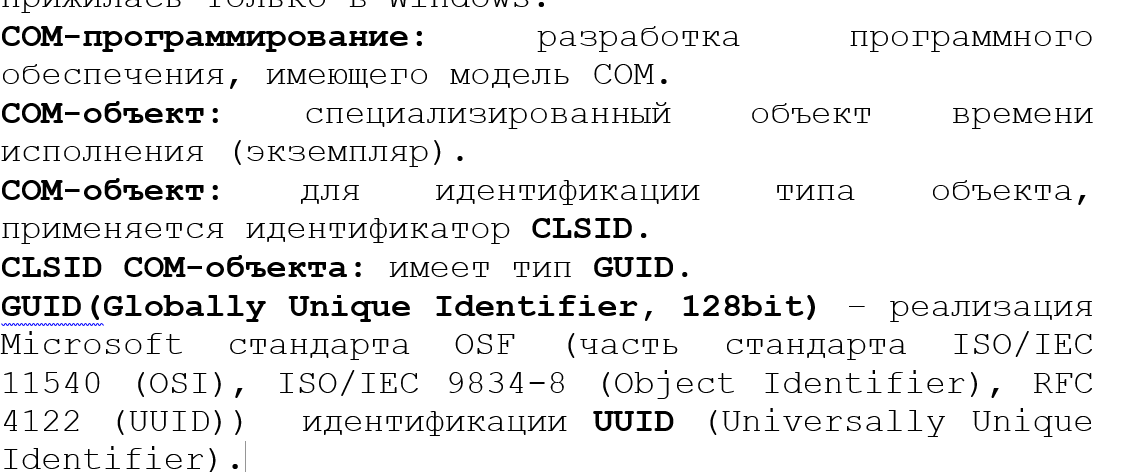
**19. Что такое COM? COM-программирование?**

Component Object Model – СОМ

Модель компонентного объекта фирмы Microsoft является, как следует из её названия, *моделью* для проектирования и создания компонентных объектов. Модель определяет множество технических приемов, которые могут быть использованы разработчиком при создании независимых от языка программных модулей, в которых соблюдается определенный двоичный стандарт.

**20. Что такое COM-объект(компонент)? CLSID?**



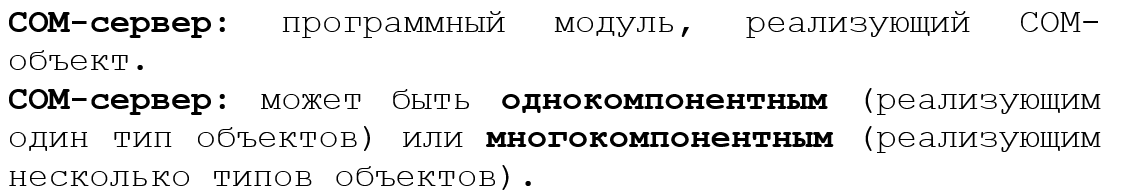
21. Что такое GUID? Где применяется GUID? Размер GUID-идентификатора?

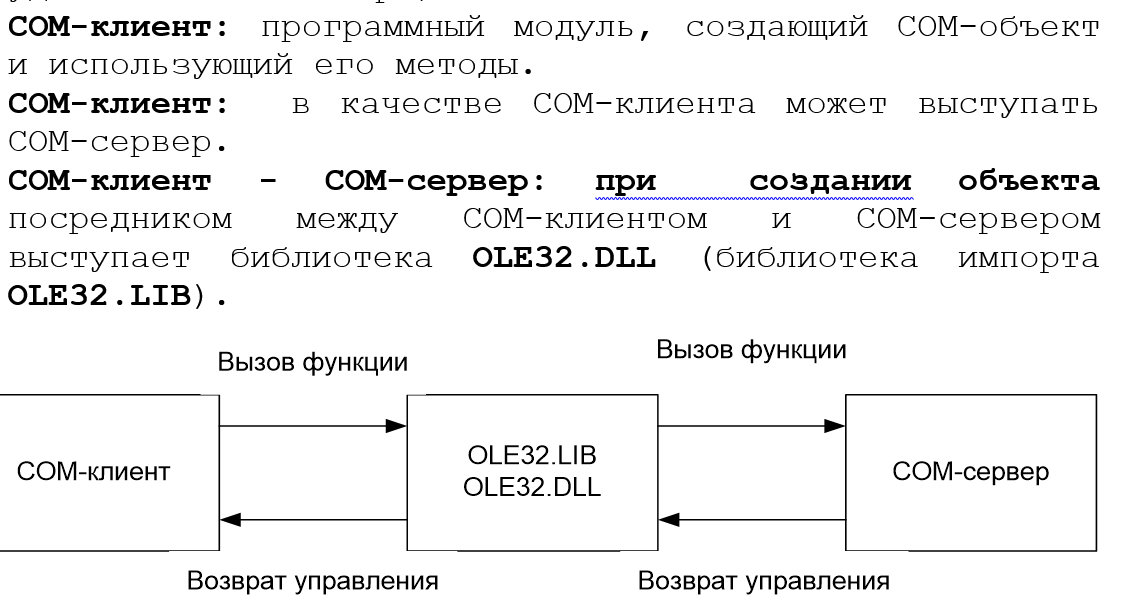
Метод QueryInterface обращается к идентификатору интерфейса (Interface Identifier — IID), который представляет собой 128-битовый уникальный идентификатор (а именно GUID, о котором вскоре пойдет речь), и возвращает указатель на определенный интерфейс (например, IUnknown, IMath), предоставляемый СОМ-объектом.

22. Какие типы COM-контейнеров бывают?

До сих пор мы обсуждали требования, предъявляемые к интерфейсам СОМ-компонентов. Однако после реализации компонента на том или ином языке программирования (в нашем случае на C++) он должен выполняться как процесс в определенной операционной системе. СОМ-компоненты хранятся либо в исполняемых файлах (ЕХЕ), либо в файлах динамически загружаемых библиотек Windows (DLL), либо в тех и в других.

23. Что является клиентом и сервером в COM?





24. Поясните понятия «однокомпонентный» и «многокомпонентный» COM-сервер. Б

25. Поясните типы COM-серверов: CLSCTX\_INPROC\_SERVER, CLSCTX\_LOCAL\_SERVER, СLSCTX\_REMOTE\_SERVER.

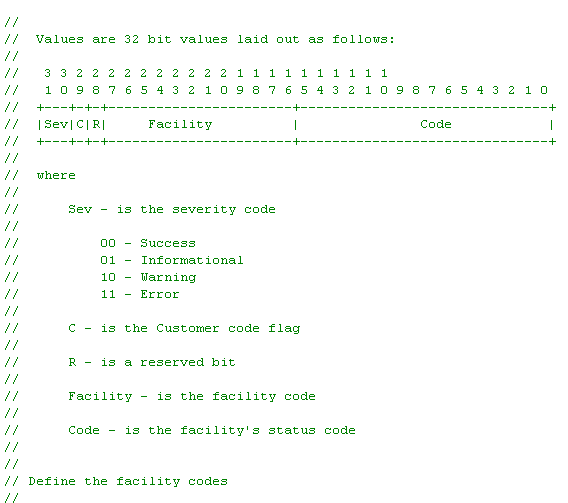
Термин *локальный сервер* используется для описания компонентов, которые хранятся в исполняемых файлах. Такие файлы, помимо СОМ-компонентов, могут содержать и другие функции. Например, Microsoft Word является локальным сервером. Он не только обеспечивает возможности обработки текстов, но также предоставляет множество СОМ-компонентов, к которым имеют доступ другие приложения.

*Внутризадачный сервер* представляет собой DLL-файл Windows. Выполнение этого компонента происходит в контексте вызывающего процесса и, таким образом, процесс клиента имеет прямой доступ к любому DLL-компоненту. Понятие внутризадачности сервера приобретет особую значимость при рассмотрении пользовательских СОМ-интерфейсов и процессов транспортировки.

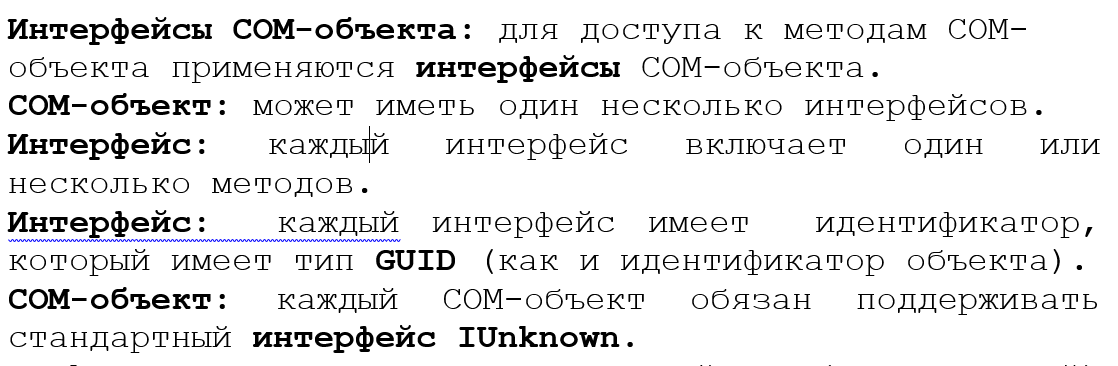
*Удаленный сервер* представляет собой компонент, загружаемый и выполняемый на удаленном компьютере. Обычно удаленный сервер строится как исполняемый файл, однако это не является обязательным требованием. Доступ к компонентам, хранящимся только в DLL-файлах, может осуществляться также и в удаленном режиме. В таких случаях модель СОМ создает *суррогатный процесс*, в котором могут выполняться удаленные DLL-файлы.

26. Как называется имя библиотеки, обеспечивающей работу COM-приложений.

27. Поясните назначение типа и структуру HRESULT.

****

28. Что такое COM-интерфейс?



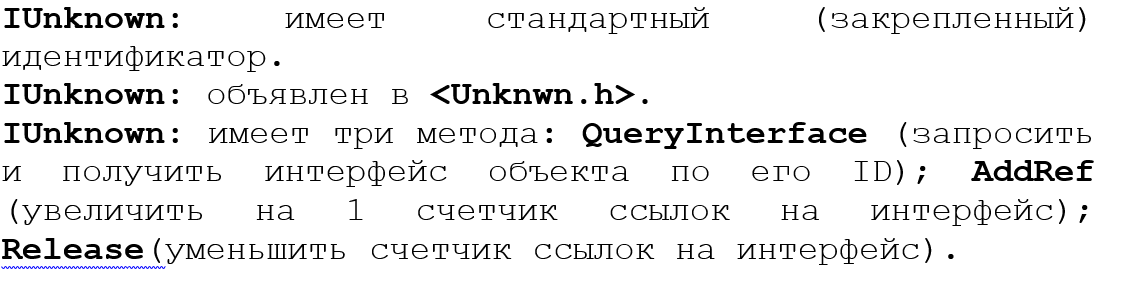
29. Чем характеризуется COM-интерфейс?

1. **Интерфейс:** каждый интерфейс должен быть производным от интерфейса **IUnknown.**

30. Что значит «стандартный» COM-интерфейс?

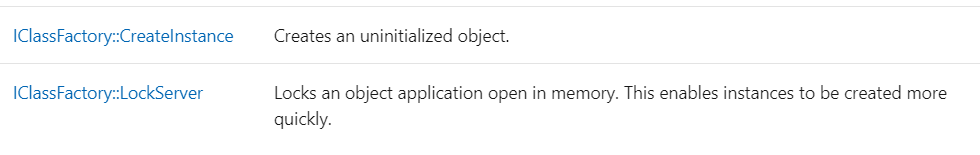
31. Назовите два стандартных COM-интерфейса.

32. Перечислите методы интерфейса IUnknown и поясните их назначение.



33. Что такое «фабрика классов» и для чего она нужна?

34. Перечислите методы интерфейса IClassFactory и поясните их назначение.



35. Что такое «счетчик ссылок на интерфейсы»? Для чего он нужен? Каким образом и когда этот счетчик увеличивается и уменьшается?

оскольку экземпляр СОМ-компонента может иметь несколько интерфейсов, связанных со многими клиентами, нашему объекту необходимо иметь некоторую возможность подсчета обращений к нему (счетчик). Всякий раз, когда клиент запрашивает интерфейс, значение счетчика будет увеличиваться, а когда клиент завершает работу с интерфейсом — уменьшаться. В конце концов, когда значение счетчика обращений станет равным нулю, СОМ-компонент будет уничтожен. Именно для этого и служат методы IUnknown::AddRef() и IUnknown::Release(). Пользователь компонента не может непосредственно удалить экземпляр С++-объекта, поскольку в его распоряжении имеется только указатель на виртуальную таблицу функций C++. В действительности клиент не имеет права удалять объект в любом случае, поскольку могут существовать другие клиенты, использующие тот же компонентный объект. Только сам компонент, основываясь на значении своего внутреннего счетчика обращений, может определить момент своего удаления.

36. Какое соглашение о вызове и возврате должен обеспечивать метод COM-объекта? Какие методы являются исключением?

1. **Методы**  **COM-объекта:** поддерживают соглашение о вызове **stdcall** (аргументы передаются через стек, справа налево, очистку стека производит **вызываемая** подпрограмма);используется макрос **STDMETHODCALLTYPE.**
2. **Методы COM-объекта:** возвращают значение типа **HRESULT** (как функции OLE32). **Исключение** составляют методы **AddRef** и **Release** интерфейса **IUnknown**.

37. Что должен «знать» COM-клиент, чтобы использовать COM-объект?

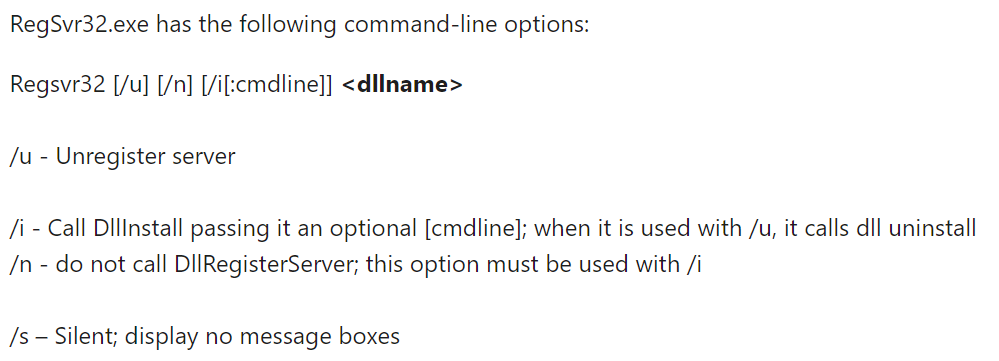
38. Объясните в чем заключается процесс регистрации COM-объекта?

The registry is a system database that contains information about the configuration of system hardware and software as well as about users of the system. Any Windows-based program can add information to the registry and read information back from the registry. Clients search the registry for interesting components to use.

The registry maintains information about all the COM objects installed in the system. Whenever an application creates an instance of a COM component, the registry is consulted to resolve either the CLSID or ProgID of the component into the pathname of the server DLL or EXE that contains it. After determining the component's server, Windows either loads the server into the process space of the client application (in-process components) or starts the server in its own process space (local and remote servers). The server creates an instance of the component and returns to the client a reference to one of the component's interfaces.

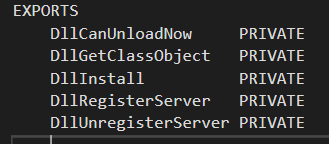
39. Поясните назначение утилиты regsvr32 и принцип ее работы.

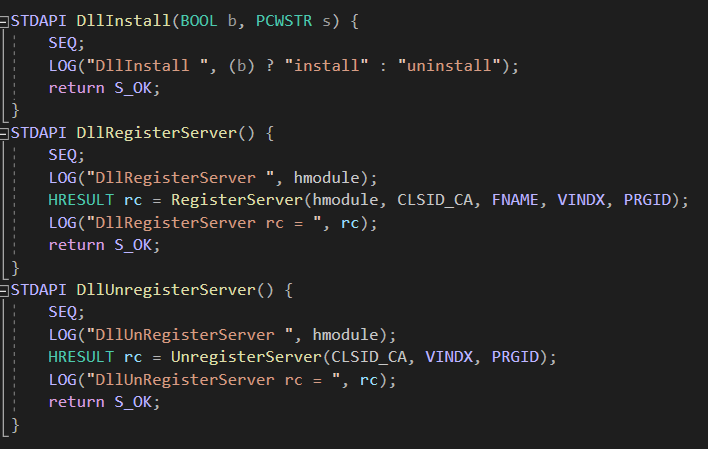
Regsvr32 is a command-line utility to register and unregister OLE controls, such as DLLs and ActiveX controls in the Windows Registry. Regsvr32.exe is installed in the %systemroot%\System32 folder in Windows XP and later versions of Windows.



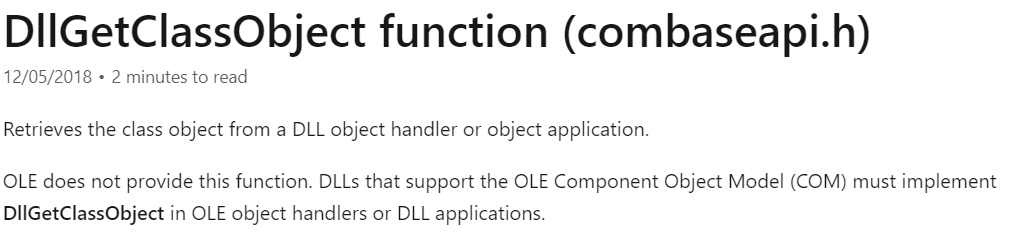
40. Поясните назначение утилиты regedit.

41. Перечислите пять функций, которые экспортируются COM/DLL-контейнером. Поясните назначение этих функций.

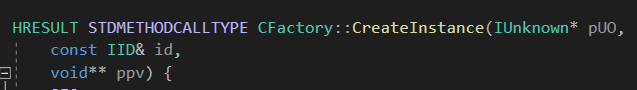




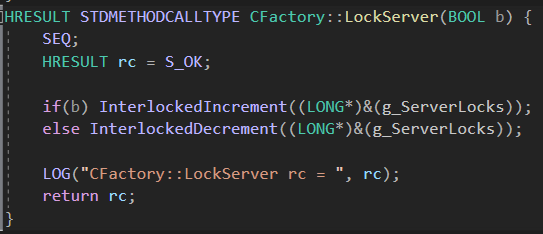
42. Назовите функцию COM-контейнера, которая вызывается OLE32 для получения указатель на фабрику классов.



43. Назовите функцию фабрики классов, в которой создается объект компонента.



44. Поясните назначение «счетчика экземпляров компонент». Где этот счетчик увеличивается и где уменьшается?





45. Назовите условие, при котором объект компонента удаляется.

// когда значение счетчика обращений

// становится равным нулю, объект удаляет сам себя

46. Объясните на механизм блокировки COM-сервера (функция LockServer фабрики классов).

**IClassFactory::LockServer** controls whether an object's server is kept in memory. Keeping the application alive in memory allows instances to be created more quickly.