sp gia thap nhat
    {
        if (!isEmpty(T))
        {
            Item x=Min_value(T);
            cout<<"Loai san pham co gia thap nhat:\n";
            cout<<x.NhaCungCap<<"\t\t"<<x.TenSanPham<<"\t\t"<<x.Soluong<<"\t\t"
            <<x.DonGia<<"\t\t"<<x.DVT<<endl;
        }
        else
        {
            cout<<"Danh sach rong!\n";
        }

        break;
    }
    case 19:// in ra Gia tren x
    {
        int x;
        bool i=false;
        cout<<"Gia tren x:";cin>>x;
        Price_up_x(T,x,i);
        if(!i){
            cout<<"Khong co Gia nao tren: "<<x;
        }
    }

```

```

        break;
    }
    case 20: // in ra gia duoi x
    {
        int x;
        cout<<"Gia tren x:";cin>>x;
        InDuoix(T,x);
        if(isEmpty(T)){
            cout<<"Khong co Gia nao duoi: "<<x;
        }
        break;
    }

    case 21:// total sl gia tren x
    {
        float x;
        cout<<"Nhap muc gia: ";
        cin>>x;
        if(x>Max_value(T).DonGia){
            cout<<"Khong tim thay gia tri nao tren "<<x;
            cout<<endl;
        }
        else
        {
            int sl=Sl_PriceX(T,x);
            cout<<"Tong Sl la: "<<sl<<endl;
        }
        break;
    }
    case 22:
    {
        float x;
        cout<<"Nhap muc gia: ";
        cin>>x;
        if(x<Min_value(T).DonGia){
            cout<<"Khong tim thay gia tri nao duoi "<<x;
            cout<<endl;
        }
        else
        {
            int sl=Total_SL_downX(T,x);
            cout<<"Tong Sl la: "<<sl<<endl;
        }
    }
}

```

```

    }
    break;
}
case 23:
{
    float x;
    cout << "Nhap muc gia it hon de xoa: ";
    cin >> x;
    if (Min_value(T).DonGia < x)
    {
        XoaSpGiaDuoix(T, x);
        cout << "Da xoa thanh cong!\n";
    }
    else
    {
        cout << "Khong co san pham nao co gia tren " << x << " de xoa khoi
danh sach\n";
    }
    break;
}
case 24:// delete gia sp tren x
{
    float x;
    cout<<"Delete Gia tren X: ";cin>>x;
    DeletePrice(T,x);
    cout<<"Da xoa xog!";
    break;
}
case 25:
{
    if (!isEmpty(T))
    {
        string name[10];
        Sl_NCC(T,name);
        int i=0;
        while (name[i]!="")
        {
            i++;
        }
        cout<<"So luong nha cung cap trong danh sach la: "<<i<<endl;
    }
    else
    {

```

```

        cout<<"Danh sach rong!\n";
    }
    break;
}
case 26:
{
    if (!isEmpty(T))
    {
        string name[DemSoSanPham(T)+1];
        SL_NCC(T,name);
        int i=0;
        cout<<"Phan phoi cac loai san pham theo nha cung cap:\n";
        while (name[i]!="")
        {
            cout<<name[i]<<endl;
            TimSanPhamHangX1(T,name[i]);
            i++;
        }
    }
    else
    {
        cout<<"Danh sach rong!\n";
    }
    break;
}
case 27://xoa sp gia cao nhat
{
    if (!isEmpty(T)){
        DeletePriceTall(T);
        cout<<"Da xoa! ";
    }else{
        cout<<"Danh sach Rong!";
    }
    break;
}
case 28:
{
    if(!isEmpty(T)){
        XoaSpGiaMin(T);
        cout<<"Da xoa"<<endl;
    }
    else{
        cout<<"Danh sach rong!"<<endl;
    }
}

```



```

        }
        break;
    }
    case 29:
    {
        float x;
        cout<<"Nhap gia san pham can xoa: ";
        cin>>x;
        if (!kiemtragia(T,x))
        {
            cout<<"Khong tim thay loai san pham nao co gia bang "<<x<<endl;
        }
        else
        {
            XoaNodex(T,x);
            cout<<"Da xoa!";
        }

        break;
    }
    case 30 :
    {
        cout<<"Tam biet!";
        break;
    }
}
}while(LuaChon!=30);
}
int main()
{
    Tree T;
    Menu(T);
    return 0;
}

```

## KẾT LUẬN

Cấu trúc cây nhị phân là một công cụ mạnh mẽ với nhiều ứng dụng trong thực tế, đặc biệt trong lĩnh vực phát triển sản phẩm. Nhờ khả năng lưu trữ và tổ chức dữ liệu hiệu quả, cây nhị phân mang lại nhiều lợi ích cho các sản phẩm, bao gồm: **Tìm kiếm và truy xuất dữ liệu nhanh chóng:** Cấu trúc cây nhị phân cho phép tìm kiếm dữ liệu một cách hiệu quả, đặc biệt là khi tập dữ liệu lớn. Việc tìm kiếm theo khóa có thể được thực hiện trong thời gian logarit của số lượng phần tử trong cây, giúp tiết kiệm thời gian và tài nguyên máy tính. **Sắp xếp dữ liệu:** Cây nhị phân có thể được sử dụng để sắp xếp dữ liệu một cách hiệu quả. Thuật toán sắp xếp nhị phân có thể sắp xếp một tập dữ liệu lớn trong thời gian logarit của số lượng phần tử. **Lưu trữ dữ liệu có thứ tự:** Cấu trúc cây nhị phân phù hợp để lưu trữ dữ liệu có thứ tự, chẳng hạn như danh sách, tập hợp và đồ thị. Việc lưu trữ dữ liệu theo thứ tự giúp truy cập và thao tác dữ liệu dễ dàng hơn. **Tối ưu hóa hiệu suất:** Cấu trúc cây nhị phân có thể được sử dụng để tối ưu hóa hiệu suất của các thuật toán và ứng dụng.

**Tóm lại:** Cấu trúc cây nhị phân là một công cụ hữu ích cho các nhà phát triển sản phẩm. Nhờ khả năng lưu trữ, tổ chức và truy xuất dữ liệu hiệu quả, cây nhị phân có thể giúp cải thiện hiệu suất, chức năng và khả năng sử dụng của sản phẩm.

## DANH MỤC TÀI LIỆU THAM KHẢO

- [1]. Đặng Hải Phong [ Code cấu trúc cây nhị phân trên lớp]  
[2].<https://www.youtube.com/watch?v=4xTLz5eKT0k&t=592s&pp=ygUfY2F1IHRydWMgY2F5IG5oaSBwaGFuICB4b2Egdm9kZQ%3D%3D>