BÁO CÁO ĐỀ TÀI OOP

Nhóm <TKD>

**Chủ đề: Quản lý điện thoại**

Danh sách thành viên

1. 49.01.104.071 Trần Quốc Khánh
2. 49.01.104.020 Ngô Lê Đức Duy
3. 49.01.104.163 Lưu Quang Trường
4. 49.01.104.043 Trần Đắc Hiểu

# MỤC LỤC

Contents

[MỤC LỤC 1](file:///D:\Download\OOP_BaoCao-1.docx#_Toc140048785)

[GIỚI THIỆU ĐỀ TÀI 2](file:///D:\Download\OOP_BaoCao-1.docx#_Toc140048786)

[LẬP TRÌNH HƯỚNG ĐỐI TƯỢNG 2](file:///D:\Download\OOP_BaoCao-1.docx#_Toc140048787)

[Mô tả đề 2](file:///D:\Download\OOP_BaoCao-1.docx#_Toc140048788)

[Sơ đồ lớp minh họa 3](file:///D:\Download\OOP_BaoCao-1.docx#_Toc140048789)

[Dữ liệu kiểm thử 4](file:///D:\Download\OOP_BaoCao-1.docx#_Toc140048790)

[Bài giải mẫu 5](file:///D:\Download\OOP_BaoCao-1.docx#_Toc140048791)

# GIỚI THIỆU ĐỀ TÀI

Hiện nay các cửa hàng điện thoại được mở rất nhiều nơi. Trong đó các cửa hàng không chỉ bán các sản phẩm điện thoại mới mà còn có những chiếc điện thoại gần như mới (like new) hay những chiếc điện thoại thanh lý cũ đã qua sử dụng để phù hợp hơn với nhiều đối tượng khách hàng với nhiều điều kiện tài chính cũng như nhu cầu khác nhau. Nhằm đảm bảo quyền lợi cho khách hàng mua điện thoại cũng như thuận tiện cho cửa hàng quản lí giá cho các dạng điện thoại nên phần mềm quản lí điện thoại được ra đời để giải quyết các yêu cầu đó.

# LẬP TRÌNH HƯỚNG ĐỐI TƯỢNG

## Mô tả đề

Một cửa hàng điện thoại cần quản lí đơn giá cũng như thông tin của từng loại sản phẩm. Đối với mỗi chiếc điện thoại ta cần quản lí các thông tin sau:

* Tên sản phẩm
* Tên hãng sản xuẩt
* Năm sản xuất
* Giá gốc

Trong đó Giá gốc là giá tiền của chiếc điện thoại khi nhập về của hàng (đã tính cả thuế và số tiền lời tối đa mà của hàng đặt ra), đây là thuộc tính dành cho tất cả sản phẩm và đơn giá của chúng khác nhau.

Những chiếc điện thoại trong cửa hàng chia làm 3 loại: điện thoại mới, likenew và thanh lí cũ. Đối với từng loại hàng có cách tính tiền khác nhau như sau:

* Đối với điện thoại mới:

**Số tiền = Đơn giá – Đơn giá \* hệ số khuyến mãi;**

Trong đó, **hệ số khuyến mãi** (tính theo phần trăm) là thông tin của sản phẩm mới được cửa hàng đặt ra.

* Đối với điện thoại like new:

**Số tiền = Đơn giá \* tỉ lệ mới;**

Trong đó, **tỉ lệ mới** (tính theo phần trăm) là thông tin của sản phẩm like new mà cửa hàng đưa ra bằng cách kiểm tra điện thoại và đánh giá một cách cẩn thận (ví dụ: điện thoại còn mới đến 99% thì tỉ lệ mới sẽ bằng 99)

* Đối với điện thoại cũ:

**Số tiền = Đơn giá + phí bảo hành - tỉ lệ hao mòn thời gian - Đơn giá \* tỉ lệ hao mòn**

Trong đó:

* **tỉ lệ hao mòn** (tính theo phần trăm) là chỉ số thể hiện độ cũ của máy
* **tỉ lệ hao mòn** **thời gian** = (**Năm bán** – **năm sản xuất**) \* **tỉ lệ giảm (tính theo phần trăm, là chỉ số thể hiện tỉ lệ hao mòn vô hình theo từng năm, do cửa hàng tự quy định theo thị trường)**
* **Phí bảo hành** là số tiền mà khách hàng phải bỏ ra để được nhận dịch vụ bảo hành của cửa hàng.

Ngoài ra, chương trình quản lí cơ bản phải xuất được thông tin của từng chiếc điện thoại và tính tiền của từng chiếc điện thoại vào thời điểm được bán.

1. Hãy khai báo thuộc tính, phương thức của các lớp và mối quan hệ giữa chúng bằng ngôn ngữ C++.
2. Viết 3 loại constructor cho các lớp điện thoại.
3. Khai báo và cài đặt phương thức để hiển thị thông tin cho mỗi loại điện thoại.
4. Khai báo và cài đặt phương thức tính tiền sử dụng cho các loại điện thoại.
5. Viết 1 hàm main có nội dung như sau: nhập vào 1 danh sách N phần tử là các điện thoại của của cửa hàng trên, yêu cầu sử dụng **tính đa hình** để xuất thông tin của từng chiếc điện thoại và xuất được giá tiền từng chiếc điện thoại.

## Sơ đồ lớp minh họa

A diagram of a computer

Description automatically generated

## Dữ liệu kiểm thử

Input:

* Dòng 1: 1 số nguyên N (số điện thoại cần tính giá tiền)
* N x 6 dòng tiếp theo là thông tin từ các loại điện thoại với cấu trúc như sau:
* NEW – ký hiệu điện thoại mới, LIKENEW là điện thoại gần như mới, OLD là điện thoại
* Những dòng tiếp theo bao gồm:

+ Tên sản phẩm

+ Tên hãng sản xuẩt

+ Năm sản xuất

+ Giá gốc

* Dòng cuối của mỗi sản phẩm là thông tin riêng của từng loại điện thoại:

+ **Hệ số khuyến mãi** đối với điện thoại mới

**+ Tỉ lệ mới** đối với điện thoại Like New

+ Riêng điện thoại cũ bao gồm 4 dòng:

* + Năm bán
  + Tỉ lệ hao mòn
  + Phí bảo hành
  + Tỉ lệ giảm

Output:

* Dòng 1: In chữ “\*\*\*HOA DON\*\*\*” (không có có dấu “)
* N x 5 dòng tiếp theo in thông tin của từng chiếc điện thoại và giá tiền của từng máy (xem ví dụ để hiểu rõ hơn về cách xuất danh sách.

| **Input** | **Output** |
| --- | --- |
| 1  NEW  I Phone 15 Pro Max  APPLE  2023  35000000  10 | \*\*\*HOA DON\*\*\*  1. Loai dien thoai: New  Ten dien thoai: I Phone 15 Pro Max  Ten hang: APPLE  Nam san xuat: 2023  So tien phai tra: 31500000 |

| **Input** | **Output** |
| --- | --- |
| 2  NEW  I Phone 15 Pro Max  APPLE  2024  35000000  10  LIKENEW  SAMSUNG S24 ULTRA  SAMSUNG  2024  36000000  99 | \*\*\*HOA DON\*\*\*  1. Loai dien thoai: New  Ten dien thoai: I Phone 15 Pro Max  Ten hang: APPLE  Nam san xuat: 2024  So tien phai tra: 31500000  2. Loai dien thoai: Like New  Ten dien thoai: SAMSUNG S24 ULTRA  Ten hang: SAMSUNG  Nam san xuat: 2024  So tien phai tra: 35640000 |

## Bài giải mẫu

Tên bài trên Upcoder: Quản lí điện thoại

Mã bài trên Upcoder: GK\_OOP\_QLSPhone

Bộ testcase (ít nhất 8 test case bao phủ tất cả các đối tượng)

|  |  |
| --- | --- |
| **Input** | **Output** |
| |  | | --- | | 3 | | NEW | | Samsung Galaxy S22 | | Samsung | | 2024 | | 20000000 | | 15 | | LIKENEW | | iPhone 12 | | Apple | | 2023 | | 18000000 | | 90 | | OLD | | OnePlus 7T | | OnePlus | | 2019 | | 12000000 | | 2022 | | 25 | | 1500000 | | 10 | | |  | | --- | | \*\*\*HOA DON\*\*\* | | 1. Loai dien thoai: New | | Ten dien thoai: Samsung Galaxy S22 | | Ten hang: Samsung | | Nam san xuat: 2024 | | So tien phai tra: 17000000 | |  | | 2. Loai dien thoai: Like New | | Ten dien thoai: iPhone 12 | | Ten hang: Apple | | Nam san xuat: 2023 | | So tien phai tra: 16200000 | |  | | 3. Loai dien thoai: OLD | | Ten dien thoai: OnePlus 7T | | Ten hang: OnePlus | | Nam san xuat: 2019 | | So tien phai tra: 6900000 | |  | |  | |  | |  | |

|  |  |
| --- | --- |
| **Input** | **Output** |
| 4  NEW  OnePlus Nord 2  OnePlus  2023  15000000  30  OLD  Google Pixel 5  Google  2020  20000000  2023  5  0  10  LIKENEW  Xiaomi Mi 11 Lite  Xiaomi  2023  12000000  80  NEW  I Phone 14 Pro  Apple  2025  50000000  0 | \*\*\*HOA DON\*\*\*  1. Loai dien thoai: New  Ten dien thoai: OnePlus Nord 2  Ten hang: OnePlus  Nam san xuat: 2023  So tien phai tra: 10500000  2. Loai dien thoai: OLD  Ten dien thoai: Google Pixel 5  Ten hang: Google  Nam san xuat: 2020  So tien phai tra: 13000000  3. Loai dien thoai: Like New  Ten dien thoai: Xiaomi Mi 11 Lite  Ten hang: Xiaomi  Nam san xuat: 2023  So tien phai tra: 9600000  4. Loai dien thoai: New  Ten dien thoai: I Phone 14 Pro  Ten hang: Apple  Nam san xuat: 2025  So tien phai tra: 50000000 |

|  |  |
| --- | --- |
| **Input** | **Output** |
| 8  NEW  Google Pixel 6  Google  2023  35000000  20  LIKENEW  OnePlus 9 Pro  OnePlus  2022  28000000  85  OLD  Samsung galaxy note 10  Samsung  2019  15000000  2023  30  2000000  15  NEW  Xiaomi redmi 10  Xiaomi  2024  80000000  10  LIKENEW  I phone SE(2nd generation)  Apple  2021  16000000  99  OLD  Samsung galaxy s9  Samsung  2018  10000000  2023  10  500000  5  NEW  Samsung galaxy A12  Samsung  2025  6000000  0  LIKENEW  IPhone 13 Pro  Apple  2023  35000000  95  LIKENEW  Xiaomi Mi 11 ULTRA  Xiaomi  2022  38000000  80 | \*\*\*HOA DON\*\*\*  1. Loai dien thoai: New  Ten dien thoai: Google Pixel 6  Ten hang: Google  Nam san xuat: 2023  So tien phai tra: 28000000  2. Loai dien thoai: Like New  Ten dien thoai: OnePlus 9 Pro  Ten hang: OnePlus  Nam san xuat: 2022  So tien phai tra: 23800000  3. Loai dien thoai: OLD  Ten dien thoai: Samsung galaxy note 10  Ten hang: Samsung  Nam san xuat: 2019  So tien phai tra: 3500000  4. Loai dien thoai: New  Ten dien thoai: Xiaomi redmi 10  Ten hang: Xiaomi  Nam san xuat: 2024  So tien phai tra: 72000000  5. Loai dien thoai: Like New  Ten dien thoai: I phone SE(2nd generation)  Ten hang: Apple  Nam san xuat: 2021  So tien phai tra: 15840000  6. Loai dien thoai: OLD  Ten dien thoai: Samsung galaxy s9  Ten hang: Samsung  Nam san xuat: 2018  So tien phai tra: 7000000  7. Loai dien thoai: New  Ten dien thoai: Samsung galaxy A12  Ten hang: Samsung  Nam san xuat: 2025  So tien phai tra: 6000000  8. Loai dien thoai: Like New  Ten dien thoai: IPhone 13 Pro  Ten hang: Apple  Nam san xuat: 2023  So tien phai tra: 33250000 |

|  |  |
| --- | --- |
| **Input** | **Output** |
| 1  LIKENEW  IPhone 11  Apple  2021  20000000  99 | \*\*\*HOA DON\*\*\*  1. Loai dien thoai: Like New  Ten dien thoai: IPhone 11  Ten hang: Apple  Nam san xuat: 2021  So tien phai tra: 19800000 |

|  |  |
| --- | --- |
| **Input** | **Output** |
| 3  LIKENEW  OnePlus 8T  OnePlus  2022  20000000  90  LIKENEW  Pixel 6  Pixel  2023  3000000  80  LIKENEW  Samsung galaxy S21 FE  Samsung  2023  25000000  70 | \*\*\*HOA DON\*\*\*  1. Loai dien thoai: Like New  Ten dien thoai: OnePlus 8T  Ten hang: OnePlus  Nam san xuat: 2022  So tien phai tra: 18000000  2. Loai dien thoai: Like New  Ten dien thoai: Pixel 6  Ten hang: Pixel  Nam san xuat: 2023  So tien phai tra: 2400000  3. Loai dien thoai: Like New  Ten dien thoai: Samsung galaxy S21 FE  Ten hang: Samsung  Nam san xuat: 2023  So tien phai tra: 17500000 |

|  |  |
| --- | --- |
| **Input** | **Output** |
| 3  NEW  Samsung galaxy A32  Samsung  2024  80000000  10  NEW  Iphone SE(3rd generation)  Apple  2025  10000000  5  NEW  Xiaomi Redmi note 10  Xiaomi  2023  12000000  15 | \*\*\*HOA DON\*\*\*  1. Loai dien thoai: New  Ten dien thoai: Samsung galaxy A32  Ten hang: Samsung  Nam san xuat: 2024  So tien phai tra: 72000000  2. Loai dien thoai: New  Ten dien thoai: Iphone SE(3rd generation)  Ten hang: Apple  Nam san xuat: 2025  So tien phai tra: 9500000  3. Loai dien thoai: New  Ten dien thoai: Xiaomi Redmi note 10  Ten hang: Xiaomi  Nam san xuat: 2023  So tien phai tra: 10200000 |

|  |  |
| --- | --- |
| **Input** | **Output** |
| 3  OLD  Iphone 11 pro  Apple  2019  15000000  2023  20  1000000  10  OLD  Samsung galaxy note 9  Samsung  2018  12000000  2022  30  2000000  5  OLD  OnePlus 7 Pro  OnePlus  2019  18000000  2023  15  1500000  15 | \*\*\*HOA DON\*\*\*  1. Loai dien thoai: OLD  Ten dien thoai: Iphone 11 pro  Ten hang: Apple  Nam san xuat: 2019  So tien phai tra: 7000000  2. Loai dien thoai: OLD  Ten dien thoai: Samsung galaxy note 9  Ten hang: Samsung  Nam san xuat: 2018  So tien phai tra: 8000000  3. Loai dien thoai: OLD  Ten dien thoai: OnePlus 7 Pro  Ten hang: OnePlus  Nam san xuat: 2019  So tien phai tra: 6000000 |

|  |  |
| --- | --- |
| **Input** | **Output** |
| 1  OLD  Samsung Galaxy S8  Samsung  2017  8000000  2022  100  1000000  0 | \*\*\*HOA DON\*\*\*  1. Loai dien thoai: OLD  Ten dien thoai: Samsung Galaxy S8  Ten hang: Samsung  Nam san xuat: 2017  So tien phai tra: 1000000 |

|  |  |
| --- | --- |
| **Input** | **Output** |
| 1  OLD  Samsung galaxy s10  Samsung  2022  15000000  2022  20  1000000  10 | \*\*\*HOA DON\*\*\*  1. Loai dien thoai: OLD  Ten dien thoai: Samsung galaxy s10  Ten hang: Samsung  Nam san xuat: 2022  So tien phai tra: 13000000 |

|  |  |
| --- | --- |
| **Input** | **Output** |
| 1  OLD  Iphone XS Max  Apple  2018  20000000  2018  5  0 | \*\*\*HOA DON\*\*\*  1. Loai dien thoai: Old  Ten dien thoai: Iphone XS Max  Ten hang: Apple  Nam san xuat: 2018  So tien phai tra: 19000000 |

Link codepad mẫu trên Upcoder: [Tại đây](https://url.1click.top/SYfzO1423)

Mã nguồn code mẫu kèm ghi chú giải thích:

#include <bits/stdc++.h>

using namespace std;

class DienThoai { // Lớp điện thoại là lớp cơ sở cho các loại điện thoại khác nhau

private: // Khai báo các biến của lớp cơ sở

string TenSP;

string TenHang;

int NamSX;

long long DonGia;

public:

// Hàm khởi tạo gán các giá trị mặc định

DienThoai(string \_TenSP = "", string \_TenHang = "", int \_NamSX = 2000, long long \_DonGia = 0) {

TenSP = \_TenSP;

TenHang = \_TenHang;

NamSX = \_NamSX;

DonGia = \_DonGia;

}

// hàm sao chép tạo bản sao cho đối tượng điện thoại hiện có

DienThoai(const DienThoai& dt) {

TenSP = dt.TenSP;

TenHang = dt.TenHang;

NamSX = dt.NamSX;

DonGia = dt.DonGia;

}

// hàm hủy

~DienThoai() {}

// phương thức getter - setter để truy cập và sửa đổi các biến private của đối tượng

void setTenSP(string s) { TenSP = s; }

string getTenSP() { return TenSP; }

void setTenHang(string s) { TenHang = s; }

string getTenHang() { return TenHang; }

void setNamSX(int n) { NamSX = n; }

int getNamSX() { return NamSX; }

void setDonGia(long long l) { DonGia = l; }

long long getDonGia() { return DonGia; }

// Hàm thuần ảo - vì không cần sử dụng hàm trong lớp cơ sở, chỉ phục vụ cho việc xuất

virtual long long TinhTien() = 0;

virtual string LoaiDienThoai() = 0;

// Hàm xuất để hiện thị thông tin điện thoại và giá của điện thoại

virtual void Xuat() {

cout << "Loai dien thoai: " << LoaiDienThoai() << endl;

cout << "Ten dien thoai: " << TenSP << endl;

cout << "Ten hang: " << TenHang << endl;

cout << "Nam san xuat: " << NamSX << endl;

cout << "So tien phai tra: " << TinhTien() << endl;

}

};

class NEW : public DienThoai { // lớp điện thoại mới (NEW) có kế thừa từ lớp điện thoại

private:

int heSoKM; // khai báo biến private hệ số khuyến mãi

public:

// Hàm khởi tạo gán các giá trị mặc định

NEW(string \_TenSP = "", string \_TenHang = "", int \_NamSX = 2000, long long \_DonGia = 0, int \_heSoKM = 0) :DienThoai(\_TenSP, \_TenHang, \_NamSX, \_DonGia) {

heSoKM = \_heSoKM;

}

// hàm hủy

~NEW() {}

// phương thức getter - setter để truy cập và sửa đổi các biến private của đối tượng

void setHeSoKM(int n) { heSoKM = n; }

int getHeSoKM() { return heSoKM; }

// định nghĩa lại hàm thuần ảo kế thừa từ lớp cơ sở

string LoaiDienThoai() { return "New"; }

long long TinhTien() { return DienThoai::getDonGia() - ((DienThoai::getDonGia() \* heSoKM) / 100); }

};

class LIKE\_NEW : public DienThoai { // lớp điện thoại mới (LIKENEW) được kế thừa từ lớp điện thoại

private:

int tiLeMoi; // Khai báo biến private tỉ lệ mới

public:

// Hàm khởi tạo giá trị mặc định

LIKE\_NEW(string \_TenSP = "", string \_TenHang = "", int \_NamSX = 2000, long long \_DonGia = 0, int \_tiLeMoi = 0) :DienThoai(\_TenSP, \_TenHang, \_NamSX, \_DonGia) {

tiLeMoi = \_tiLeMoi;

}

// Hàm hủy

~LIKE\_NEW() {}

// phương thức getter - setter để truy cập và sửa đổi các biến private của đối tượng

void settiLeMoi(int n) { tiLeMoi = n; }

int gettiLeMoi() { return tiLeMoi; }

// định nghĩa lại hàm thuần ảo kế thừa từ lớp cơ sở

string LoaiDienThoai() { return "Like New"; }

long long TinhTien() { return DienThoai::getDonGia() \* tiLeMoi / 100; }

};

class OLD : public DienThoai { // Lớp địện thoại cũ (OLD) được kế thừa từ lớp điện thoại

private: // Khai báo biến private

int tiLeHaoMon;

int NamBan;

long long phiBaoHanh;

int tiLeGiam;

public:

// Hàm khởi tạo giá trị mặc định

OLD(string \_TenSP = "", string \_TenHang = "", int \_NamSX = 2000, long long \_DonGia = 0, int \_NamBan = 2000, int \_tiLeHaoMon = 0, long long \_phiBaoHanh = 0, int \_tiLeGiam = 0) : DienThoai(\_TenSP, \_TenHang, \_NamSX, \_DonGia) {

tiLeHaoMon = \_tiLeHaoMon;

NamBan = \_NamBan;

phiBaoHanh = \_phiBaoHanh;

tiLeGiam = \_tiLeGiam;

}

// Hàm hủy

~OLD() {}

// phương thức getter - setter để truy cập và sửa đổi các biến private của đối tượng

void setTiLeHaoMon(int n) { tiLeHaoMon = n; }

int getTiLeHaoMon() { return tiLeHaoMon; }

void setNamBan(int n) { NamBan = n; }

int getNamBan() { return NamBan; }

void setPhiBaoHanh(long long n) { phiBaoHanh = n; }

long long getPhiBaoHanh() { return phiBaoHanh; }

void setTiLeGiam(int n) { tiLeGiam = n; }

int getTiLeGiam() { return tiLeGiam; }

// định nghĩa lại hàm thuần ảo kế thừa từ lớp cơ sở

string LoaiDienThoai() { return "OLD"; }

long long TinhTien() {

return DienThoai::getDonGia() + phiBaoHanh - (DienThoai::getDonGia() \* (NamBan - DienThoai::getNamSX()) \* tiLeGiam / 100)

- ((DienThoai::getDonGia() \* tiLeHaoMon) / 100);

}

};

int main()

{

// Khai báo các biến sẽ được sử dụng trong chương trình

string sLoaiDienThoai, sTenHang, sTenSP;

long long DonGia, PhiBaoHanh;

int NamSX, HeSoKM, tiLeMoi, TiLeHaoMon, NamBan, TiLeGiam;

// Khai báo một vector dt có kiểu dữ liệu là con trỏ của lớp cơ sở điện thoại để lưu thông tin các loại điện thoại

vector<DienThoai\*> dt;

// Nhập vào sô lượng điện thoại cần xử lí

long long n; cin >> n;

// vòng lặp để chương trình thực hiện đến khi hết số lượng phần tử

for (long long i = 0; i < n; i++) {

// đọc vào các thông tin của điện thoại

cin >> sLoaiDienThoai;

cin.ignore();

getline(cin, sTenSP);

getline(cin, sTenHang);

cin >> NamSX;

cin >> DonGia;

// Dựa vào giá trị của biến sLoaiDienThoai để xác định loại điện thoại cần xử lí

if (sLoaiDienThoai == "NEW") {

// nếu là điện thoại mới sẽ nhập thêm giá trị hệ số khuyễn mãi

cin >> HeSoKM;

// đẩy thông tin của loại điện thoại vào vector bằng hàm khởi tạo

dt.push\_back(new NEW(sTenSP, sTenHang, NamSX, DonGia, HeSoKM));

}

else if (sLoaiDienThoai == "LIKENEW") {

// nếu là điện thoại like new sẽ nhâp thêm giá trị riêng là tỉ lệ mới

cin >> tiLeMoi;

// đẩy thông tin của loại điện thoại vào vector bằng hàm khởi tạo

dt.push\_back(new LIKE\_NEW(sTenSP, sTenHang, NamSX, DonGia, tiLeMoi));

}

else {

// nếu là điện thoại cũ sẽ nhập thêm 4 giá trị là năm bán, tỉ lệ hao mòn, phí bảo hành và tỉ lệ giảm

cin >> NamBan;

cin >> TiLeHaoMon;

cin >> PhiBaoHanh;

cin >> TiLeGiam;

// đẩy thông tin của loại điện thoại vào vector bằng hàm khởi tạo

dt.push\_back(new OLD(sTenSP, sTenHang, NamSX, DonGia, NamBan, TiLeHaoMon, PhiBaoHanh, TiLeGiam));

}

}

cout << "\*\*\*HOA DON\*\*\*" << endl;

// in thông tin của từng sản phẩm thông qua hàm xuất

for (long long i = 0; i < n; i++) {

cout << i + 1 << ". ";

dt[i]->Xuat();

cout << endl;

}

return 0;

}

Trên đây là bài báo cáo về đề tài OOP của nhóm em đó là quản lí giá bán của điện thoại, cảm ơn thầy đã xem qua. Nếu có phần nào cần sửa thì chúng em rất mong nhận được sự phản hồi từ thầy. Chúng em cảm ơn !