1. Что такое класс и объект? Что они представляют собой с точки зрения предметной области с одной стороны и программы с другой?

Класс — это тип данных, описывающий наиболее существенные свойства предметов и понятий ПрО, стереотипы их поведения и место в их окружении.

Объект — это переменная (константа), создаваемая по шаблону, описываемому классом.

С точки зрения предметной области класс может представлять абстрактную сущность, а объект —конкретную. С точки зрения программы класс — это код, который определяет структуру и поведение объектов, а объект — это данные в программе, которые были созданы на основе этого класса и которые можно использовать для выполнения операций или решения задач.

1. В чем заключается принцип инкапсуляции ООП? Какие выделяют области видимости в классе и в чем их суть?

В классической реализации принципа инкапсуляции существует 3 области видимости:

- public (общая) - это область, в которой описываются члены класса, образующие его интерфейс,

- private (частная) - это область скрывает свои описания от окружения, эти члены класса могут использоваться только в реализации (алгоритмах) методов самого класса,

- protected (защищенная) - эта область, значение которой будет рассмотрено позже при рассмотрении вопросов наследования.

1. Что общего и в чем различие классов, описываемых как class, struct, union?

Классы, описываемые с помощью ключевых слов и struct class отличаются между собой только не именованной областью видимости:

- для struct - это public,

- для class - это private.

Классы, описываемые с помощью ключевого слова union могут иметь только одну область видимости, public, указание на которую опускается.

1. Что из себя представляют атрибуты и методы классов? Чем отличаются различные объекты одного и того же класса?

Совокупность свойств образует структуру данных класса, компоненты которой имеют ряд синонимов, таких как поля, атрибуты. Эта группа полей является идентифицирующей, т.е. определяющей идентичность или различия отдельных экземпляров класса.

Стереотипами поведения выступают функции, называемые методами, реализация которых использует значения полей экземпляров класса.

1. Что такое конструктор? Сколько их может быть?

Конструктором класса называется специальный метод класса, который выполняется при создании объекта класса.

Конструкторов в классе может сколь угодно много, в том числе и не быть вообще

1. Что такое конструктор копирования? Для чего он используется?

Конструктора копирования-это конструктор, формальный параметр которого - это объект класса.

Динамический контекст вновь создаваемого объекта при копировании необходимо создавать заново и копировать его, а не его указатель из статического контекста.

1. В чем отличие переменной, указателя на переменную и ссылки на переменную?

Ссылки не есть полноправные объекты, подобные переменным, либо указателям. После инициализации значение ссылки изменить нельзя, она всегда смотрит на тот участок памяти (на тот объект), с которым она связана инициализацией. Ни одна из операций не действует на ссылку, а относится к тому объекту, с которым она связана.

1. Почему в конструкторе копирования должна использоваться именно ссылка?

При использовании ссылки в качестве формального параметра обеспечивается доступ из тела функции к соответствующему фактическому параметру напрямую, минуя копирование параметра в стек, имеющего рассмотренную выше проблему, которая при очистке стека ведет к разрушению объекта. Ссылка же обеспечивает прямой доступ к участку памяти, выделенному для фактического параметра.