

Системное программирование

1. Системное программирование: определение, назначение, применение.
Два подхода есть системного программирования:
 1. Разработка на уровне API ОС
 2. Разработка приложений для других программистов
Разработка низкоуровневых библиотек
Системный вызов
2. Фреймворк операционной системы: определение, назначение, применение, состав. Стандарт POSIX.
Набор библиотек (там DLL хуе мое) которые нам позволяют...
которые представляют нам API операционной системы, и мы через фреймворк ОС добираемся к системным вызовам.
Что это такое? Как он устроен? Какие отдельные функции вы знаете отдельные... управления процессами, потоками, набор функций синхронизации, набор функций файловой системы, работы с виртуальной памятью и heapом (то есть какие разделы API мы знаем)
POSIX
3. Применение процессов в ОС Windows, API для работы с процессами.
Как можно применить процессы? Для чего? Как их можно запустить? Какие параметры указать при запуске? Как остановить процесс? Как запустить дочерний процесс? Как можно работать с процессами? Как управлять процессами?
4. Методы межпроцессного взаимодействия в ОС Windows: обмен данными, синхронизация.
Есть два процесса каким образом мы можем обеспечить обмен данными между ними?
Named pipe, mailslot, файл
Через память в дочернем процессе, если там что-то выделена виртуальная память, я нихуя не поняла
Как можно два процесса синхронизировать?
Через крит сек нельзя
5. Применение потоков в ОС Windows, API для работы с потоками, API для синхронизации потоков.
Как можно применять потоки
Разница между асинхронностью и параллельностью:
Параллельность – два потока работают параллельно если они одновременно работают на разных процессорах

Асинхронная операция происходит в два этапа 1в рамках одного процесса идет заявка на исполнение 2извлечение результата(спустя время), и вот между этими двумя событиями можно что-то делать

Асинхронная операция это схема?

В случае поток не работает ? Что он делает если он не работает? В каком состоянии он находится?

1 может стоять в очереди

2 может быть заблокирован(перация ввода.вывода)

3 suspended (кто-то его приостановил)

4 Sleep

Можно сначала сделать его suspend, а потом сверху sleep -

Можно, проснется, но работать не будет

Volatile — ключевое слово языков C/C++, которое информирует компилятор о том, что значение переменной может меняться извне и что компилятор не будет оптимизировать эту переменную.

Если есть общая область памяти между двумя потоками вы ему говорите сука не трогай ее не оптимизируй

6. Применение механизмов синхронизации в ОС Windows, API для синхронизации.

Механизмы апи

7. Файловая система: логическая и физическая организация данных, определение файловой системы, отличие файловых систем, оглавление файловой системы, файлы, каталоги, основные функции файловой системы, буферы ввода/вывода, кэширование ввода/вывода, основные функции API файловой системы, маркер файла, текущая позиция файла, блокировка файлов, наблюдение за изменением в каталоге, особенности устройства файловой системы в Linux.

Какие есть фс, чем отличаются, что можем делать, какие есть команды в файловой системе?

Чем отличается moved от copied?

Как проверить есть изменения в файле? Синхронная/

8. Работа с оперативной памятью в ОС Windows: API для работы с виртуальной памятью, API для работы с Heap.

Для чего есть две памяти **виртуальная?** и heap?

Heap находится в виртуальной памяти.

Когда вы просите память у виртуальной памяти то она выделяет ее страницами

4килобайта за страницу, и вот если нам так дохуя не нада используется куча

Когда нам нужны большие куски памяти то пользуемся виртуальной, маленькие heap

Как устроена куча? Это два двухсвязных списка, один список содержит занятые области памяти, второй свободные, которые находятся в рамках heap

Что такое фрагментация хипа

Упаковка? хипа

9. Механизм отображение файлов в памяти: последовательность системных вызовов Windows для создания образа файла в оперативной памяти, использование образа файла, как средства межпроцессного взаимодействия.

Что это такое как это такое? Как это сделать? Для чего САМОЕ ГЛАВНОЕ?

Для межпроцессорного обмена и еще для чего-то

10. Динамически вызываемые библиотеки: структура DLL-библиотеки, экспорт функций, загрузка динамической библиотеки, динамический вызов функций динамической библиотеки, создание и применение библиотеки импорта.

Что это такое? Как они проецируются в адресное пространство процесса? Функция DllMain для чего нужна и когда выполняется? Какие параметры передаются в нее? Причина вызова? Что такое экспорт функции? Как делается? Как применять после этого? Что такое библиотека импорта, как ее получить? И для чего она нужна? На каком этапе все происходит для всех типов? Как между собой взаимодействуют?

11. Спецификация COM: понятие позднего связывания программных модулей, COM-интерфейс, стандартные COM-интерфейсы, структура COM-клиента, структура COM/DLL-сервера, экспортируемые стандартные функции, регистрация COM/DLL-сервера.

Надо рассказать спецификацию COM, какие есть типы серверов, как устроена, как регистрация осуществляется, каким образом создаются компоненты, чем характеризуются компоненты и интерфейсы? Какие счетчики блокируются (два типа счетчиков)? Стандартные функции какие-то?

12. Управление пользователями и группами пользователей в Windows: понятие дискреционной системы безопасности, типы Windows-пользователей, группы пользователей, возможности API управления пользователями и группами.

Короче все вопросы с коллока, какие есть уровни доступа,

13. Структурная обработка ошибок в Windows: программное исключение, программные конструкции для обработки ошибок в Windows, фильтры, возможности API для структурной обработки ошибок, генерация ошибок, финальная обработка исключений.

Три варианта фильтра

С каким образом можно сгенерировать ошибку

14. Windows-консоль: определение, применение стандартных потоков для ввода/вывода в консоль, возможности API для управления консолью.

Расскажи все что ты можешь делать с консолью

15. Windows-сервисы: определение, назначение, применение, API.

Для чего он нужен, как им можно управлять

16. Асинхронные операции ввода вывода: понятие асинхронной операции ввода/вывода, особенности программирования асинхронного ввода/вывода.

Уже было

17. Порты завершения ввода/вывода: назначение, применение, API.

Рассказать принцип их работы, какая-то схема, для чего и в каких случаях это эффективно применять

18. Платформа Docker: архитектура, назначение, принципы устройства, файловая система UFS, контейнеры, образы, основные команды.

Что можно сделать с помощью него? Из чего состоит (две части)?

Что можно записать в докер файл и как? Как можно создать имедж или чо он спизданул?

В конце я поняла, что фраза "это схема" значит что он рисовал схему.

Доцент каф. ИСиТ

В.В. Смелов