

DANH SÁCH CÂU HỎI LÝ THUYẾT ÔN TẬP LẬP TRÌNH HƯỚNG ĐỐI TƯỢNG

Phần 1: 2 điểm

Câu 1: Nêu khái niệm về sự kế thừa và những ưu nhược điểm của kế thừa trong việc lập trình. Trường hợp nào thì có thể vi phạm tính kế thừa? Cho ví dụ minh họa.

Câu 2: Nêu khái niệm về trừu tượng hóa và những ưu nhược điểm của trừu tượng hóa trong việc lập trình. Cho ví dụ minh họa.

Câu 3: Nêu khái niệm về đa hình và những ưu nhược điểm của đa hình trong việc lập trình. Cho ví dụ minh họa.

Câu 4: Nêu khái niệm về đóng gói và những ưu nhược điểm của đóng gói trong việc lập trình. Trường hợp nào thì có thể vi phạm tính đóng gói? Cho ví dụ minh họa.

Câu 5:

- a. Phân biệt các phạm vi truy cập private, protected và public.
- b. Cho biết ý nghĩa và mục đích của các hàm get/set trong một lớp

Câu 6:

- a. Phân biệt khái niệm lớp và đối tượng trong lập trình hướng đối tượng.
- b. Trình bày khái niệm đa hình trong lập trình hướng đối tượng. Cho ví dụ minh họa.

Câu 7: Phân biệt các kiểu kế thừa private, protected, public

Câu 8: Trình bày các đặc điểm quan trọng của lập trình hướng đối tượng

Câu 9: Trình bày khái niệm của lớp cơ sở trừu tượng (abstract class). Lớp cơ sở trừu tượng được cài đặt trong C++ như thế nào?

Câu 10:

- a. Hàm thuần ảo là gì? Lớp trừu tượng là gì? Cho ví dụ minh họa.
- b. Hãy nêu các đặc điểm quan trọng của lập trình hướng đối tượng.

Câu 11: Phân biệt khái niệm overload (tải chồng) và override (ghi đè) trong lập trình hướng đối tượng.

Câu 12: Trình bày khái niệm Hàm bạn, lớp bạn. Ưu nhược điểm. Cho ví dụ minh họa.

Câu 13: Nêu vai trò của hàm tạo (Constructor), hàm hủy (destructor) trong định nghĩa lớp. Ưu nhược điểm của hai loại hàm này khi sử dụng trong kế thừa. Cho ví dụ minh họa.

Câu 14: Trình bày phép gán trong lập trình hướng đối tượng. Tại sao phải xây dựng phép gán cho lớp? Cho ví dụ minh họa.

Câu 15: Trình bày kỹ thuật nạp chồng (overloading) trong các tình huống không lập trình hướng đối tượng, trong lập trình hướng đối tượng và hàm bạn. Cho ví dụ minh họa.

Câu 16: Trình bày kỹ thuật đa năng hóa toán tử (nạp chồng toán tử) trong xây dựng một lớp. So sánh với cách xây dựng hàm tính toán tương ứng với toán tử. Cho ví dụ minh họa.

Câu 17: Trình bày kỹ thuật liên kết động. Cho ví dụ minh họa.

Câu 18: Trình bày toán tử (). Cho ví dụ minh họa.

Câu 19: Trình bày con trỏ hàm. Cho ví dụ minh họa.

Câu 20: So sánh hàm tạo sao chép và phép gán. Cho ví dụ minh họa.

PHẦN 2: 3 ĐIỂM:

Câu 1:

Xây dựng lớp đa thức bậc nhất để thể hiện các đa thức bậc nhất có dạng:

$$F(x) = ax + b \text{ (a luôn khác 0)}$$

Xây dựng các phương thức: (3 điểm)

- Phương thức cho phép xác định giá trị của đa thức ứng với $x=x_0$ (tính $F(x_0)$)
- Phương thức trả về nghiệm đa thức bậc 1 (nghĩa là $F(x)=0$)
- Phép toán cộng (operator +) để cộng hai đa thức bậc nhất

Câu 2. Xây dựng lớp Thời gian (giờ, phút giây). Định nghĩa các phép toán

++ để tăng thời gian thêm 1 giây. Xây dựng toán tử >> và << để nhập, xuất dữ liệu thời gian

Câu 3:

Xây dựng lớp đơn thức có bậc trong khoảng từ 1 đến 100.

$$F(x) = ax^n \text{ (a luôn khác 0, } 1 \leq n \leq 100 \text{)}$$

Xây dựng các phương thức: (3 điểm)

- Phương thức cho phép xác định giá trị của đơn thức ứng với $x=x_0$ (tính $F(x_0)$)
- Phương thức tính nghiệm đơn thức khi $F(x) = b$ ($b \in \mathbb{R}$)
- Phép toán cộng (operator +) để cộng hai đơn thức

Câu 4 (3 điểm):

Cho đoạn chương trình tính toán với lớp đối tượng ngày tháng năm (cNgày) như sau:

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

Hãy định nghĩa lớp cNgày thích hợp để chương trình 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

Câu 5:

Cho lớp Phân số (CPhanSo). Hãy khai báo và định nghĩa các phương thức cần thiết để các đối tượng thuộc lớp CPhanSo có thể thực hiện được các câu lệnh sau:

CPhanSo a(5, 3);

CPhanSo b, c, kq;

```

cin>>b>>c;
kq = a + b + 5 + c;
cout<<"Kết quả là: "<<kq;
if ( a == b )
    cout<<"Phan so a bang phan so b"<<endl;

```

Câu 6:

Định nghĩa lớp CDate biểu diễn khái niệm ngày, tháng, năm (0.5 đ) với phép toán ++ (thêm bớt một ngày) theo dạng prefix ++a và postfix a++ (1 đ). Phép toán <<, >> để xuất, nhập dữ liệu loại CDate (1 đ) .

Câu 7: (bài tập tuần 2)

Thiết lập lớp PhanSo để biểu diễn **khái niệm phân số** với hai thành phần dữ liệu **tử số, mẫu số** và các hàm thành phần **cộng, trừ, nhân, chia** hai phân số, các hàm thành phần **xuất, nhập, định giá trị cho phân số**. Viết chương trình cho phép nhập vào hai phân số, in ra kết quả các phép toán cộng, trừ, nhân, chia hai phân số kể trên.

Câu 8: (bài tập tuần 2)

Xây dựng lớp biểu diễn **khái niệm số phức** với hai thành phần dữ liệu **thực, ảo** và các hàm thành phần **xuất, nhập, định giá trị cho số phức, cộng, trừ, nhân, chia hai số phức**. Viết chương trình cho phép nhập vào hai số phức, in ra kết quả các phép toán cộng, trừ, nhân, chia hai số phức kể trên.

Ví dụ: Cho hai số phức $A(a_1, a_2)$, $B(b_1, b_2)$

- $A + B = (a_1 + b_1, a_2 + b_2)$
- $A - B = (a_1 - b_1, a_2 - b_2)$
- $A * B = (a_1 * b_1 - a_2 * b_2, a_1 * b_2 + a_2 * b_1)$
- $A / B = \left(\frac{a_1 * b_1 + a_2 * b_2}{b_1^2 + b_2^2}, \frac{b_1 * a_2 - a_1 * b_2}{b_1^2 + b_2^2} \right)$

Câu 9: (bài tập tuần 2)

Xây dựng lớp Candidate (Thí sinh) gồm các thuộc tính: mã, tên, ngày tháng năm sinh, điểm thi Toán, Văn, Anh và các phương thức cần thiết.

Xây dựng lớp TestCandidate để kiểm tra lớp trên:

- Nhập vào n thí sinh (n do người dùng nhập)
- In ra thông tin về các thí sinh có tổng điểm lớn hơn 15

Các dạng bài trong tuần 1 đến tuần 4