**Экзаменационные вопросы**

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. Введение. Что такое операционная система (далее – ОС)? Для чего предназначена ОС? Что такое POSIX? Какие абстракции над железом даёт ОС разработчику ПО? Что такое ядро ОС? Каковы задачи ядра ОС? Что такое пользовательский режим и режим ядра?
4. Введение. Что такое операционная система (далее – ОС)? Что такое системный вызов? Что такое ловушки? Что такое прерывания? Что такое объекты ядра ОС? Какие объекты вы знаете? Что такое дескриптор?
5. Процессы. Что такое процесс? Какие ресурсы процесса вам известны? Контекст процесса: определение, назначение и состав. Понятие дочернего и родительского процессов. Что такое процессная многозадачность? В каких случаях лучше всего прибегать к процессной многозадачности?
6. Процессы. Межпроцессное взаимодействие (далее – IPC): определение, классификация механизмов. Общие концепции передачи данных. В каких ситуациях предпочтительнее использовать те или иные IPC механизмы взаимодействия?
7. Процессы. Что такое действие? Контекст действия? Что такое атомарное действие? Что такое синхронизация процессов? Какие механизмы синхронизации вам известны? В каких ситуациях предпочтительнее использовать те или иные механизмы синхронизации?
8. Потоки. Что такое поток выполнения? Контекст потока: определение, назначение и состав. Что такое потокобезопасная функция? Какая функция будет называться реентерабельной? Что такое Thread Local Storage? В каких случаях лучше всего прибегать к потоковой многозадачности?
9. Память. Что такое физическая, логическая и виртуальная память? Как они связаны между собой? Опишите страничную организацию памяти. Что такое рабочее множество страниц процесса? Какие виды памяти вам известны? В каких ситуациях предпочтительнее работать с виртуальной памятью напрямую?
10. Память. Перечислите и поясните назначение секций адресного пространства процесса? Куча: определение, принцип работы. Стек: определение, принцип работы. Советы при работе с кучей и стеком. В каких ситуациях предпочтительнее работать с кучами вместо виртуальной памяти?
11. Память. Виды памяти уровня пользователя. Виды памяти уровня ядра. Два системных пула памяти. Что такое тегирование пула и зачем оно необходимо? Что такое резервные списки? Общие советы при работе с памятью.
12. Файлы. Хранение данных на магнитном диске (далее – МД): основные понятия. Что такое доступ к данным? Как получить доступ к данным на МД? Виды доступа к данным. Поясните понятия сектора и кластера. Из чего они состоят? Каковы их приблизительные аналоги на твердотельных накопителях? Что такое файл на физическом уровне?
13. Файлы. Что такое файловая система (далее – ФС)? Что такое файл на прикладном уровне? Что такое указатель файла? Перечислите основные функции ФС над файлами. Что такое каталог? Корневой каталог? Текущий каталог? Перечислите основные функции ФС над каталогами.
14. Файлы. Что такое буферы ввода-вывода? Для чего они предназначены? Что такое кэширование? Что подразумевает кэширование ввода данных?
15. Файлы. Что такое отображение файла в память? Алгоритм действий при работе с файлами, отображенными в память? Как устроен данный механизм в Windows? В Linux? Когда следует использовать данный механизм? Когда не следует?
16. 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.
17. Библиотеки. Что такое статическая библиотека? Какое связывание лежит в основе статических библиотек? Как собрать приложение с использованием статических библиотек? Преимущества и недостатки статических библиотек.
18. 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? Преимущества и недостатки динамических библиотек.
19. Библиотеки. Что такое неявный способ подключения разделяемой (динамической) библиотеки? Опишите алгоритм работы. Какое связывание лежит в основе неявного подключения? Что такое библиотека импорта? Что такое раздел экспорта? Какие способы экспорта функций существуют? Как их реализовать?
20. Библиотеки. Что такое явный способ подключения разделяемой (динамической) библиотеки? Опишите алгоритм работы. Какое связывание лежит в основе явного подключения? Что такое раздел импорта? Что такое name mangling? Как его избежать?
21. 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.
22. СOM. Что такое Component Object Model (далее – COM)? Два свойства лежащих в основе COM? 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?
23. СOM. 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?
24. СOM. 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 и структуру HRESULT.
25. СOM. 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.
26. СOM. 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.
27. Сервисы. Что такое сервис? Виды сервисов. Характеристики сервисов. Что такое SCM? Для чего он предназначен? Опишите структуру сервиса. Что такое функция обратного вызова? Где хранится информация о сервисах Windows?
28. 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.
29. 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?
30. Драйверы. 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?
31. Драйверы. Что такое пакет запроса на ввод/вывод (далее – IRP)? Опишите «жизненный цикл» IRP. Что такое виртуальные файлы? Что такое уровни запросов прерываний? Что такое отложенные вызовы процедур? Поясните эти две концепции на примере.
32. 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.
33. 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?
34. 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?
35. Перехват API. 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? Перечислите и кратко опишите какие существуют соглашения о вызовах.
36. 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. Расскажите всё о перехвате API-вызовов путём модификации таблиц импорта. Для чего может использоваться перехват API-функций?
37. Перехват API. Что такое перехват API-функций? Расскажите всё о перехват API-вызовов путём модификации системных таблиц. Расскажите всё об использование драйверов-фильтров для перехвата. Сложности в перехвате в Windows. Для чего может использоваться перехват API-функций?
38. Безопасное программирование. Что такое безопасное программирование? Какова его цель? Что такое уязвимость? Что такое недостаток программы? Классификация уязвимостей. Категории ошибок ПО. Список распространённых ошибок ПО. Поясните ошибку переполнения буфера и как её можно избежать. Поясните ошибку целочисленного переполнения и как её можно избежать.
39. Безопасное программирование. Поясните ошибку форматирования строк и как её можно избежать. Что такое валидация и очистка? На чём основывается достижение безопасности ОС? Какие ещё есть способы повышения безопасности? Что такое ASLR? Что такое DEP? Что такое PoLP и какие принципы лежат в его основе?
40. SEH. Что такое исключение? Сравните исключения и прерывания. 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? Причины, по которым следует применять обработчики завершения?
41. 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?
42. Виртуализация. Что такое BPF? Каковы преимущества использования BPF? Что такое трассировка? Что такое инструменты выборки? Что такое наблюдаемость? Виды программ в eBPF. Что такое CRIU? Что такое checkpoint dumping? Зачем он нужен?
43. Виртуализация. Что такое виртуализация? Перечислите и опишите типы виртуализации. Из каких компонент состоит виртуализация? Опишите гипервизоры первого и второго типов. Что такое контейнеризация? Принципы работы и основные компоненты контейнеризации. Namespaces и cgroups в Linux.
44. 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?
45. 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.
46. 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?