1. **Поясните понятие «виртуальная память».**

Виртуальная память–метод управления памятью процессора, предназначенный для выполнения программ, которым выделяется адресное пространство превышающее доступный физический объем памяти компьютера.

1. **Поясните понятие «свопинг».**

Свопинг–аппаратно-программный механизм OS обмена (вытеснения и загрузки) содержимым блоков оперативной физической памяти компьютера с устройством хранения данных с целью расширения адресуемого объема оперативной памяти компьютера.

1. **Поясните понятие «страничная память».**

Страничная память – реализации виртуальной памяти, при которой физическая память и адресное пространство разбивается на блоки (страницы), а также осуществляется страничный свопинг.

1. **Поясните понятие MMU.**

Memory Management Unit – диспетчер памяти**,** входящее в состав процессора и предназначенное для трансляции виртуальных адресов оперативной памяти в реальные.

1. **Поясните понятие TLB.**

Translation Lookaside Buffer – буфер быстрого преобразования адреса, компонент MMU, предназначенный для вычисления реальных адресов.

1. **Какая информация содержится в строке таблицы страниц**

****

1. **Поясните принцип применения хэш-таблиц.**

Хэш-таблицы, созданные на основе виртуальных адресов, решают проблему инвертированной таблицы, которая усложняет преобразование виртуальных адресов в физические: все находящиеся на данный момент в памяти виртуальные страницы, имеющие одинаковые хэш-значения, связываются в одну цепочку. Как только будет найден номер страничного блока, в TLB будет введена новая пара значений (виртуального, физического)

1. **Поясните применение «инвертированной таблицы физических» страничной памяти.**

Инвертированная таблица страниц – таблица для физических страниц, в которой имеется одна запись для каждого страничного блока в реальной памяти, а не одна запись на каждую страницу в виртуальном адресном пространстве. В каждой записи отслеживается, что именно находится в страничном блоке.

1. **Поясните понятие «рабочий набор страниц».**

Рабочий набор страниц - это набор страниц памяти, которые в настоящее время использует процесс (принадлежат вашему процессу и не выгружены).

1. **Поясните принцип работы алгоритма LRU.**

Least recently used - это алгоритм, при котором вытесняются кэшированные значения, которые дольше всего не запрашивались. Соответственно, необходимо хранить время последнего запроса к значению. И как только число закэшированных значений превосходит максимально возможный размер кэша, необходимо вытеснить из кеша значение, которое дольше всего не запрашивалось.

1. **Windows: поясните назначение сервиса SysMain.**

Сервис SysMain постоянно пребывает в фоновом режиме, анализируя модели использования ОЗУ и выясняя, какие страницы используются чаще всего и предварительно подкачивает их в сжатом виде, делая последующий доступ к ним намного быстрее.

1. **Windows: поясните назначение файла hiberfil.sys.**

Файл hiberfil.sys — это файл гибернации(сна), используемый в Windows для хранения данных и их последующей быстрой загрузки в оперативную память при включении компьютера или ноутбука.

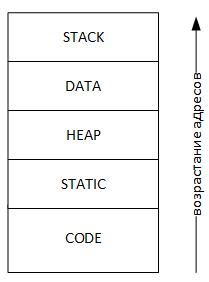
1. **Windows: поясните назначение файла pagefile.sys.**

pagefile.sys - это файл подкачки Windows.

1. **Windows: поясните назначение файла swapfile.sys.**

swapfile.sys - файл подкачки (в дополнение к pagefile.sys), но служащий исключительно для приложений из магазина приложений (UWP). В swapfile.sys записываются данные приложений из магазина, которые не требуются в настоящий момент времени, но могут внезапно потребоваться, и работает отличным от обычного файла подкачки Windows образом, представляя собой своего рода механизм «гибернации» для приложений.

1. **Windows: перечислите области адресного пространства (от младших к старшим адресам) и поясните их назначения.**



Stack - для временного хранения данных программы, результатов промежуточных вычислений

Data – для постоянного хранения инициализированных данных программы

Heap – для хранения динамических данных программы

Static – для хранения статических данных программы

Code – для хранения скомпилированного кода программы

1. **Windows: какой стандартный начальный размер области heap?**

По умолчанию – 1MB, из них 4K сразу забирает процесс.

1. **Windows: каким образом можно изменить начальный размер области памяти heap приложения?**

Можно установить стартовое значение величины HEAP в параметрах Linker. В Visual Studio : Проект -> Свойства -> Свойства конфигурации -> Компоновщик -> Все параметры -> Резервный размер кучи.

1. **Windows: какой стандартный размер области памяти stack?**

1MB

1. **Windows: каким образом можно изменить размер области памяти stack приложения?**

в Visual Studio : Проект -> Свойства -> Свойства конфигурации -> Компоновщик -> Все параметры -> Резервный размер стека.

1. **Windows: поясните назначение функции Windows API: GlobalMemoryStatus.**

Получает информацию о текущем состоянии памяти компьютера, например, процент использования памяти, объем физической памяти, объем виртуальной памяти, размер файла подкачки и т.д.

1. **Windows: поясните назначение функции Windows API: VirtualQuery; перечислите значения атрибутов Protect, State и Type.**

Извлекает информацию о диапазоне страниц в виртуальном адресном пространстве вызывающего процесса.

****

****

****

1. **Windows: что такое «рабочее множество»? поясните принцип управления рабочим множеством с помощью OS API.**

Рабочее множество - количество памяти, требующееся процессу в заданный интервал времени.

Процесс устанавливает минимальный и максимальный размеры рабочего множества с помощью SetProcessWorkingSetSize.

Процесс удаляет как можно больше страниц из рабочего набора множества с помощью EmptyWorkingSet.

1. **Windows: что означает «страница заблокирована»? с помощью каких функций OS API можно установить блокировку страниц и снять блокировку? Какое максимальное количество страниц можно заблокировать?**

Если страница заблокирована, это означает, что остальным процессам запрещен доступ к этой странице, и последующий доступ к ней текущим процессом не приведет к ошибке. Можно установить/снять блокировку страниц с помощью функций VirtualLock и VirtualUnlock соответственно. Максимальное количество страниц, которые может заблокировать процесс, равно количеству страниц в его минимальном рабочем множестве за вычетом небольших накладных расходов.

1. **Windows: что такое «heap»? Что такое «heap процесса»? Что такое «пользовательская heap»? Поясните принцип устройства heap.**

Heap – область памяти адресного пространства, предназначенного для использования программной фрагментов динамически выделяемой памяти (malloc, new).

Heap процесса – куча, создаваемая в адресном пространстве процесса при его инициализации (ее размер по умолчанию — 1 Мб).

Пользовательская heap – куча, создаваемая пользователем.

При запуске [процесса](https://ru.wikipedia.org/wiki/%D0%9F%D1%80%D0%BE%D1%86%D0%B5%D1%81%D1%81_(%D0%B8%D0%BD%D1%84%D0%BE%D1%80%D0%BC%D0%B0%D1%82%D0%B8%D0%BA%D0%B0)) [ОС](https://ru.wikipedia.org/wiki/%D0%9E%D0%BF%D0%B5%D1%80%D0%B0%D1%86%D0%B8%D0%BE%D0%BD%D0%BD%D0%B0%D1%8F_%D1%81%D0%B8%D1%81%D1%82%D0%B5%D0%BC%D0%B0) выделяет [память](https://ru.wikipedia.org/wiki/%D0%9E%D0%BF%D0%B5%D1%80%D0%B0%D1%82%D0%B8%D0%B2%D0%BD%D0%B0%D1%8F_%D0%BF%D0%B0%D0%BC%D1%8F%D1%82%D1%8C) для размещения кучи. В дальнейшем [память](https://ru.wikipedia.org/wiki/%D0%9E%D0%BF%D0%B5%D1%80%D0%B0%D1%82%D0%B8%D0%B2%D0%BD%D0%B0%D1%8F_%D0%BF%D0%B0%D0%BC%D1%8F%D1%82%D1%8C) для кучи (под кучу) может выделяться динамически. [Память](https://ru.wikipedia.org/wiki/%D0%9E%D0%BF%D0%B5%D1%80%D0%B0%D1%82%D0%B8%D0%B2%D0%BD%D0%B0%D1%8F_%D0%BF%D0%B0%D0%BC%D1%8F%D1%82%D1%8C) кучи можно разделить на **занятую** (выделенную программе с помощью [функций](https://ru.wikipedia.org/wiki/%D0%A4%D1%83%D0%BD%D0%BA%D1%86%D0%B8%D1%8F_(%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5)), подобных malloc()) и **свободную** (ещё не занятую или уже освобождённую с помощью [функций](https://ru.wikipedia.org/wiki/%D0%A4%D1%83%D0%BD%D0%BA%D1%86%D0%B8%D1%8F_(%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5)), подобных free()). Для хранения данных о том, какая область кучи является занятой, а какая — свободной, обычно используется дополнительная область памяти. Перед началом работы программы выполняется инициализация кучи, в ходе которой [память](https://ru.wikipedia.org/wiki/%D0%9E%D0%BF%D0%B5%D1%80%D0%B0%D1%82%D0%B8%D0%B2%D0%BD%D0%B0%D1%8F_%D0%BF%D0%B0%D0%BC%D1%8F%D1%82%D1%8C), выделенная под кучу, отмечается как свободная.

1. **Linux: в какой части адресного пространства выделяется памяти с помощью функций malloc, calloc?**

В heap.