

EXAM ANSWERS
Object-Oriented Programming in C++

EX 1: (10 pts)

1) a	6) b
2) b	7) d
3) b	8) c
4) c)	9) b
5) a	10) b

EX 2: (10 pts)

```
#include <iostream>
#include <string>
using namespace std;
```

```
//1)
```

```
class Book {
private:
```

```
    string title;
```

```
    string author;
```

1pt

```
    float price;
```

```
    int stock;
```

```
public:
```

```
    // 2) Default Constructor
```

```
    Book() : title(""), author(""), price(0), stock(0) {
```

```
        cout << "Default constructor called!" << endl; }
```

1pt

```
    // 3) Parameterized Constructor
```

```
    Book(string ttl, string ath, float pr, int st) : title(ttl), author(ath), price(pr), stock(st) {
```

```
        cout << "Parameterized constructor called!" << endl; }
```

1pt

```
    // 4)
```

```
    ~Book() {
```

```
        cout << "Destructor called for " << title << "!" << endl; }
```

1pt

```
    // 5) Member function to display information
```

```
    void displayInfo() {
```

```
        cout << "Title: " << title << endl;
```

```
cout << "Author: " << author << endl;
cout << "Price: " << price << endl;
cout << "Stock: " << stock << endl; }
```

1pt

//6) Member function to update price

```
void updatePrice(float UpdatedPrice) {
    price = UpdatedPrice; }
```

1pt

// 7) Member function to update stock

```
void updateStock(int UpdatedStock) {
    stock = UpdatedStock; }
```

1pt

```
};
```

```
int main()
```

```
{ // 8) Create objects using different constructors
```

```
    Book book1; // Default constructor
```

```
    Book book2("C++ programming and cheating", "ChatGpt", 17.50, 200); // Parameterized
constructor
```

1pt

//9) Updating price and stock of book2

```
    cout << "\nUpdating price and stock for book 2..." << endl;
    book2.updatePrice(12.95);
    book2.updateStock(220);
```

1pt

//10) Displaying information about the 2 books

```
    cout << "\nDisplaying book 1 info:" << endl;
    book1.displayInfo();
    cout << "\nDisplaying updated book 2 info:" << endl;
    book2.displayInfo();
    return 0; }
```

1 pt