**1. Cho biết giá trị của a và ::a trong đoạn chương trình sau:**

Inta=5;  
intmain(){  
 inta=10,b=20;  
 ::a+=b+++++a;  
        cout<<"\n Gia tri a = "<<a<<"\n ::a = "<<::a;

10 30

**11 36**

10 36

11 30

**2. Thành phần private của lớp là thành phần:**

Cho phép truy xuất từ bên ngoài lớp cũng như từ các phương thức của lớp.

Cho phép truy xuất từ bên ngoài lớp.

**Không cho phép truy xuất từ bên ngoài lớp cũng như từ các phương thức của lớp.**

Không cho phép truy xuất từ bên ngoài lớp, chỉ có các phương thức bên trong lớp mới có thể truy xuất được.

**3. Chọn phương án đúng nhất:**

Giả sử có lớp C (class C)

Chỉ có thể khai báo đối tượng thuộc lớp C, không thể khai báo con trỏ hay tham chiếu kiểu lớp C.

**Có thể khai báo đối tượng thuộc lớp C, con trỏ kiểu lớp C, tham chiếu kiểu lớp C.**

Có thể khai báo đối tượng thuộc lớp C, tham chiếu kiểu lớp C.

Có thể khai báo đối tượng thuộc lớp C, con trỏ kiểu lớp C.

**4. Yếu tố nào sau đây có thể được truy cập bởi một hàm thành phần của lớp?...**

Yếu tố nào sau đây có thể được truy cập bởi một hàm thành phần của lớp?

Các thành phần công khai của lớp.

**Tất cả các thành phần của lớp.**

Các thành phần riêng tư của lớp.

Đối tượng của lớp.

**5. Khái niệm nào sau đây là không chính xác?**

**Hàm thiết lập bạn.**

Hàm thiết lập sao chép.

Hàm thiết lập có tham số.

Hàm thiết lập ngầm định.

**6. Cho biết kết quả của chương trình sau?**

#include <iostream>  
using namespace std;  
class Test {  
    public: int n;  
    private: void Display(){ cout<<n; }  
    public: Test(){ n=5; }  
};  
int main() {  
    Test t;  
    t.Display();  
    return 0;}

**Chương trình lỗi.**

5

Chương trình chạy nhưng không hiển thị nội dung gì lên màn hình.

10

**7. Cho khai báo sau:**

class SinhVien{  
    private:  
        int Masv;  
        char Hoten[40];  
        char Ngaysinh;  
        int Gioitinh;  
    ...  
};

Để khai báo một mảng gồm 50 đối tượng SinhVien, sử dụng khai báo nào sau đây:

Không thể khai báo mảng các đối tượng.

**SinhVien ds[10].**

Sinh\_vien \*ds[10].

Sinh\_vien ds(10).

**8. Cho biết kết quả xuất ra màn hình khi thực thi chương trình sau:**

#include <iostream>  
using namespace std;  
class Teacher {  
    public:  
        static int n;  
    public :   
        Teacher (){ cout<<" "<<n++ ; }  
};  
int Teacher::n = 0;  
int main() {  
    Teacher t1;  
    Teacher t2;  
    Teacher t3;  
    cout<<" "<<t1.n ;  
    return 0;  
}

0 1 2 2

1 2 3 1

**0 1 2 3**

1 2 3 4

**9. Chồng hàm là:**

Khả năng các hàm có thể trùng tên nhau nhưng khác nhau về kiểu dữ liệu trả về hoặc kiểu dữ liệu của tham số hoặc số lượng tham số. Chỉ có thể chồng hàm thành phần của lớp.

Cả ba phương án còn lại đều đúng.

**Khả năng các hàm có thể trùng tên nhau nhưng khác nhau về kiểu dữ liệu trả về hoặc kiểu dữ liệu của tham số hoặc số lượng tham số.**

Khả năng các hàm có thể trùng tên nhau, trùng cả về kiểu dữ liệu trả về, kiểu dữ liệu của tham số, số lượng tham số.

**10. Khi xây dựng lớp, nếu trong lớp có thành phần thuộc tính là kiểu con trỏ thì b...**

Khi xây dựng lớp, nếu trong lớp có thành phần thuộc tính là kiểu con trỏ thì bắt buộc phải:

Không bắt buộc phải xây dựng gì hết.

Xây dựng phương thức xuất.

Xây dựng phương thức nhập

Xây dựng hàm hủy bỏ.

**11. Sự trừu tượng là:**

Abstraction

Inheritance

Encapsulation

Polymorphism

**12. Nhận xét về sự khác biệt của tham chiếu so với con trỏ:**

Tham chiếu không nhận giá trị null

.**Tham chiếu không nhận giá trị null, tham chiếu không thể thay đổi sau khi khởi tạo, không cần toán tử để giải phóng vùng nhớ.**

Tham chiếu không cần toán tử để giải phóng vùng nhớ.

Tham chiếu không thể thay đổi sau khởi tạo

**13. Thành phần nào sau đây không thể khai báo bạn?**

Toán tử

**Đối tượng**

Lớp

Hàm

**14. Trong khai báo lớp Dathuc (Đa thức) sau đây:**

class Dathuc{  
    private:  
        int n;  
        float \*v;   
    public:   
};

Những hàm nào buộc phải xây dựng mà không sử dụng được hàm ngầm định:

**Hàm thiết lập, Hàm hủy bỏ, Hàm thiết lập sao chép.**

Hàm thiết lập, Hàm thiết lập sao chép.

Hàm thiết lập sao chép, Hàm hủy bỏ.

Hàm thiết lập, Hàm hủy bỏ.

**15. Phát biểu nào sau đây là đúng?**

Một lớp là một thể hiện của kiểu dữ liệu có trong lớp.

Một lớp là một thể hiện của các đối tượng.

Đối tượng là một thể hiện của lớp.

**Một đối tượng là một thể hiện của kiểu dữ liệu có trong lớp.**

**16. Mô đun hóa có nghĩa là:**

Ẩn một phần của chương trình.

Đóng gọi mọi thứ vào một khối.

Nạp chồng các phần của chương trình.

**Chia chương trình lớn thành các chương trình nhỏ, càng độc lập càng tốt.**

**17. Nhận xét tính đúng đắn của hai phát biểu sau: (1) Tất cả các toán tử của C++...**

Nhận xét tính đúng đắn của hai phát biểu sau: (1) Tất cả các toán tử của C++ đều có thể nạp chồng; (2) Có thể thay ý nghĩa nguyên bản của một toán tử trong C++.

Chỉ 2 đúng

**Cả 1 và 2 đều sai**

Cả 1 và 2 đều đúng

Chỉ 1 đúng

**18. Cho biết kết quả a, r, t trong đoạn chương trình sau**

int main() {

            int a=5; int &r=a; int &t=a;

            a+=r+t;

            cout<<"\n a = "<<a<<"    r = "<<r<<"    t = "<<t;

}

15 10 15

**15 15 15**

15 15 10

10 15 15

**19. Câu 25: Thành phần public của lớp là thành phần:**

Chỉ cho phép truy xuất từ bên ngoài lớp.

Cho phép truy xuất từ bên trong lớp nhưng không cho phép truy xuất từ bên ngoài lớp.

Chỉ cho phép truy xuất từ bên trong lớp.

**Cho phép truy xuất từ bên trong và bên ngoài lớp.**

**20. Cho biết kết quả xuất ra màn hình sau khi thực thi chương trình sau:**

include <iostream>

using namespace std;

class ABC{

            public:

                        ABC(){cout<<"\nHam thiet lap";   }

                        ~ABC(){cout<<"\nHam huy bo";   }

};

int main()

{

            ABC x;

}

Ham huy bo

Không xuất hiện thông tin gì

**Ham thiet lap  
 Ham huy bo**

Ham thiet lap

**21. Khi khai báo lớp có thể:**

Chỉ sử dụng một lần từ khóa private nhưng có thể nhiều lần từ khóa public

Chỉ sử dụng một lần từ khóa public nhưng có thể nhiều lần từ khóa private

Chỉ sử dụng một lần từ khóa private và public

**Sử dụng nhiều lần các từ khóa private, public**

**22. Trong khai báo lớp:**

**Có thể khai báo nhiều hàm thiết lập nhưng chỉ một hàm hủy bỏ.**

Có thể khai báo nhiều hàm thiết lập và nhiều hàm hủy bỏ.

Chỉ được khai báo một hàm thiết lập và một hàm hủy bỏ.

Chỉ được khai báo một hàm thiết lập nhưng nhiều hàm hủy bỏ.

**23. Cho lớp Diem (Điểm) như sau:**

class Diem{  
    private:  
        int x,y;   
public:   
    void nhap(){  
    }  
    friend khoang\_cach(Diem D1, Diem D2);  
};

Hàm thành phần của một lớp khác bạn với lớp Diem.

Hàm tự do là bạn của lớp Diem.

Hàm thông thường

Phương thức của lớp Diem.

**24. Cho biết kết quả của đoạn chương trình sau?**

int f(int &a){

            ++a; return a;

}

int main() {

            int x=5;

            cout<<f(x)<<"   "<<x;

}

5 5

Không cho kết quả

6 6

**6 5**

**25. Cho biết phát biểu nào sau đây là sai? Hàm thành phần của lớp có thể:**

Khai báo theo thứ tự tùy ý.

Khai báo trong lớp và định nghĩa ngoài lớp.

Khai báo và định nghĩa trong lớp.

**Khai báo và định nghĩa ngoài lớp.**

**26. Cho chương trình sau:**

#include <iostream>  
using namespace std;  
class Songuyen{  
    private:  
        int x;   
public:   
    Songuyen(int ax=0){x=ax;  
    }  
    Songuyen operator+(Songuyen a){  
    return x+a.x;}  
};

int main(){  
Songuyen m,n,t;  
}  
Câu lệnh nào sau đây bị lỗi?

t=m+3;

**t=3+n;**

t=m+n;

t=m.operator+(n);

**27. Trong một chương trình có thể xây dựng tối đa bao nhiêu lớp?**

10 lớp.

1 lớp.

3 lớp.

**Vô số tuỳ theo bộ nhớ.**

**28. Kiểu nào sau đây là một kiểu dữ liệu trừu tượng?**

**Kiểu lớp.**

Kiểu char.

Kiểu float.

Kiểu int.

**29. C++ là:**

Ngôn ngữ lập trình hướng thủ tục.

Ngôn ngữ lập trình hàm.

**Ngôn ngữ lập trình lai cả hướng thủ tục và cả hướng đối tượng.**

****Ngôn ngữ lập trình hướng đối tượng.

**30. Sự đóng gói là:**

Inheritance

Polymorphism

**Encapsulation**

Abstraction

**31. Cho biết kết quả của chương trình sau?**

#include <iostream>

using namespace std;

int main(){

    int a = 9;

    int &aref = a;

    a++;

    cout << "Gia tri a = " << aref;

    return 0;

}

**10**

Chương trình lỗi

12

9

**32. Cho chương trình sau:**

#include <iostream>

using namespace std;

class A{

            private:

                        int x;

            public:

                        void nhap();

};

int main(){

            A obj;

            obj.nhap();

}

Không đươc phép truy cập đến phương thức nhap()

**Không định nghĩa nội dung của phương thức nhap()**

Không khai báo biến x

Không được khai báo phương thức nhap() ở quyền public

**33. Thuộc tính của lớp là:**

**Dữ liệu trình bày các đặc điểm của một đối tượng.**

Yếu tố quyết định quyền truy nhập các thành phần của đối tượng.

Liên quan tới những thứ mà đối tượng có thể làm. Một phương thức đáp ứng một chức năng tác động lên dữ liệu của đối tượng.

Những chức năng của đối tượng.

**34. Cho biết kết quả xuất ra màn hình khi thực thi chương trình sau:**

#include <iostream>  
using namespace std;  
class Point{  
    private:  
        int xVal, yVal;   
public:   
    void Print(){  
        cout<<"("<<xVal<<","<<yVal<<")";  
    }  
    Point(int ox=0, int oy=0){xVal = ox; yVal = oy;}  
    friend Point operator + (Point, Point);  
};  
Point operator + (Point p1, Point p2)  
{  
    return Point(p1.xVal+p2.xVal, p1.yVal+p2.yVal);  
}  
int main(){  
    Point p1(3,4);  
    Point p2(5);   
    Point p = p1+p2;   
    p.Print();  
    return 0;  
}

**(8,4)**

Lỗi biên dịch

(8,10)

Lỗi thực thi

**35. Cho khai báo lớp Time (Thời gian) như sau:**

class Time{  
    private:  
        int hour, min, sec;   
    public:   
        void add(){  
    };  
};

Khi định nghĩa phương thức add() bên ngoài lớp ta phải viết là:

void Time:add(){//Nội dung}

void add(){//Nội dung}

void add();{//Nội dung}

**void Time::add(){//Nội dung}**

**36. Cho biết phát biểu nào sau đây không đúng với quy định của hàm thiết lập?**

Hàm thiết lập không có giá trị trả về

**Hàm thiết lập có quyền private**

Hàm thiết lập có tên trùng với tên lớp

Có thể định nghĩa nhiều hàm thiết lập trong một lớp.

**37. OOP là viết tắt của:**

Open Object Programming.

Object Open Programming.

**Object Oriented Programming.**

Object Oriented Proccessing.

**38. Hàm nào đúng trong các hàm sau?**

void f(int a=1, int b, int c)

void f(int a=1, int b, int c=3)

**void f(int a=1, int b=2, int c=3)**

void f(int a=1, int b=2, int c)

**39. Toán tử nào sau đây được dùng với đối tượng cin?**

<

**>>**

>

<<

**40. Hàm thiết lập được gọi như thế nào?**

**Không cần phải gọi tới hàm thiết lập vì ngay khi khai báo, đối tượng sẽ tự gọi tới hàm thiết lập.**

****Tất cả đều sai.

Gọi bằng cách: Tên\_đối\_tượng.Tên\_hàm\_thiết\_lập

Gọi bằng cách: Tên\_lớp.Tên\_hàm\_thiết\_lập