

C++ object-oriented questions

What is a modifier?

A modifier, also called a modifying function is a member function that changes the value of at least one data member. In other words, an operation that modifies the state of an object. Modifiers are also known as ‘mutators’. Example: The function mod is a modifier in the following code snippet:

```
class test
{
    int x,y;
public:
    test()
    {
        x=0; y=0;
    }
    void mod()
    {
        x=10;
        y=15;
    }
};
```

What is an accessor?

An accessor is a class operation that does not modify the state of an object. The accessor functions need to be declared as const operations

Differentiate between a template class and class template.

Template class: A generic definition or a parameterized class not instantiated until the client provides the needed information. It’s jargon for plain templates. Class template: A class template specifies how individual classes can be constructed much like the way a class specifies how individual objects can be constructed. It’s jargon for plain classes.

When does a name clash occur?

A name clash occurs when a name is defined in more than one place. For example., two different class libraries could give two different classes the same name. If you try to use many class libraries at the same time, there is a fair chance that you will be unable to compile or link the program because of name clashes.

Define namespace.

It is a feature in C++ to minimize name collisions in the global name space. This namespace keyword assigns a distinct name to a library that allows other libraries to use the same identifier names without creating any name collisions.

Furthermore, the compiler uses the namespace signature for differentiating the definitions.

What is the use of ‘using’ declaration.

A using declaration makes it possible to use a name from a namespace without the scope operator.

What is an Iterator class?

A class that is used to traverse through the objects maintained by a container class. There are five categories of iterators: input iterators, output iterators, forward iterators, bidirectional iterators, random access. An iterator is an entity that gives access to the contents of a container object without violating encapsulation constraints. Access to the contents is granted on a one-at-a-time basis in order. The order can be storage order (as in lists and queues) or some arbitrary order (as in array indices) or according to some ordering relation (as in an ordered binary tree). The iterator is a construct, which provides an interface that, when called, yields either the next element in the container, or some value denoting the fact that there are no more elements to examine. Iterators hide the details of access to and update of the elements of a container class. The simplest and safest iterators are those that permit read-only access to the contents of a container class.

List out some of the OODBMS available.

GEMSTONE/OPAL

of Gemstone systems, ONTOS of Ontos, Objectivity of Objectivity Inc, Versant of Versant object technology, Object store of Object Design, ARDENT of ARDENT software, POET of POET software.

List out some of the object-oriented methodologies.

Object Oriented Development (OOD) (Booch 1991,1994), Object Oriented Analysis and Design (OOA/D) (Coad and Yourdon 1991), Object Modelling Techniques (OMT) (Rumbaugh 1991), Object Oriented Software Engineering (Objectory) (Jacobson 1992), Object Oriented Analysis (OOA) (Shlaer and Mellor 1992), The Fusion Method (Coleman 1991).

What is an incomplete type?

Incomplete types refers to pointers in which there is non availability of the implementation of the referenced location or it points to some location whose value is not available for modification.

```
int *i=0x400 // i points to address 400
*i=0; //set the value of memory location
pointed by i.
```

Incomplete types are otherwise called uninitialized pointers.

What is a dangling pointer?

A dangling pointer arises when you use the address of an object after its lifetime is over. This may occur in situations like returning addresses of the automatic variables from a function or using the address of the memory block after it is freed. The following code snippet shows this:

```
class Sample
{
public:
    int *ptr;
    Sample(int i)
    {
        ptr = new int(i);
    }
}
```

```

        ~Sample()
        {
            delete ptr;
        }
        void PrintVal()
        {
            cout << "The value is " << *ptr;
        }
};

void SomeFunc(Sample x)
{
    cout << "Say i am in someFunc " << endl;
}

int main()
{
    Sample s1 = 10;
    SomeFunc(s1);
    s1.PrintVal();
}

```

In the above example when PrintVal() function is called it is called by the pointer that has been freed by the destructor in SomeFunc.

Differentiate between the message and method.

Message:

- Objects communicate by sending messages to each other.
- A message is sent to invoke a method.

Method

- Provides response to a message.
- It is an implementation of an operation.

What is an adaptor class or Wrapper class?

A class that has no functionality of its own. Its member functions hide the use of a third party software component or an object with the non-compatible interface or a non-object-oriented implementation.

What is a Null object?

It is an object of some class whose purpose is to indicate that a real object of that class does not exist. One common use for a null object is a return value from a member function that is supposed to return an object with some specified properties but cannot find such an object.

What is class invariant?

A class invariant is a condition that defines all valid states for an object. It is a logical condition to ensure the correct working of a class. Class invariants must hold

when an object is created, and they must be preserved under all operations of the class. In particular all class invariants are both preconditions and post-conditions for all operations or member functions of the class.

What do you mean by Stack unwinding?

It is a process during exception handling when the destructor is called for all local objects between the place where the exception was thrown and where it is caught.

Define precondition and post-condition to a member function.

Precondition: A precondition is a condition that must be true on entry to a member function. A class is used correctly if preconditions are never false. An operation is not responsible for doing anything sensible if its precondition fails to hold. For example, the interface invariants of stack class say nothing about pushing yet another element

on a stack that is already full. We say that `isful()` is a precondition of the push operation. **Post-condition:** A post-condition is a condition that must be true on exit from a member function if the precondition was valid on entry to that function. A class is implemented correctly if post-conditions are never false. For example, after pushing an element on the stack, we know that `isempty()` must necessarily hold. This is a post-condition of the push operation.

What are the conditions that have to be met for a condition to be an invariant of the class?

- The condition should hold at the end of every constructor.
- The condition should hold at the end of every mutator (non-const) operation.

What are proxy objects?

Objects that stand for other objects are called proxy objects or surrogates.

```
template <class T="">
class Array2D
{
    public:
    class Array1D
    {
        public:
        T& operator[] (int index);
        const T& operator[] (int index) const;
    };

    Array1D operator[] (int index);
    const Array1D operator[] (int index) const;
};
```

The following then becomes legal:

```
Array2D<float> data(10,20);
cout<<data[3][6];    // fine
```

Here `data[3]` yields an `Array1D` object

and the operator `[]` invocation on that object yields the float in position(3,6) of the original two dimensional array. Clients of the `Array2D` class need not be aware of the presence of the `Array1D` class.

Objects of this latter class stand for one-dimensional array objects that, conceptually, do not exist for clients of Array2D. Such clients program as if they were using real, live, two-dimensional arrays. Each Array1D object stands for a one-dimensional array that is absent from a conceptual model used by the clients of Array2D. In the above example, Array1D is a proxy class. Its instances stand for one-dimensional arrays that, conceptually, do not exist.

Name some pure object oriented languages. Smalltalk, Java, Eiffel, Sather.

Name the operators that cannot be overloaded. sizeof, ., .*, .->, ::, ?:

Salam in the comments notes that -> **can** be overloaded.

What is a node class?

A node class is a class that,

- relies on the base class for services and implementation,
- provides a wider interface to the users than its base class,
- relies primarily on virtual functions in its public interface
- depends on all its direct and indirect base class
- can be understood only in the context of the base class
- can be used as base for further derivation
- can be used to create objects.

A node class is a class that has added new services or functionality beyond the services inherited from its base class.

What is an orthogonal base class?

If two base classes have no overlapping methods or data they are said to be independent of, or orthogonal to each other. Orthogonal in the sense means that two classes operate in different dimensions and do not interfere with each other in any way. The same derived class may inherit such classes with no difficulty.

What is a container class? What are the types of container classes?

A container class is a class that is used to hold objects in memory or external storage. A container class acts as a generic holder. A container class has a predefined behavior and a well-known interface. A container class is a supporting class whose purpose is to hide the topology used for maintaining the list of objects in memory. When a container class contains a group of mixed objects, the container is called a heterogeneous container; when the container is holding a group of objects that are all the same, the container is called a homogeneous container.

[COM interview questions](#)

What is IUnknown? What methods are provided by IUnknown?

It is a generally good idea to have an answer for this question if you claim you know COM in your resume. Otherwise, you may consider your interview failed at this point. IUnknown is the base interface of COM. All other interfaces must derive directly or indirectly from IUnknown. There are three methods in that interface: AddRef, Release and QueryInterface.

What are the purposes of AddRef, Release and QueryInterface functions?

AddRef increments reference count of the object, Release decrements reference counter of the object and QueryInterface obtains a pointer to the requested interface.

What should QueryInterface functions do if requested object was not found?

Return E_NOINTERFACE and nullify its out parameter.

How can would you create an instance of the object in COM?

Well, it all depends on your project. Start your answer from CoCreateInstance or CoCreateInstanceEx, explain the difference between them. If interviewer is still not satisfied, you'll have to explain the whole kitchen behind the scenes, including a difference between local server and inproc server, meaning and mechanism of class factory, etc. You may also mention other methods of object creation like CoGetInstanceFromFile, but discussion will likely turn to discussion of monikers then.

What happens when client calls CoCreateInstance?

Again, all depends on the level of detail and expertise of interviewer. Start with simple explanation of class object and class factory mechanism. Further details would depend on a specific situation.

What the limitations of CoCreateInstance?

Well, the major problems with CoCreateInstance is that it is only able to create one object and only on local system. To create a remote object or to get several objects, based on single CLSID, at the same time, one should use CoCreateInstanceEx.

What is aggregation? How can we get an interface of the aggregated object?

Aggregation is the reuse mechanism, in which the outer object exposes interfaces from the inner object as if they were implemented on the outer object itself. This is useful when the outer object would always delegate every call to one of its interfaces to the same interface in the inner object. Aggregation is actually a specialized case of containment/delegation, and is available as a convenience to avoid extra implementation overhead in the outer object in these cases. We can get a pointer to the inner interface, calling QueryInterface of the outer object with IID of the inner interface.

C is aggregated by B, which in turn aggregated by A. Our client requested C.

What will happen?

QueryInterface to A will delegate request to B which, in turn, will delegate request for the interface to C. This pointer will be returned to the client.

What is a moniker ?

An object that implements the IMoniker interface. A moniker acts as a name that uniquely identifies a COM object. In the same way that a path identifies a file in the file system, a moniker identifies a COM object in the directory namespace.

What's the difference, if any, between OLE and COM?

OLE is build on top of COM. The question is not strict, because OLE was built over COM for years, while COM as a technology was presented by Microsoft a few years ago. You may mention also that COM is a specification, while OLE is a particular implementation of this specification, which in today's world is not

exactly true as well, because what people call COM today is likely implementation of COM spec by Microsoft.

What's the difference between COM and DCOM?

Again, the question does not require strict answer. Any DCOM object is yet a COM object (DCOM **extends** COM) and any COM object may participate in DCOM transactions. DCOM introduced several improvements/optimizations for distributed environment, such as MULTI_QI (multiple QueryInterface()), security contexts etc. DCOM demonstrated importance of surrogate process (you cannot run in-proc server on a remote machine. You need a surrogate process to do that.) DCOM introduced a *load balancing*.

What is a dual interface?

Dual interface is one that supports both - IDispatch interface and vtbl-based interface. Therefore, it might be used in scripting environment like VBScript and yet to use power and speed of vtbl-based interface for non-scripting environment. Discussion then may easily transform into analyzing of dual interface problems - be prepared to this twist.

Can you have two dual interfaces in one class?

Yes. You **may** have two dual interfaces in one class, but only one of them may be default. The bottom line is that you cannot work with two dual interfaces at the same time due to nature of dual interface! To support two dual interfaces in VB you would write something like:

```
dim d1 as IDualInterface1
dim d2 as IDualInterface2
set d1 = new MyClassWithTwoDuals
set d2 = d1
```

In ATL's class you would have to use macro COM_INTERFACE_ENTRY2(IDispatch, IDualInterface1), to distinguish between different dual interfaces.

What is marshalling by value?

Some objects can essentially be considered static: regardless of which methods are called, the state of the object does not change. Instead of accessing such an object remotely, it is possible to *copy the static state of the object and create a new object with the same state information* on the caller side. The caller won't be able to notice the difference, but calls will be more efficient because they do not involve network round trips. This is called "marshaling by value".

What is a multi-threaded apartment (MTA)? Single-threaded apartment (STA)?

This is pretty difficult question to describe shortly. Anyway, apartments were introduced by Microsoft in NT 3.51 and late Windows 95 to isolate the problem of running legacy non-thread safe code into multithreaded environment. Each thread was "encapsulated" into so called single-threaded apartment. The reason to create an object in apartment is thread-safety. *COM* is responsible synchronize access to the object even if the object inside of the apartment is not thread-safe. Multithreaded apartments (MTA, or free threading apartment) were introduced in NT 4.0. Idea behind MTA is that COM is not responsible to synchronize object calls between threads. In MTA the developer is responsible for that. See

“Professional DCOM Programming” of Dr. Grimes et al. or “Essential COM” of Don Box for the further discussion on this topic.

Let’s assume we have object B and aggregated object C (in-proc server), created by B. Can you access any interface of B from C? What’s the difference between aggregated and contained objects?

Yes, you can. This is fundamental postulate of COM: “If you can get there from here, you can get there from anywhere”, i.e. QI’ing for IUnknown you may proceed and to get a pointer to any other interface, supported by the object. Aggregated object exposes its interface directly, without visible intervention of the object container. Contained object is created within the object container and its interfaces might be altered or filtered by the object container.

What is ROT ? GIT ? Count pros and cons of both.

By definition, *running object table* (ROT) is a globally accessible table on each computer that keeps track of all COM objects in the running state that can be identified by a moniker. Moniker providers register an object in the table, which increments the object’s reference count. Before the object can be destroyed, its moniker must be released from the table. *Global Interface Table* (GIT) allows any apartment (either single- or multi-threaded) in a process **to get access to an interface** implemented on an object **in any other apartment** in the process.

If you have an object with two interfaces, can you custom marshal one of them?

No! The decision to use custom marshaling is an all-or-nothing decision; an object has to custom marshal all its interfaces or none of them.

Is there a way to register in-proc server without regsvr32.exe?

Yes. Call DllRegisterServer() from the client. Do not forget to call DllUnregisterServer from the same client. You may also use Registrar object for the same purpose or use direct manipulation of the windows registry.

What is VARIANT? Why and where would you use it?

VARIANT is a huge union containing automation type. This allows easy conversion of one automation type to another. The biggest disadvantage of VARIANT is size of the union.

How can you guarantee that only remote server is ever created by a client?

Create an object (call CoCreateObjectEx()) with CLSCTX_REMOTE_SERVER flag.

What is an IDL?

IDL stands for **I**nterface **D**efinition **L**anguage. IDL is the language to describe COM interfaces.

What is In-proc?

In-proc is in-process COM object, i.e. COM object that implemented as DLL and supposed to be hosted by a container. When you have to instantiate the in-proc object remotely, you may use DLLHost.exe application that was design specially for this purpose.

What is OLE?

OLE is an object and embedding first implementation of COM spec available from MS before COM was officially named COM.

Give examples of OLE usage.

The most famous examples are probably drag and drop and structured storage implementations.

What are 2 storage types for composite document? Storage and Stream.

Is .doc document a compound document? Is it a structured storage? Compound document is a document that contains information about other documents hosted in this document. All office documents *_may_* be compound documents, but may be not. Word documents from version 6.0 and up are stored as structured storage.

Explain which of the following declarations will compile and what will be constant - a pointer or the value pointed at:

- `const char *`
- `char const *`
- `char * const`

Note: Ask the candidate whether the first declaration is pointing to a string or a single character. Both explanations are correct, but if he says that it's a single character pointer, ask why a whole string is initialized as `char*` in C++. If he says this is a string declaration, ask him to declare a pointer to a single character. Competent candidates should not have problems pointing out why `const char*` can be both a character and a string declaration, incompetent ones will come up with invalid reasons.

You're given a simple code for the class BankCustomer. Write the following functions:

- Copy constructor
- `=` operator overload
- `==` operator overload
- `+` operator overload (customers' balances should be added up, as an example of joint account between husband and wife)

Note: Anyone confusing assignment and equality operators should be dismissed from the interview. The applicant might make a mistake of passing by value, not by reference. The candidate might also want to return a pointer, not a new object, from the addition operator. Slightly hint that you'd like the value to be changed outside the function, too, in the first case. Ask him whether the statement `customer3 = customer1 + customer2` would work in the second case.

What problems might the following macro bring to the application?

```
#define sq(x) x*x
```

Consider the following struct declarations:

```
struct A { A(){ cout << "A"; } };
struct B { B(){ cout << "B"; } };
struct C { C(){ cout << "C"; } };
struct D { D(){ cout << "D"; } };
struct E : D { E(){ cout << "E"; } };
struct F : A, B
{
    C c;
    D d;
    E e;
    F() : B(), A(), d(), c(), e() { cout << "F"; }
};
```

What constructors will be called when an instance of F is initialized? Produce the program output when this happens.

Anything wrong with this code?

```
T *p = new T[10];
delete p;
```

Note: Incorrect replies: “No, everything is correct”, “Only the first element of the array will be deleted”, “The entire array will be deleted, but only the first element destructor will be called”.

Anything wrong with this code?

```
T *p = 0;
delete p;
```

Note: Typical wrong answer: Yes, the program will crash in an attempt to delete a null pointer. The candidate does not [understand pointers](http://www.joelonsoftware.com/articles/fog0000000073.html) <http://www.joelonsoftware.com/articles/fog0000000073.html>. A **very smart** candidate will ask whether delete is overloaded for the class T.

Explain virtual inheritance. Draw the diagram explaining the initialization of the base class when virtual inheritance is used.

Note: Typical mistake for applicant is to draw an inheritance diagram, where a single base class is inherited with virtual methods. Explain to the candidate that this is not virtual inheritance. Ask them for [the classic definition of virtual inheritance](http://cplusplus.about.com/library/weekly/aa121502e.htm) <http://cplusplus.about.com/library/weekly/aa121502e.htm>. Such question might be too complex for a beginning or even intermediate developer, but any applicant with advanced C++ experience should be somewhat familiar with the concept, even though he'll probably say he'd avoid using it in a real project. Moreover, even the experienced developers, who know about virtual inheritance, cannot coherently explain the initialization process. If you find a candidate that knows both the concept and the initialization process well, he's hired.

What's potentially wrong with the following code?

```
long value;
//some stuff
value &= 0xFFFF;
```

Note: Hint to the candidate about the base platform they're developing for. If the person still doesn't find anything wrong with the code, they are not experienced with C++.

What does the following code do and why would anyone write something like that?

```
void send (int *to, int * from, int count)
{
    int n = (count + 7) / 8;
    switch ( count % 8)
    {
        case 0: do { *to++ = *from++;
        case 7: *to++ = *from++;
        case 6: *to++ = *from++;
        case 5: *to++ = *from++;
```

```

        case 4: *to++ = *from++;
        case 3: *to++ = *from++;
        case 2: *to++ = *from++;
        case 1: *to++ = *from++;
    } while ( --n > 0 );
}
}

```

In the H file you see the following declaration:

```

class Foo {
void Bar( void ) const ;
};

```

Tell me all you know about the Bar() function.

What will print out?

```

main()
{
    char *p1="name";
    char *p2;
    p2=(char*)malloc(20);
    memset (p2, 0, 20);
    while(*p2++ = *p1++);
    printf("%sn",p2);
}

```

Answer:empty string.

What will be printed as the result of the operation below:

```

main()
{
    int x=20,y=35;
    x=y++ + x++;
    y= ++y + ++x;
    printf("%d%dn",x,y);
}

```

Answer : 5794

What will be printed as the result of the operation below:

```

main()
{
    int x=5;
    printf("%d,%d,%dn",x,x< <2,x>>2);
}

```

Answer: 5,20,1

What will be printed as the result of the operation below:

```

#define swap(a,b) a=a+b;b=a-b;a=a-b;
void main()
{
    int x=5, y=10;
    swap (x,y);
    printf("%d %dn",x,y);
    swap2(x,y);
    printf("%d %dn",x,y);
}
int swap2(int a, int b)
{

```

```

        int temp;
        temp=a;
        b=a;
        a=temp;
        return 0;
    }

```

Answer: 10, 5
10, 5

What will be printed as the result of the operation below:

```

main()
{
    char *ptr = " Cisco Systems";
    *ptr++;
    printf("%sn",ptr);
    ptr++;
    printf("%sn",ptr);
}

```

Answer:Cisco Systems
isco systems

What will be printed as the result of the operation below:

```

main()
{
    char s1[]="Cisco";
    char s2[]="systems";
    printf("%s",s1);
}

```

Answer: Cisco

What will be printed as the result of the operation below:

```

main()
{
    char *p1;
    char *p2;
    p1=(char *)malloc(25);
    p2=(char *)malloc(25);
    strcpy(p1,"Cisco");
    strcpy(p2,"systems");
    strcat(p1,p2);
    printf("%s",p1);
}

```

Answer: Ciscosystems

The following variable is available in file1.c, who can access it?:

```

static int average;

```

Answer: all the functions in the file1.c can access the variable.

WHAT will be the result of the following code?

```

#define TRUE 0
//      somecode.
while(TRUE)

```

```

{
    //      somecode
}

```

Answer: This will not go into the loop as TRUE is defined as 0.
What will be printed as the result of the operation below:

```

int x;

int modifyvalue()
{
    return(x+=10);
}

int changevalue(int x)
{
    return(x+=1);
}

void main()
{
    int x=10;
    x++;
    changevalue(x);
    x++;
    modifyvalue();
    printf("First output:%dn",x);
    x++;
    changevalue(x);
    printf("Second output:%dn",x);
    modifyvalue();
    printf("Third output:%dn",x);
}

```

Answer: 12 , 13 , 13
What will be printed as the result of the operation below:

```

main()
{
    int x=10, y=15;
    x = x++;
    y = ++y;
    printf("%d %dn",x,y);
}

```

Answer: 11, 16
What will be printed as the result of the operation below:

```

main()
{
    int a=0;
    if(a==0) printf("Cisco Systemsn");
    printf("Cisco Systemsn");
}

```

Answer: Two lines with “Cisco Systems” will be printed.

[C++ code examples for job interviews](#)

Write a short code using C++ to print out all odd number from 1 to 100 using a for loop(Asked by Intacct.com people)

```
for( unsigned int i = 1; i < = 100; i++ )
    if( i & 0x00000001 )
        cout << i<<" ";
```

ISO layers and what layer is the IP operated from?(Asked by Cisco system people)
Application, Presentation, Session, Transport, Network, Data link and Physical. The IP is operated in the Network layer.

Q: Write a program that ask for user input from 5 to 9 then calculate the average(Asked by Cisco system people)

```
A.int main()
{
int MAX=4;
int total =0;
int average=0;
int numb;
cout<<"Please enter your input from 5 to 9";
cin>>numb;
if((numb <5)&&(numb>9))
cout<<"please re type your input";
else
for(i=0;i<=MAX; i++)
{
total = total + numb;
average= total /MAX;
}
cout<<"The average number is"<<average<<endl;
return 0;
}
```

Q: Can you be bale to identify between Straight- through and Cross- over cable wiring? and in what case do you use Straight- through and Cross-over? (Asked by Cisco system people)

A. Straight-through is type of wiring that is one to one connection Cross- over is type of wiring which those wires are got switched
We use Straight-through cable when we connect between NIC Adapter and Hub. Using Cross-over cable when connect between two NIC Adapters or sometime between two hubs.

Q: If you hear the CPU fan is running and the monitor power is still on, but you did not see any thing show up in the monitor screen. What would you do to find out what is going wrong? (Asked by WNI people)

A. I would use the ping command to check whether the machine is still alive(connect to the network) or it is dead.

C++ networking questions

In C++, what is the difference between method overloading and method overriding?

Overloading a method (or function) in C++ is the ability for functions of the same name to be defined as long as these methods have different signatures (different set of parameters). Method overriding is the ability of the inherited class rewriting the virtual method of the base class.

What methods can be overridden in Java?

In C++ terminology, all public methods in Java are virtual. Therefore, all Java methods can be overwritten in subclasses except those that are declared final, static, and private.

In C, what is the difference between a static variable and global variable?

A static variable declared outside of any function is accessible only to all the functions defined in the same file (as the static variable). However, a global variable can be accessed by any function (including the ones from different files).

In C, why is the void pointer useful? When would you use it?

The void pointer is useful because it is a generic pointer that any pointer can be cast into and back again without loss of information.

What are the defining traits of an object-oriented language?

The defining traits of an object-oriented language are:

- encapsulation
- inheritance
- polymorphism

What is pure virtual function?

A class is made abstract by declaring one or more of its virtual functions to be pure. A pure virtual function is one with an initializer of = 0 in its declaration

Q. Write a Struct Time where integer m, h, s are its members

struct Time

```
{  
int m;  
int h;  
int s;  
};
```

How do you traverse a Btree in Backward in-order?

Process the node in the right subtree

Process the root

Process the node in the left subtree

Q. What is the two main roles of Operating System?

As a resource manager

As a virtual machine

Q. In the derived class, which data member of the base class are visible?

In the public and protected sections.

[C++ algorithms interview questions from Microsoft and IBM](#)

Assume I have a linked list contains all of the alphabets from 'A' to 'Z'. I want to find the letter 'Q' in the list, how does you perform the search to find the 'Q'?

Answer: In a linked list, we only know about the header and other elements are invisible unless we go through the node one by one. Since we have go through every single node to find 'Q', the search time for a linked list is linear which is $O(N)$.

(From IBM) How do you think about your school?

: I enjoy studying in our school because we have many professors and instructors are from local companies. Their professions lead us more close to local industries.

(From IBM) What classes you have enjoyed the most during your school years?

Answer: I like the class I am taking this semester, which involves a group project that needs great amount of team efforts. I really enjoy work with a group of people because we can learn new materials mutually.

>

(From IBM) According to your group project you just mentioned, what's the responsibility for each member in your group?

Answer: We have five people in our group. So far we have two web servers set up; one will be the back up system and the other will be the main system. Our leader coordinates the schedule. Two members are working on the database and do the coding for the connection between database and Java serverlets. One member is working on the user browser interface. All members will assign some classes to work on and perform the final test at the end. We have group meeting every Saturday to ensure our schedule is on track.

Can you work under pressure?

Answer: I worked for Sega of America in the hardware development group three years ago. They were working on the next generation of gaming machine (which is the "Dreamcast" we seen today in the market). My duty was to ensure the quality of prototypes that just built from manufacture were ready for engineers to test. I managed to balance the schedules and pressures from school and work.

[C++ object-oriented interview questions](#)

1.How do you write a function that can reverse a linked-list? (Cisco System)

```
void reverselist(void)
{
if(head==0)
return;
if(head->next==0)
return;
if(head->next==tail)
{
```

```

head->next = 0;
tail->next = head;
}
else
{
node* pre = head;
node* cur = head->next;
node* curnext = cur->next;
head->next = 0;
cur->next = head;
for(; curnext!=0; )
{
cur->next = pre;
pre = cur;
cur = curnext;
curnext = curnext->next;
}
curnext->next = cur;
}
}

```

2. What is polymorphism?

Polymorphism is the idea that a base class can be inherited by several classes. A base class pointer can point to its child class and a base class array can store different child class objects.

3. How do you find out if a linked-list has an end? (i.e. the list is not a cycle)

You can find out by using 2 pointers. One of them goes 2 nodes each time. The second one goes at 1 nodes each time. If there is a cycle, the one that goes 2 nodes each time will eventually meet the one that goes slower. If that is the case, then you will know the linked-list is a cycle.

4. How can you tell what shell you are running on UNIX system?

You can do the Echo \$RANDOM. It will return a undefined variable if you are from the C-Shell, just a return prompt if you are from the Bourne shell, and a 5 digit random numbers if you are from the Korn shell. You could also do a ps -l and look for the shell with the highest PID.

5. What is Boyce Codd Normal form?

A relation schema R is in BCNF with respect to a set F of functional dependencies if for all functional dependencies in F+ of the form a->b, where a and b is a subset of R, at least one of the following holds:

- a->b is a trivial functional dependency (b is a subset of a)
- a is a superkey for schema R

[C++ networking questions](#)

Q.What is the difference between Stack and Queue?

A: Stack is a Last In First Out (LIFO) data structure.

Queue is a First In First Out (FIFO) data structure

Q: Write a function that will reverse a string. (Microsoft)

```
A: char *strrev(char *s)
{
int i = 0, len = strlen(s);
char *str;
if ((str = (char *)malloc(len+1)) == NULL) /*cannot allocate memory */
err_num = 2;
return (str);
}
while(len)
str[i++]=s[--len];
str[i] = NULL;
return (str);
}
```

Q: What is the software Life-Cycle?

A: The software Life-Cycle are

- 1) Analysis and specification of the task
- 2) Design of the algorithms and data structures
- 3) Implementation (coding)
- 4) Testing
- 5) Maintenance and evolution of the system
- 6) Obsolescence

Q: What is the difference between a Java application and a Java applet?

A: The difference between a Java application and a Java applet is that a Java application is a program that can be executed using the Java interpreter, and a JAVA applet can be transferred to different networks and executed by using a web browser (transferable to the WWW).

Q: Name 7 layers of the OSI Reference Model? (from Cisco)

A: -Application layer
-Presentation layer
-Session layer
-Transport layer
-Network layer
-Data Link layer
-Physical layer

[C++ algorithm specific interview questions](#)

Q1 What are the advantages and disadvantages of B-star trees over Binary trees? (Asked by Motorola people)

A1 B-star trees have better data structure and are faster in search than Binary trees, but it's harder to write codes for B-start trees.

Q2 Write the psuedo code for the Depth first Search.(Asked by Microsoft)

A2

dfs(G, v) //OUTLINE

Mark v as "discovered"

For each vertex w such that edge vw is in G:

If w is undiscovered:

dfs(G, w); that is, explore vw, visit w, explore from there as much as possible, and backtrack from w to v.

Otherwise:

"Check" vw without visiting w.

Mark v as "finished".

Q3 Describe one simple rehashing policy.(Asked by Motorola people)

A3 The simplest rehashing policy is linear probing. Suppose a key K hashes to location i. Suppose other key occupies H[i]. The following function is used to generate alternative locations:

$\text{rehash}(j) = (j + 1) \bmod h$

where j is the location most recently probed. Initially $j = i$, the hash code for K. Notice that this version of rehash does not depend on K.

Q4 Describe Stacks and name a couple of places where stacks are useful. (Asked by Microsoft)

A4 A Stack is a linear structure in which insertions and deletions are always made at one end, called the top. This updating policy is called last in, first out (LIFO). It is useful when we need to check some syntax errors, such as missing parentheses.

Q5 Suppose a 3-bit sequence number is used in the selective-reject ARQ, what is the maximum number of frames that could be transmitted at a time? (Asked by Cisco)

A5 If a 3-bit sequence number is used, then it could distinguish 8 different frames. Since the number of frames that could be transmitted at a time is no greater half the number of frames that could be distinguished by the sequence number, so at most 4 frames can be transmitted at a time.

Basic C++ interview questions

1.Question: Suppose that data is an array of 1000 integers. Write a single function call that will sort the 100 elements data [222] through data [321].

Answer: quicksort ((data + 222), 100);

2.Question: Which recursive sorting technique always makes recursive calls to sort subarrays that are about half size of the original array?

Answer: Mergesort always makes recursive calls to sort subarrays that are about half size of the original array, resulting in $O(n \log n)$ time.

3.Question: What is the difference between an external iterator and an internal iterator? Describe an advantage of an external iterator.

Answer: .An internal iterator is implemented with member functions of the class that has items to step through. .An external iterator is implemented as a separate class that can be "attach" to the object that has items to step through. .An external iterator has the advantage that many difference iterators can be active simultaneously on the same object.

4.Question: Why are arrays usually processed with for loop?

Answer: The real power of arrays comes from their facility of using an index variable to traverse the array, accessing each element with the same expression `a[i]`. All that is needed to make this work is an iterated statement in which the variable `i` serves as a counter, incrementing from 0 to `a.length - 1`. That is exactly what a loop does.

5.Question: What is an HTML tag?

Answer: An HTML tag is a syntactical construct in the HTML language that abbreviates specific instructions to be executed when the HTML script is loaded into a Web browser. It is like a method in Java, a function in C++, a procedure in Pascal, or a subroutine in FORTRAN.

[C++ coding interview questions](#)

1. Design and implement a String class that satisfies the following:

- Supports embedded nulls
- Provide the following methods (at least)
 - Constructor
 - Destructor
 - Copy constructor
 - Assignment operator
 - Addition operator (concatenation)
 - Return character at location
 - Return substring at location
 - Find substring
- Provide versions of methods for String and for `char*` arguments

2. Given the following classes

```
class Fruit {
// ...
}
class Apple : public Fruit {
// ...
}
class Peach : public Fruit {
// ...
}
// Container of fruit
class BasketOfFruit {
BasketOfFruit() ;
void insert( Fruit & f ) ;
// ...
}
// Container of apples
class BasketOfApples /* ??? */ {
// ...
}
```

Should BasketOfApples derive from BasketOfFruit? Why or why not?
What is the general principle that determines the answer?

3. Describe briefly what the following function does. What standard function is it most

like ?

```
int f( char *p ) {  
    int n = 0 ;  
    while ( *p != 0 ) n = 10*n + *p++ - '0' ;  
    return n ;  
}
```

4. Describe briefly what function 'a' does in the following code fragment.

```
struct s {  
    struct s *next ;  
}  
a( struct s *p, struct s *x ) {  
    while ( p->next != 0 ) p = p->next ;  
    p->next = x ;  
    x->next = 0 ;  
}
```

5. What default methods are declared implicitly by the C++ compiler for the class below:

```
class Empty  
{  
};
```

6. Given a system with a hard realtime priority, multithreaded architecture, with priorities from 1 (least) to 5 (most), explain the major flaw in the design below:

The following objects are shared between the threads:

Disk : This class is a singleton. The read() and write() methods both block on a simple atomic lock()/unlock() shared between the two. (ie, only one thread can access the disk, thru either read or write, at any given time). It also has a waitForData() method, which blocks (without claiming the lock) until either a timeout elapses, or data is ready. It returns true upon returning due to new data, false upon returning due to the timeout.

Network : This class is a singleton. The read() and write() methods both block on a simple atomic lock()/unlock() shared between the two. (ie, only one thread can access the disk, thru either read or write, at any given time).

Sensor: The Sensor class accesses a number of physical sensors. The first method, 'waitForData()', blocks until data has been collected from the sensors. The second method, 'processData()', does a series of long and cpu-intensive calculations on the data. These calculations often take several minutes. It then returns the processed data.

Each of the following threads is running concurrently. Assume that the pseudocode in each thread is looped infinitely (ie, encased in a while(true) { }). It is extremely important that information buffered to the disk be sent to the network as quickly as possible, this is why Thread 1 runs at priority 5. The system conditions checked in thread 3 are not particularly important events (not as important as the calculations done in thread 2). If the events aren't transmitted over the network for several minutes, it's not a problem at all. They do, however, contain a large amount of system information. Thread 4 watches for

serious system alarms, indicating serious problems. These are a serious concern and if not quickly buffered to the disk and sent to the network, can cause serious revenue loss.

```
Thread 1: (priority: 5)
while(!Disk.waitForData()) { yield(); } /* Wait until
someone has
written data to the disk */
Network.write(Disk.read()); /* Write the data buffered on
the disk to
the network */
Thread 2: (priority: 2)
while(!Sensor.waitForData()) { yield(); } /* Wait until the
sensors
have picked up data */
Disk.write(Sensor.processData()); /* process the data and
write it to
the disk. */
Thread 3: (priority: 1)
if (checkSystemCondition1()) /* If system condition 1 is
true.. */
Disk.write(SystemCond1Data); /* Grab the data on the system
condition and buffer it to disk */
if (checkSystemCondition2()) /* see above*/
Disk.write(SystemCond2Data);
if (checkSystemCondition3()) /* see above */
Disk.write(SystemCond3Data);
yield();
Thread 4: (priority: 4)
if (checkAlarms()) /* If any serious alarms exist */
Disk.write(AlarmData); /* Buffer that data to disk for
immediate
network transmit */
yield();
```

[Advanced C++ and STL interview questions](#)

Q: How do you link a C++ program to C functions?

A: By using the extern "C" linkage specification around the C function declarations.

Q: Explain the scope resolution operator.

A: It permits a program to reference an identifier in the global scope that has been hidden by another identifier with the same name in the local scope.

Q: What are the differences between a C++ struct and C++ class?

A: The default member and base-class access specifiers are different.

Q: How many ways are there to initialize an int with a constant?

A: Two.

There are two formats for initializers in C++ as shown in the example that follows. The first format uses the traditional C notation. The second format uses constructor notation.

```
int foo = 123;
```

int bar (123);

Q: How does throwing and catching exceptions differ from using setjmp and longjmp?

A: The throw operation calls the destructors for automatic objects instantiated since entry to the try block.

Q: What is your reaction to this line of code?

delete this;

A: It's not a good practice.

Q: What is a default constructor?

A: A constructor that has no arguments.

Q: What is a conversion constructor?

A: A constructor that accepts one argument of a different type.

Q: What is the difference between a copy constructor and an overloaded assignment operator?

A: A copy constructor constructs a new object by using the content of the argument object. An overloaded assignment operator assigns the contents of an existing object to another existing object of the same class.

Q: When should you use multiple inheritance?

A: There are three acceptable answers: "Never," "Rarely," and "When the problem domain cannot be accurately modeled any other way."

Q: What is a virtual destructor?

A: The simple answer is that a virtual destructor is one that is declared with the virtual attribute.

Q: Explain the ISA and HASA class relationships. How would you implement each in a class design?

A: A specialized class "is" a specialization of another class and, therefore, has the ISA relationship with the other class. An Employee ISA Person. This relationship is best implemented with inheritance. Employee is derived from Person. A class may have an instance of another class. For example, an employee "has" a salary, therefore the Employee class has the HASA relationship with the Salary class. This relationship is best implemented by embedding an object of the Salary class in the Employee class.

Q: When is a template a better solution than a base class?

A: When you are designing a generic class to contain or otherwise manage objects of other types, when the format and behavior of those other types are unimportant to their containment or management, and particularly when those other types are unknown (thus, the genericity) to the designer of the container or manager class.

Q: What is a mutable member?

A: One that can be modified by the class even when the object of the class or the member function doing the modification is const.

Q: What is an explicit constructor?

A: A conversion constructor declared with the explicit keyword. The compiler does not use an explicit constructor to implement an implied conversion of types. It's purpose is reserved explicitly for construction.

Q: What is the Standard Template Library?

A: A library of container templates approved by the ANSI committee for inclusion in the standard C++ specification.

A programmer who then launches into a discussion of the generic programming model, iterators, allocators, algorithms, and such, has a higher than average understanding of the new technology that STL brings to C++ programming.

Q: Describe run-time type identification.

A: The ability to determine at run time the type of an object by using the typeid operator or the dynamic_cast operator.

Q: What problem does the namespace feature solve?

A: Multiple providers of libraries might use common global identifiers causing a name collision when an application tries to link with two or more such libraries. The namespace feature surrounds a library's external declarations with a unique namespace that eliminates the potential for those collisions.

This solution assumes that two library vendors don't use the same namespace identifier, of course.

Q: Are there any new intrinsic (built-in) data types?

A: Yes. The ANSI committee added the bool intrinsic type and its true and false value keywords.

[C++ programming on UNIX platforms](#)

What is a Make file?(Fujitsu)

Make file is a utility in Unix to help compile large programs. It helps by only compiling the portion of the program that has been changed.

What is deadlock? (Novell)

Deadlock is a situation when two or more processes prevent each other from running. Example: if T1 is holding x and waiting for y to be free and T2 holding y and waiting for x to be free deadlock happens.

What is semaphore? (Novell)

Semaphore is a special variable, it has two methods: up and down. Semaphore performs atomic operations, which means ones a semaphore is called it can not be interrupted.

Is C an object-oriented language? (Microsoft)

C is not an object-oriented language, but limited object-oriented programming can be done in C.

Name some major differences between C++ and Java.

C++ has pointers; Java does not. Java is platform-independent; C++ is not. Java has garbage collection; C++ does not.

[Basic C++ and UNIX OS programming](#)

Have you studied buses? What types?

Have you studied pipelining? List the 5 stages of a 5 stage pipeline. Assuming 1 lock per stage, what is the latency of an instruction in a 5 stage machine? What is the throughput of this machine ?

How many bit combinations are there in a byte?

For a single computer processor computer system, what is the purpose of a processor cache and describe its operation?

Explain the operation considering a two processor computer system with a cache for each processor.

What are the main issues associated with multiprocessor caches and how might you solve them?

Explain the difference between write through and write back cache.

Are you familiar with the term MESI?

Are you familiar with the term snooping?

Describe a finite state machine that will detect three consecutive coin tosses (of one coin) that results in heads.

In what cases do you need to double clock a signal before presenting it to a synchronous state machine?

You have a driver that drives a long signal & connects to an input device. At the input device there is either overshoot, undershoot or signal threshold violations, what can be done to correct this problem?

What are the total number of lines written by you in C/C++? What is the most complicated/valuable program written in C/C++?

What compiler was used?

What is the difference between = and == in C?

Are you familiar with VHDL and/or Verilog?

What types of CMOS memories have you designed? What were their size? Speed?

What work have you done on full chip Clock and Power distribution? What process technology and budgets were used?

What types of I/O have you designed? What were their size? Speed?

Configuration? Voltage requirements?

Process technology? What package was used and how did you model the package/system? What parasitic effects were considered?

What types of high speed CMOS circuits have you designed?

What transistor level design tools are you proficient with? What types of designs were they used on?

What products have you designed which have entered high volume production?

What was your role in the silicon evaluation/product ramp? What tools did you use?

If not into production, how far did you follow the design and why did not you see it into production?

What programming language are you using?

What C++ libraries are you proficient with?

Which algorithm do you like the most? Why?

How do you debug SSH?

What is the QA process?

How do you train another QA engineer?

What bug tracking tools you have used? Have you used any free tools?

How do you start your QA if there are no system requirements?

Have you used MSVC? What do you think of it?

There are 3 lights (in one room) and 3 switches (in another room), one for each, if you only enter into the light room once. How can you find out which switch corresponds to which light?

What is your weakness?

Why do you think you are suited for this job?

If there is a day, when you find yourself not fitting in our team, what will you do?

What makes you think you are qualified for this job?

Do you like music? Which composers are your favourite?

What kind of PC games you like most? Why?

Are you familiar with collaboration tools? Which communication method do you prefer for talk, email and chat?

When will you be available to start work?

What security tools have you used?

Tell me about yourself.

Tell me about your experience with this type of work

What do you like and dislike about our company?

Why do you want to work for us?

What should we hire you? What can you do for us? What can you do that others can not?

What is the job's most attractive and least attractive factor?

What do you look for in a job?

Please give me your definition of *software test engineer*.

How long would it take you to make a meaningful contribution to our firm?

How long would you stay with us?

Are you thinking of going back to school or college?

What kind of programs/machines or equipment have you worked with?

You may be overqualified for this position we have to offer.

Give me an example of a project you handled from start to finish.

What was your last employer's opinion of you?

Can you work under pressure, deadline etc?

Do you have any questions?

What is it you liked and disliked about your last job?

What does static variable mean?

What is a pointer?

What is a structure?

What are the differences between structures and arrays?

In header files whether functions are declared or defined?

What are the differences between malloc() and calloc()?

What are macros? What are the advantages and disadvantages?

Difference between pass by reference and pass by value?

What is static identifier?

Where are the auto variables stored?

Where does global, static, local, register variables, free memory and C Program instructions get stored?

Difference between arrays and linked list?
What are enumerations?
Describe about storage allocation and scope of global, extern, static, local and register variables?
What are register variables? What are the advantage of using register variables?
What is the use of typedef?
Can we specify variable field width in a scanf() format string? If possible how?
Out of fgets() and gets() which function is safe to use and why?
Difference between strdup and strcpy?
What is recursion?
Differentiate between a for loop and a while loop? What are it uses?
What are the different storage classes in C?
Write down the equivalent pointer expression for referring the same element a[i][j][k][l]?
What is difference between Structure and Unions?
What the advantages of using Unions?
What are the advantages of using pointers in a program?
What is the difference between Strings and Arrays?
In a header file whether functions are declared or defined?
What is a far pointer? where we use it?
How will you declare an array of three function pointers where each function receives two ints and returns a float?
What is a NULL Pointer? Whether it is same as an uninitialized pointer?
What is a NULL Macro? What is the difference between a NULL Pointer and a NULL Macro?
What does the error 'Null Pointer Assignment' mean and what causes this error?
What is near, far and huge pointers? How many bytes are occupied by them?
How would you obtain segment and offset addresses from a far address of a memory location?
Are the expressions arr and *arr same for an array of integers?
Does mentioning the array name gives the base address in all the contexts?
Explain one method to process an entire string as one unit?
What is the similarity between a Structure, Union and enumeration?
Can a Structure contain a Pointer to itself?
How can we check whether the contents of two structure variables are same or not?
How are Structure passing and returning implemented by the compiler?
How can we read/write Structures from/to data files?
What is the difference between an enumeration and a set of pre-processor # defines?
What do the 'c' and 'v' in argc and argv stand for?
Are the variables argc and argv are local to main?
What is the maximum combined length of command line arguments including the space between adjacent arguments?

If we want that any wildcard characters in the command line arguments should be appropriately expanded, are we required to make any special provision? If yes, which?

Does there exist any way to make the command line arguments available to other functions without passing them as arguments to the function?

What are bit fields? What is the use of bit fields in a Structure declaration?

To which numbering system can the binary number 1101100100111100 be easily converted to?

Which bit wise operator is suitable for checking whether a particular bit is on or off?

Which bit wise operator is suitable for turning off a particular bit in a number?

Which bit wise operator is suitable for putting on a particular bit in a number?

Which bit wise operator is suitable for checking whether a particular bit is on or off?

Which one is equivalent to multiplying by 2?

- Left shifting a number by 1
- Left shifting an unsigned int or char by 1?

Write a program to compare two strings without using the strcmp() function.

Write a program to concatenate two strings.

Write a program to interchange 2 variables without using the third one.

Write programs for String Reversal. The same for Palindrome check.

Write a program to find the Factorial of a number.

Write a program to generate the Fibonacci Series?

Write a program which employs Recursion?

Write a program which uses command line arguments.

Write a program which uses functions like strcmp(), strcpy(), etc.

What are the advantages of using typedef in a program?

How would you dynamically allocate a one-dimensional and two-dimensional array of integers?

How can you increase the size of a dynamically allocated array?

How can you increase the size of a statically allocated array?

When reallocating memory if any other pointers point into the same piece of memory do you have to readjust these other pointers or do they get readjusted automatically?

Which function should be used to free the memory allocated by calloc()?

How much maximum can you allocate in a single call to malloc()?

Can you dynamically allocate arrays in expanded memory?

What is object file? How can you access object file?

Which header file should you include if you are to develop a function which can accept variable number of arguments?

Can you write a function similar to printf()?

How can a called function determine the number of arguments that have been passed to it?

Can there be at least some solution to determine the number of arguments passed to a variable argument list function?

How do you declare the following:

- An array of three pointers to chars
 - An array of three char pointers
 - A pointer to array of three chars
 - A pointer to function which receives an int pointer and returns a float pointer
 - A pointer to a function which receives nothing and returns nothing
- What do the functions atoi(), itoa() and gcvt() do?
- Does there exist any other function which can be used to convert an integer or a float to a string?
- How would you use qsort() function to sort an array of structures?
- How would you use qsort() function to sort the name stored in an array of pointers to string?
- How would you use bsearch() function to search a name stored in array of pointers to string?
- How would you use the functions sin(), pow(), sqrt()?
- How would you use the functions memcpy(), memset(), memmove()?
- How would you use the functions fseek(), fread(), fwrite() and ftell()?
- How would you obtain the current time and difference between two times?
- How would you use the functions randomize() and random()?
- How would you implement a substr() function that extracts a sub string from a given string?
- What is the difference between the functions rand(), random(), srand() and randomize()?
- What is the difference between the functions memmove() and memcpy()?
- How do you print a string on the printer?
- Can you use the function fprintf() to display the output on the screen?