**Task 1**

#include <iostream>

#include <string>

using namespace std;

class Novel {

private:

string bookName;

float cost;

int\* availableStock;

public:

Novel() {

bookName = "Unknown";

cost = 0.0;

availableStock = new int(30);

}

void modifyDetails() {

cout << "Enter new title for Novel " << bookName << ": ";

getline(cin, bookName);

cout << "Enter new price: $";

cin >> cost;

cin.ignore();

cout << "Enter new stock quantity: ";

cin >> \*availableStock;

cin.ignore();

}

void buyNovel() {

if(\*availableStock <= 0) {

cout << "Out of stock... Purchase not possible" << endl;

}

else {

(\*availableStock)--;

cout << "Novel " << bookName << " bought!" << endl;

if(\*availableStock < 5) {

cout << "WARNING: LOW STOCK!" << endl;

}

}

}

};

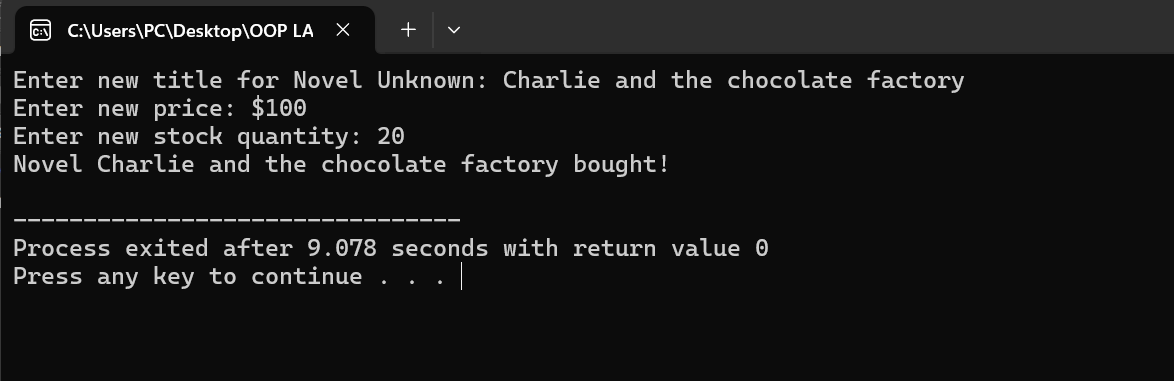
int main() {

Novel thisNovel;

thisNovel.modifyDetails();

thisNovel.buyNovel();

}



**Task 2**

#include <iostream>

#include <string>

using namespace std;

class Novel {

private:

string bookName;

float cost;

int\* availableStock;

public:

Novel() {

bookName = "Unknown";

cost = 0.0;

availableStock = new int(30);

}

Novel(string bookName, float cost, int availableStock) {

this->bookName = bookName;

this->cost = cost;

this->availableStock = new int(availableStock);

}

void modifyDetails() {

cout << "Enter new title for Novel : ";

getline(cin, bookName);

cout << "Enter new price: $";

cin >> cost;

cin.ignore();

cout << "Enter new stock quantity: ";

cin >> \*availableStock;

cin.ignore();

}

void buyNovel() {

if (\*availableStock == 0) {

cout << "Out of stock... Purchase not possible." << endl;

return;

}

int amount;

cout << "Enter quantity to buy: ";

cin >> amount;

if (\*availableStock < amount) {

cout << "Requested quantity exceeds available stock (" << \*availableStock << "), cannot purchase Novel " << bookName << endl;

return;

}

int rebate = getDiscount(amount);

float finalCost = cost \* amount \* (100 - rebate) / 100.0;

cout << "Purchasing Novel " << bookName << "..." << endl;

cout << "Total: $" << finalCost << endl;

cout << "Discount applied: " << rebate << "%" << endl;

cout << "Novel " << bookName << " bought." << endl;

\*availableStock -= amount;

if (\*availableStock < 5) {

cout << "WARNING: LOW STOCK!" << endl;

}

}

int getDiscount(int amount) {

if(amount > 10) {

return 10;

} else if (amount > 5) {

return 5;

}

return 0;

}

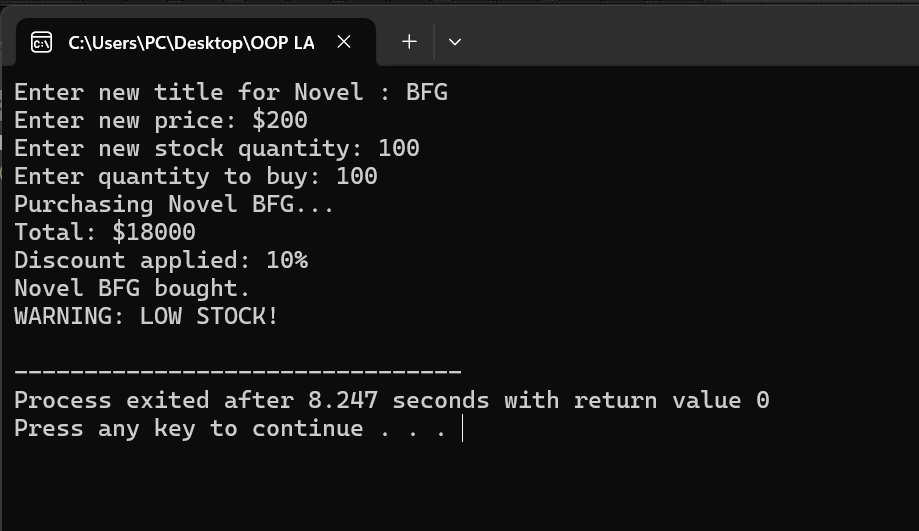
};

int main() {

Novel myNovel;

myNovel.modifyDetails();

myNovel.buyNovel();

}  


**Task 3**

#include <iostream>

#include <string>

using namespace std;

class Novel {

private:

string bookName;

float cost;

int\* availableStock;

public:

Novel() {

bookName = "Unknown";

cost = 0.0;

availableStock = new int(30);

}

Novel(string bookName, float cost, int availableStock) {

this->bookName = bookName;

this->cost = cost;

this->availableStock = new int(availableStock);

}

Novel(const Novel& source) {

bookName = source.bookName;

cost = source.cost;

availableStock = new int(\*source.availableStock);

}

void modifyDetails() {

cout << "Enter new title for Novel " << bookName << ": ";

getline(cin, bookName);

cout << "Enter new price: $";

cin >> cost;

cin.ignore();

cout << "Enter new stock quantity: ";

cin >> \*availableStock;

cin.ignore();

}

void buyNovel() {

if (\*availableStock == 0) {

cout << "Out of stock... Purchase not possible." << endl;

return;

}

int amount;

cout << "Enter quantity to buy: ";

cin >> amount;

if (\*availableStock < amount) {

cout << "Requested quantity exceeds available stock (" << \*availableStock << "), cannot purchase Novel " << bookName << endl;

return;

}

int rebate = getDiscount(amount);

float finalCost = cost \* amount \* (100 - rebate) / 100.0;

cout << "Purchasing Novel " << bookName << "..." << endl;

cout << "Total: $" << finalCost << endl;

cout << "Discount applied: " << rebate << "%" << endl;

cout << "Novel " << bookName << " bought." << endl;

\*availableStock -= amount;

if (\*availableStock < 5) {

cout << "WARNING: LOW STOCK!" << endl;

}

}

int getDiscount(int amount) {

if(amount > 10) {

return 10;

} else if (amount > 5) {

return 5;

}

return 0;

}

~Novel() {

delete availableStock;

}

};

int main() {

Novel novel1;

Novel novel2("C++ Guide", 39.99, 20);

Novel novel3 = novel2;

cout << "Updating details for novel1 (default constructor):" << endl;

novel1.modifyDetails();

cout << "\nPurchasing novel2 (parameterized constructor):" << endl;

novel2.buyNovel();

cout << "\nPurchasing novel3 (copy constructor):" << endl;

novel3.buyNovel();

return 0;

}

A screenshot of a computer program

AI-generated content may be incorrect.

**Task 4**

#include <iostream>

#include <string>

using namespace std;

class Novel {

private:

string bookName;

float cost;

int\* availableStock;

public:

Novel() {

bookName = "Unknown";

cost = 0.0;

availableStock = new int(30);

}

Novel(string bookName, float cost, int availableStock) {

this->bookName = bookName;

this->cost = cost;

this->availableStock = new int(availableStock);

}

Novel(const Novel& source) {

this->bookName = source.bookName;

this->cost = source.cost;

this->availableStock = new int(\*source.availableStock);

}

void modifyDetails() {

cout << "Enter new title for Novel " << bookName << ": ";

getline(cin, bookName);

cout << "Enter new price: $";

cin >> cost;

cin.ignore();

cout << "Enter new stock quantity: ";

cin >> \*availableStock;

cin.ignore();

}

void buyNovel() {

if (\*(this->availableStock) == 0) {

cout << "Out of stock... Purchase not possible." << endl;

return;

}

int amount;

cout << "Enter quantity to buy: ";

cin >> amount;

if (\*(this->availableStock) < amount) {

cout << "Requested quantity exceeds available stock (" << \*(this->availableStock) << "), cannot purchase Novel " << this->bookName << endl;

return;

}

int rebate = this->getDiscount(amount);

float finalCost = this->cost \* amount \* (100 - rebate) / 100.0;

cout << "Purchasing Novel " << this->bookName << "..." << endl;

cout << "Total: $" << finalCost << endl;

cout << "Discount applied: " << rebate << "%" << endl;

cout << "Novel " << this->bookName << " bought." << endl;

\*(this->availableStock) -= amount;

if (\*(this->availableStock) < 5) {

cout << "WARNING: LOW STOCK!" << endl;

}

}

int getDiscount(int amount) {

if(amount > 10) {

return 10;

} else if (amount > 5) {

return 5;

}

return 0;

}

~Novel() {

delete availableStock;

}

};

int main() {

Novel novel1;

Novel novel2("C++ Guide", 39.99, 20);

Novel novel3 = novel2;

cout << "Updating details for novel1 (default constructor):" << endl;

novel1.modifyDetails();

cout << "\nPurchasing novel2 (parameterized constructor):" << endl;

novel2.buyNovel();

cout << "\nPurchasing novel3 (copy constructor):" << endl;

novel3.buyNovel();

return 0;

}

A screenshot of a computer

AI-generated content may be incorrect.

**Task 5**

#include <iostream>

#include <string>

using namespace std;

class Novel {

private:

string bookName;

float cost;

int\* availableStock;

int novelID;

static int IDcounter;

public:

Novel() : bookName("Unknown"), cost(0.0), novelID(++IDcounter) {

availableStock = new int(30);

}

Novel(string bookName, float cost, int availableStock) : bookName(bookName), cost(cost), novelID(++IDcounter) {

this->bookName = bookName;

this->cost = cost;

this->availableStock = new int(availableStock);

}

Novel(const Novel& source) : novelID(++IDcounter) {

this->bookName = source.bookName;

this->cost = source.cost;

this->availableStock = new int(\*source.availableStock);

}

void modifyDetails() {

cout << "Enter new title for Novel " << bookName << ": ";

getline(cin, bookName);

cout << "Enter new price: $";

cin >> cost;

cin.ignore();

cout << "Enter new stock quantity: ";

cin >> \*availableStock;

cin.ignore();

}

void buyNovel() {

if (\*(this->availableStock) == 0) {

cout << "Out of stock... Purchase not possible." << endl;

return;

}

int amount;

cout << "Enter quantity to buy: ";

cin >> amount;

if (\*(this->availableStock) < amount) {

cout << "Requested quantity exceeds available stock (" << \*(this->availableStock) << "), cannot purchase Novel " << this->bookName << endl;

return;

}

int rebate = this->getDiscount(amount);

float finalCost = this->cost \* amount \* (100 - rebate) / 100.0;

cout << "Purchasing Novel " << this->bookName << "..." << endl;

cout << "Total: $" << finalCost << endl;

cout << "Discount applied: " << rebate << "%" << endl;

cout << "Novel " << this->bookName << " bought." << endl;

\*(this->availableStock) -= amount;

if (\*(this->availableStock) < 5) {

cout << "WARNING: LOW STOCK!" << endl;

}

}

int getDiscount(int amount) {

if(amount > 10) {

return 10;

} else if (amount > 5) {

return 5;

}

return 0;

}

void displayNovel() {

cout << "ID: " << novelID << endl << "Title: " << bookName << endl << "Price: $" << cost << endl << "Stock: " << \*availableStock << endl;

}

~Novel() {

delete availableStock;

}

};

int Novel::IDcounter = 0;

int main() {

Novel novel1;

Novel novel2("C++ Guide", 39.99, 20);

Novel novel3 = novel2;

cout << "Updating details for novel1 (default constructor):" << endl;

novel1.modifyDetails();

cout << "\nPurchasing novel2 (parameterized constructor):" << endl;

novel2.buyNovel();

cout << "\nPurchasing novel3 (copy constructor):" << endl;

novel3.buyNovel();

novel1.displayNovel();

novel2.displayNovel();

novel3.displayNovel();

return 0;

}

A screenshot of a computer program

AI-generated content may be incorrect.