№1

Для организации коммуникации между одновременно работающими процессами применяются средства межпроцессного взаимодействия (Interprocess Communication - IPC).

Выделяются три уровня средств IPC:

• локальный;

• удаленный;

• высокоуровневый

Средства локального уровня IPC привязаны к процессору и возможны только в пределах компьютера.   
Удаленные IPC предоставляют механизмы, которые обеспечивают взаимодействие как в пределах одного процессора, так и между программами на различных процессорах, соединенных через сеть.  
Под высокоуровневыми IPC обычно подразумеваются пакеты программного обеспечения, которые реализуют промежуточный слой между системной платформой и приложением. Эти пакеты предназначены для переноса уже испытанных протоколов коммуникации приложения на более новую архитектуру.

№2

Файлы, отображаемые в память - это механизм, который позволяет отображать файлы на участок памяти. Таким образом, при чтении данных из неё, производится считывание соответствующих байт из файла. С записью аналогично.

Разделяема память – это техника, позволяющая осуществлять обмен информацией через общий для процессов сегмент памяти **без использования системных вызовов ядра**. Сегмент разделяемой памяти подключается в свободную часть [виртуального адресного пространства](https://ru.wikipedia.org/wiki/%D0%90%D0%B4%D1%80%D0%B5%D1%81%D0%B0%D1%86%D0%B8%D1%8F_%D0%BF%D0%B0%D0%BC%D1%8F%D1%82%D0%B8#%D0%B2%D0%B8%D1%80%D1%82%D1%83%D0%B0%D0%BB%D1%8C%D0%BD%D0%B0%D1%8F) процесса. Таким образом, два разных процесса могут иметь разные адреса одной и той же ячейки подключенной разделяемой памяти.

№3

* Программист не знает расположения контейнера в памяти
* Для размещения контейнера в разделяемой памяти придётся использовать специальный аллокатор STL
* Элементы контейнера могут быть непоследовательными

№4

Именованные примитивы синхронизации хранятся внутри ядра, как следствие их имя знает операционная система, анонимные же находятся в разделяемой памяти.

№5

При использовании динамической компоновки загрузочный код нескольких (или нескольких десятков) функций объединяется в отдельные файлы, загружаемые в оперативную память в единственном экземпляре. Программы, работающие параллельно, вызывают функции, загруженные в память из файлов библиотек динамической компоновки, а не из файлов программ.

Таким образом, используя механизм динамической компоновки, в загрузочном файле программы можно расположить только те функции, которые являются специфическими для данной программы. Те же функции, которые нужны всем (или многим) программам, работающим параллельно, можно вынести в отдельные файлы - библиотеки динамической компоновки, и хранить в памяти в единственном экземпляре (рис. 3.2). Эти файлы можно загружать в память только при необходимости, например, когда какая-нибудь программа захочет вызвать функцию, код которой расположен в библиотеке.