1. Какими способами можно задать значения для данных-членов структуры?  
     
   Можно прописывать значения по умолчанию в полях структуры

● Можно использовать список инициализации (агрегатную инициализацию):

○ Person vanya = {“Ivan”, “Ivanov”, 18}; //salary 0 либо мусор

○ Person vanya {“Ivan”, “Ivanov”, 18}; //uniform - инициализация

● Можно использовать конструкторы - специальная функция-член,

вызываемая при создании экземпляра класса/структуры.

1. Для чего предназначены списки инициализации в конструкторах структур?  
     
   Можно инициализировать переменные в конструкторе, не присваивая им значение.
2. Как идиома RAII реализовывается с помощью конструктора и деструктора?  
     
   Конструктор + деструктор могут реализовать идиому RAII (Resource Acquisition Is Initialization, Получения ресурса есть инициализация) -помощью тех или иных программных механизмов получение некоторого ресурса неразрывно совмещается с инициализацией, а освобождение — с уничтожением объекта. Это удобно и безопасно и активно используется в std автоматически

Например:

● Открытие\закрытие файла

● Работа операторов new\delete

1. Когда стоит рассматривать использование битовых полей и объединений?  
     
   ● Если ограничено место для хранения информации, можно сохранить несколько логических (истина/ложь) переменных в одном байте.

● Некоторые интерфейсы устройств передают информацию, закодировав биты в один байт.

● Некоторым процедурам кодирования необходимо получить доступ к отдельным битам в байте.

1. Какими особенностями обладают перечисления с областью видимости?  
     
   1) У перечислений с областью видимости не возникает конфликтов из-за названий переменных.

2) Для приведения объекта из перечисления к целочисленному типу, нужно вручную прописывать static\_cast.