Экзаменационные вопросы по дисциплине «Системное программирование» Специальность - 1-40 05 01-03 Информационные системы и технологии

1. **Введение.** Введение в системное программирование. Что такое программа? Что такое ПО? Классификация ПО. Что такое системное ПО? Классификация и функции системного ПО.
2. **Введение.** Что такое системное программирование? Системы программирования: определение и состав. Что такое транслятор? Какие существуют виды трансляторов? Назовите и опишите этапы подготовки программы. Назовите и опишите результат работы каждого из этапов.
3. **clang и CMake.** Что такое clang? LLVM? Преимущества использования clang? Что такое система сборки и зачем она нужна? Что такое CMake и в чём особенность таких систем? Что такое генератор? Какие бывают сборки в CMake? Что такое мультиконфигурация? Что такое CMakeLists? Опишите структуру CMakeLists для базового проекта.
4. **Файлы, отображенные в память.** Что такое отображение файла в память? Для чего они применяются в ОС? Как устроен данный механизм в Windows? Классификация объектов секций в Windows? Что такое представление секции? Согласован ли доступ к данным в нескольких секциях? Алгоритм взаимодействия с файлами, отображенными в память с использованием WinAPI.
5. **Файлы, отображенные в память.** Что такое отображение файла в память? Для чего они применяются в ОС? Как устроен данный механизм согласно POSIX? Классификация отображений согласно POSIX? Поясните каждый тип. Что такое Copy on Write? Алгоритм взаимодействия с файлами, отображенными в память с использованием POSIX API.
6. **Библиотеки.** Что такое библиотека? Какова причина возникновения библиотек? Какие бывают библиотеки? Что такое связывание? Какие виды связывания существуют? Как они соотносятся с типами библиотек? Поясните каждый из видов связывания.
7. **Библиотеки.** Что такое статическая библиотека? Какое связывание лежит в основе статических библиотек? Как создать статическую библиотеку используя clang напрямую? Используя CMake? Как собрать приложение с использованием статических библиотек используя clang напрямую? Используя CMake? Преимущества и недостатки статических библиотек.

8. **Библиотеки.** Что такое разделяемая (динамическая) библиотека? В чем ключевая идея таких библиотек? Какой механизм ОС лежит в основе работы разделяемых библиотек? Какие способы подключения разделяемых библиотек существуют? Как создать разделяемую библиотеку используя clang напрямую? Используя CMake? Преимущества и недостатки разделяемых библиотек.

9. **Библиотеки.**. Что такое неявный способ подключения разделяемой (динамической) библиотеки? Опишите алгоритмы неявного связывания с использованием clang и CMake. Какое связывание лежит в основе неявного подключения? Что такое библиотека импорта? Что такое раздел экспорта? Какие способы экспорта функций существуют? Как их реализовать?

10. **Библиотеки.**. Что такое явный способ подключения разделяемой (динамической) библиотеки? Опишите алгоритмы явного связывания с использованием clang и CMake. Какое связывание лежит в основе явного подключения? Что такое раздел импорта? Что такое name mangling? Как его избежать?

11. **Библиотеки.** Общий алгоритм загрузки и очистки разделяемой (динамической) библиотеки в/из памяти. Функции жизненного цикла разделяемых библиотек в Windows и Linux. Что такое DLL Injection? Алгоритм внедрения DLL в Windows с помощью удаленных потоков. Алгоритм внедрения SO в Linux.

12. **Registry.**. Что такое реестр Windows? Каковы причины его возникновения? В каких случаях стоит использовать реестр? Каких видов бывают данные в реестре? Опишите структуру реестра. Какие типы данных поддерживаются в реестре? Каковы ограничения? Назовите пять основных ульев и опишите их назначение. Опишите API для работы с реестром.

13. **COM.** Что такое Component Object Model (далее – COM)? Два свойства лежащих в основе COM? Почему COM называют двоичным стандартом? Что такое COM-компонент? Что такое COM-интерфейс? Два типа COM-интерфейсов. Чем характеризуется COM-интерфейс? Назовите два стандартных COM-интерфейса. Что такое GUID? CLSID? IID?

14. **COM.** Какие типы COM-контейнеров бывают? Что такое COM-сервер? Что такое COM-клиент? Назовите типы COM-серверов. Что такое «однокомпонентные» и «многокомпонентные» COM-сервера? Что должен «знать» COM-клиент, чтобы использовать COM-объект? Алгоритм создания «однокомпонентного» COM-сервера. Что такое IDL?

15. **COM.** Интерфейс IUnknown. Перечислите методы интерфейса IUnknown и поясните их назначение. Что такое «счетчик ссылок на интерфейсы»? Для чего он нужен? Каким образом и когда этот счетчик увеличивается и уменьшается? Какое соглашение о вызове и возврате должен обеспечивать метод COM-объекта? Какие методы являются исключением? Поясните назначение типа HRESULT и его структуру.

16. **COM.** Интерфейс IClassFactory. Что такое «фабрика классов» и для чего она нужна? Перечислите методы интерфейса IClassFactory и поясните их назначение. Поясните назначение «счетчика экземпляров компонент». Где этот счетчик увеличивается и где уменьшается? Назовите условие, при котором объект компонента удаляется. Опишите жизненный цикл COM-сервера в целом.

17. **COM.** Объясните в чем заключается процесс регистрации COM-объекта? Поясните назначение утилиты regsvr32 и принцип ее работы. Перечислите пять функций, которые экспортируются COM/DLL-контейнером. Поясните назначение этих функций. Работа с памятью в COM и почему она такая?

18. **Сервисы.** Что такое сервис? Виды сервисов. Характеристики сервисов. Что такое SCM? Для чего он предназначен? Опишите структуру сервиса. Какова особенность точки входа сервиса? Что такое функция обратного вызова? Где и какая хранится информация о сервисах Windows? Что такое группа порядка загрузки?

19. **Сервисы.** Что такое демон? Опишите и поясните алгоритм создания процесса-демона вручную. Рекомендации при создании демонов. Что такое systemd и init? Опишите процесс создания сервиса на примере systemd или init.

20. **Драйверы.** Что такое драйвер? Какое место занимает драйвер в структуре ОС? Основные концепции драйверов. Что такое подсистема ввода/вывода? Какие функциональные возможности она предоставляет? Перечислите из чего состоит подсистема ввода/вывода?

21. **Драйверы.** Что такое драйвер устройства? Что такое диспетчер ввода/вывода? Какого его назначение? Что такое PnP-диспетчер и каково его назначение? Что такое диспетчер электропитания? Для чего используется реестр в случае с драйверами и что такое INF-файлы? Что такое HAL?

22. **Драйверы.** Что такое драйвер? Опишите «жизненный цикл» IRP. Что такое виртуальные файлы? Особенности программирования на уровне ядра. Что такое уровни запросов прерываний? Что такое отложенные вызовы процедур? Поясните эти две концепции на примере.

23. **Драйверы.** Что такое драйвер? Какие бывают драйверы? Что такое WDM-драйверы и какие они бывают? Что такое стек драйверов? Какие бывают многоуровневые WDM-драйверы? Опишите последовательность вызова функционала, реализованного многоуровневым драйвером.

24. **Драйверы.** Что такое драйвер? Кто занимается запуском драйвера? Что для этого требуется: перечислите и поясните назначение. Какие дополнительные возможности может включать в себя драйвер? Что такое объекты драйвера и файла и зачем они нужны? Что такое файловый объект?

25. **Драйверы.** Что такое пакет запроса на ввод/вывод (далее – IRP)? Какие бывают IRP? Опишите их. Что такое Plug and Play (далее – PnP)? Какие возможности

предоставляет ПО с поддержкой PnP? Из чего состоит система PnP? С чем может работать PnP? Какие условия драйвер должен выполнить для осуществления полной поддержки PnP?

26. **SEH.** Что такое исключение? Сравните их с прерываниями. Что такое Structured Exception Handling (далее – SEH)? Что такое блок исключения? Какие основные возможности предоставляет SEH? Что такое защищённый блок? Поясните принципы работы обработчика завершения. Что такое локальная раскрутка? Как избежать локальной раскрутки? Причины, по которым следует применять обработчики завершения?

27. **SEH.** Что такое исключение? Что такое аппаратное и программное исключения? Что такое защищённый блок? Поясните принципы работы обработчика исключений. Что такое фильтры? Какие есть стандартные фильтры и как они работают? Что такое глобальная раскрутка? Как возбудить исключения в SEH? Что такое необработанное исключение?

28. **Безопасное программирование.** Что такое безопасное программирование? Какова его цель? Что такое уязвимость? Что такое недостаток программы? Классификация уязвимостей. Категории ошибок ПО. Список распространенных ошибок ПО. Поясните ошибку переполнения буфера и как её можно избежать. Поясните ошибку целочисленного переполнения и как её можно избежать.

29. **Безопасное программирование.** Поясните ошибку форматирования строк и как её можно избежать. Что такое канонизация, валидация и очистка? Что такое триада CIA? Какие ещё есть способы повышения безопасности в рамках ОС? Что такое ASLR? Что такое DEP? Что такое PoLP и какие аспекты лежат в его основе? Какие есть лучшие практики в области безопасного программирования?

30. **Управление доступом.** Что такое контроль доступа к ресурсам? Что такое объекты и субъекты? Понятия политики безопасности и менеджера безопасности и связь между ними. Разница между правами и привилегиями? Порядок разработки политики безопасности. Что такое модель безопасности? Что такое состояние системы безопасности и какова общая задача системы безопасности?

31. **Управление доступом.** Что такое дискреционная политика безопасности и в чём её суть? Алгоритм её построения. Что такое матрица управления доступом? Что такое режимы доступа к объекту? Режимы управления объектами? Опишите модели управления в дискреционной модели безопасности. Опишите два подхода к хранению матрицы управления.

32. **Управление доступом.** Что такое маркер доступа? Что такое охраняемые объекты? Что такое дескриптор безопасности? Из чего он состоит? Поясните понятия DACL, SACL. Что такое учётная запись пользователя? Какие бывают? Что такое группа пользователей? Какие бывают? Что такое SID и какова его структура?

33. **Перехват API.** Что такое перехват API-функций? Поясните как происходит выполнение кода программы в ОС. Что такое функция? Что происходит при вызове

функции? Что такое стек вызовов и каков принцип его работы? Что такое стековый кадр? Что такое соглашение о вызовах и почему они важны при перехвате? Перечислите и кратко опишите какие существуют соглашения о вызовах.

34. **Перехват API.** Что такое перехват API-функций? Перечислите основные методы перехвата. Разделите их по критерию режима выполнения. Расскажите всё о перехвате API-вызовов путём модификации исходного кода: сплайсинг, трамплин, шелл-код, встраиваемый хук. Расскажите всё о перехвате API-вызовов путём модификации таблиц импорта. Для чего может использоваться перехват API-функций?

35. **Перехват API.** Что такое перехват API-функций? Расскажите всё о перехвате API-вызовов путём модификации системных таблиц: суть, алгоритм, что такое SSDT. Расскажите всё об использование драйверов-фильтров для перехвата. Сложности перехвата в Windows. Для чего может использоваться перехват API-функций?

36. **Оптимизация кода.** Что такое оптимизация кода? Какие характеристики могут быть оптимизированы? Стоит ли оптимизировать код вручную? Основные принципы проведения оптимизации. Ключевые аспекты связи оптимизации и системного программирования. Какие уровни оптимизации существуют?

37. **Оптимизация кода.** Что такое оптимизация кода? Принципы оптимизации кода компилятором. Что такое анализ потока данных? Что такое базовые блоки? Зачем они нужны? Что такое упрощаемые графики потоков? Какая программа будет упрощаемой? Перечислите основные типы оптимизаций компилятора.

38. **Оптимизация кода.** Перечислите и поясните основные типы оптимизаций компилятора. Что такое вычисления по короткой схеме? Примерная иерархия скорости выполнения операторов процессором. Что такое безопасные оптимизации? Что такое блокировщики оптимизации? Какие существуют блокировщики? Что такое локальность данных? Какая бывает локальность?

39. **Виртуализация.** Что такое виртуализация? Понятие монитора виртуальных машин. Пример «виртуализации»ресурсов в рамках ОС. Что такое виртуальная машина? Что такое гостевая и хост системы? В каких направлениях должен развиваться гипервизор? Какие бывают гипервизоры и как они работают? Что является центральной концепцией в любом виде виртуализации? Какие виды виртуализации вы знаете?

40. **Виртуализация.** Что такое виртуализация? Какие виды виртуализации на глобальном уровне вы можете выделить? На какие типы делится клиентская виртуализация? Серверная? Поясните принципы контейнеризации, полной виртуализации и паравиртуализации? Виртуализация хранилищ данных.