

Dạng câu 1 (Lý thuyết)

1. Phân biệt các kiểu kế thừa private, protected, public.
2. Phân biệt các phạm vi truy cập private, protected, public.
3. Trình bày các đặc điểm quan trọng của lập trình hướng đối tượng.
4. Trình bày sự hiểu biết và cho ví dụ minh họa về khái niệm class (lớp) và object (đối tượng) trong lập trình hướng đối tượng.
5. Trình bày những đặc điểm của tính đóng gói (Encapsulation) trong lập trình hướng đối tượng. Trường hợp nào có thể vi phạm tính đóng gói? Cho ví dụ minh họa.
6. Trình bày những ưu điểm của kế thừa trong lập trình hướng đối tượng và cho ví dụ minh họa?
7. Phân biệt khái niệm overload (nạp chồng) và override (ghi đè).
8. Hàm thuần ảo là gì? Lớp trừu tượng là gì? Cho ví dụ minh họa.
9. Nêu khái niệm constructor và destructor. Phân biệt các kiểu constructor.

Dạng câu 2 (Thiết kế lớp đơn giản)

1. Xây dựng lớp Thời gian (giờ, phút, giây). Định nghĩa các phép toán:
 - ++ để tăng thời gian thêm 1 giây.
 - >> và << để nhập, xuất dữ liệu thời gian.
2. Xây dựng class `IntArr` để hàm main hoạt động đúng như mong đợi.

```
class IntArr {  
private:  
    int count; // tổng số lượng phần tử có trong values  
    int* values; // mảng các số nguyên đang có trong đối tượng hiện tại  
public:  
    /* Sinh viên bổ sung đầy đủ các thành phần cần thiết để hàm main hoạt động  
    như mong đợi */  
};  
int main() {  
    IntArr l1; // tạo mảng không chứa bất kì phần tử nào  
    IntArr l2(3, 2); /* tạo một mảng với 3 phần tử, tất cả phần tử đều có giá
```

trị là 2 */

IntArr l3(2); // tạo một mảng với 2 phần tử, tất cả phần tử đều có giá trị là 0

IntArr l4 = l2.concat(l3); /* tạo ra một IntArr mới có nội dung là kết quả của việc nối các phần tử l3 vào cuối các phần tử của l2 theo thứ tự */
l2.push(3); // thêm số 3 vào cuối danh sách trong đối tượng l2
cin >> l2; /* Xóa các giá trị hiện có trong l2 và cho phép người dùng nhập số lượng phần tử mới và giá trị các phần tử mới vào l2 (cần xóa các vùng nhớ không sử dụng nếu có) */

cout << l2; // in ra các số nguyên có trong danh sách

/* Khi vượt quá phạm vi sử dụng cần hủy tất cả các vùng nhớ được cấp phát cho các values của IntArr */

return 0;

}

3. Hãy định nghĩa lớp **Date** thích hợp để chương trình dưới đây không bị lỗi biên dịch và chạy đúng. Lưu ý rằng không được chỉnh sửa hàm main và sinh viên cần viết cả các lệnh #include thích hợp.

```
int main()
{
    Date ng1; // ng1 sẽ có giá trị là ngày 1 tháng 1 năm 1
    Date ng2(2017, 1); // ng2 sẽ có giá trị là ngày 1 tháng 1 năm 2017
    Date ng3(2017, 1, 7); // ng3 sẽ có giá trị là ngày 7 tháng 1 năm 2017
    cin >> ng1;
    ng1++;
    cout << ng1;
    if (ng1 < ng2)
        cout << "Ngày 1 trước ngày 2" << endl;
    else
        cout << "Ngày 1 không trước ngày 2" << endl;
    return 0;
}
```

Dạng câu 3 (Thiết kế lớp phức tạp)

(Đề thi lập trình hướng đối tượng cuối học kỳ I (2022-2023)): Một tổ chức chuyên trưng bày, mua bán các sản phẩm liên quan đến nghệ thuật đang muốn xây dựng một ứng dụng để quản lý các hoá đơn khi bán các sản phẩm. Mỗi lần bán sản phẩm thành công, cửa hàng sẽ lưu trữ các hoá đơn chứa thông tin sản phẩm liên quan. Mỗi hoá đơn sẽ có thông tin: mã hoá đơn, thông tin khách hàng, ngày lập hoá đơn, danh sách sản phẩm, tổng giá (tổng giá trị các sản phẩm trong đơn hàng). Tổ chức này hiện tại chỉ kinh doanh 2 loại sản phẩm: tranh ảnh và CD âm nhạc (tương lai có thể thay đổi sản phẩm kinh doanh khác). Mỗi sản phẩm sẽ có thông tin chung cần quản trị: mã số, tiêu đề, giá bán. Ngoài thông tin chung, các sản phẩm tranh ảnh cần thêm thông tin kích thước của bức tranh (chiều rộng, chiều cao), tên họa sĩ. Sản phẩm CD âm nhạc sẽ có thêm tên ca sĩ, tên đơn vị sản xuất. Mỗi khách hàng sẽ được lưu trữ các thông tin: mã khách hàng, tên khách hàng, số điện thoại. Áp dụng tư tưởng lập trình hướng đối tượng (có sử dụng kế thừa, đa hình), anh/chị hãy:

1. Thiết kế và vẽ sơ đồ lớp cho ứng dụng theo bài toán được mô tả
2. Cài đặt chi tiết theo sơ đồ lớp đã thiết kế và cũng như thành phần cần thiết khác để xây dựng chương trình thực hiện các tính năng sau:
 - a) Nhập và xuất danh sách các hoá đơn bán hàng
 - b) Tính tổng thu nhập của cửa hàng
 - c) Tìm các khách hàng mua nhiều nhất ở cửa hàng (dựa vào tổng giá trị các hoá đơn khách hàng đã mua).