

Программа экзамена

Операционные системы

Мехмат, фундаментальная информатика и информационные технологии, дневное отделение

1. *Введение.* Определение и основные особенности операционной системы. Основные компоненты операционных систем. Ядро, его функции. Процесс, адресное пространство, многозадачность. Виды многозадачности. Представление процесса внутри операционной системы. Планирование исполнения задач, поток, его преимущества. Прерывание. Виды прерываний. Режимы исполнения процессов. Пример: уровни привилегий в архитектуре X86. Виды управления памятью. Виртуальная память, её преимущества. Отображение файла в память. Драйвер устройства. Концепции организации ядра, их особенности, примеры реализаций. Гипервизор. Избирательное управление доступом. Источники определения прав доступа. Основные документы, определяющие степень безопасности операционных систем. Требования классов безопасности C1 и C2. Основные особенности и история развития операционных систем на примерах OS/360, CP/M, MS-DOS, UNIX, BSD, Linux, macOS, Windows. Институт инженеров электротехники и электроники, проект GNU, стандарт POSIX. Операционные системы, совместимые с POSIX. Командная оболочка.
2. *Интерфейс прикладных программ и процессы.* Код возврата процесса, стандартные потоки ввода/вывода. Виды библиотек и способы подключения разделяемых библиотек. Проблемы использования разделяемых библиотек. Атрибуты процесса, определяемые при его запуске. Сценарии завершения процесса. Состояния процесса. Прикладные программные интерфейсы. Создание и завершение процесса в интерфейсах Windows API и POSIX. Сигнал POSIX. Ожидание завершения дочернего процесса в POSIX. Дерево процессов.
3. *Виртуальная файловая система POSIX.* Виртуальная файловая система. Канал. Идентификатор пользователя/группы POSIX. Основные особенности структуры POSIX-совместимой виртуальной файловой системы. Абсолютный и относительный путь. Краткая характеристика физических файловых

систем на примерах ext4, NTFS, FAT, ISO 9660, UDF. Процесс и точка монтирования. Специальная файловая система, примеры. Жёсткая и символьная ссылка, ограничения жёстких ссылок. Типы файлов виртуальной файловой системы. Виды файлов устройств. Классы пользователей, права классов пользователей. Файловый и индексный дескриптор, открытый файл. Информация, связанная с индексным дескриптором и открытым файлом. Открытие файла, получение информации об открытом файле, чтение/запись информации из/в открытый файл, перемещение указателя внутри открытого файла, закрытие открытого файла в интерфейсе POSIX. Основные операции с индексными дескрипторами в интерфейсе POSIX. Отображение файла в память в интерфейсе POSIX.

4. *Потоки.* Основные области применения потоков. Основной цикл обработки сообщений в программе с графическим интерфейсом пользователя. Политика планирования, виды планирования. Критерии планирования. Голодная смерть. Простейшие алгоритмы планирования, их характеристики с точки зрения критериев планирования. Классы приоритетов и уровни приоритетов в Windows. Диспетчер ядра. Приоритеты потоков в Windows. Состояния потоков в Windows и переходы между ними. База данных диспетчера ядра. Ситуации, приводящие к вытеснению потока в Windows. Поток простоя. Динамический приоритет. Инверсирование приоритета. Правила повышения динамического приоритета в Windows. Выбор размера кванта планирования. Планирование на многопроцессорных системах. Состояния процессов в Linux. Функции POSIX добровольного переключения контекста, изменения приоритета и политики планирования. Политики планирования реального времени в Linux. Классификации процессов в планировщике с постоянным временем работы. Алгоритм «полностью справедливого планировщика». Создание и ожидание завершения потоков в интерфейсах Windows API и POSIX.
5. *Синхронизация потоков.* Ожидание наступления условия при помощи цикла. События Windows API, области их применения и проблемы. Условие гонки, атомарность, критическая секция, взаимная блокировка. Критические секции и мьютексы. Блокировка с двойной проверкой. Барьер по памяти. Семафоры, организация кольцевого буфера чтения/записи. Барьер. Задача о читателях и писателях. Блокировки чтения/записи. Инвариант, предикат. Условные переменные. Ложное и украденное пробуждение. Задача производителей и потребителей, её решение при помощи условных переменных.
6. *Алгоритмы блокировок.* Сеть Петри. Функции инцидентности, порождённые сетью Петри. Правила срабатывания переходов в сети Петри. Функционирование сети Петри. Примеры сетей Петри. Последовательность срабатываний. Живучесть переходов и сети. Мёртвая и живая блокировка. Задача об обедающих философах. Требования к реализации взаимных исключений.

Программные алгоритмы критической секции со строгим чередованием, попаданием в раздел двух процессов одновременно, мёртвой и живой блокировкой. Алгоритмы Деккера и Петерсона. Доказательство корректности алгоритма Петерсона. Алгоритм Лампорта. Алгоритмы с аппаратной реализацией атомарных операций «проверка и установка» (вариант с проверкой и «проверкой и установкой») и «обмен». Преимущества и недостатки алгоритмов с аппаратными атомарными операциями. Алгоритмы реализации семафоров. Реализации блокировок чтения/записи с возможностью и без возможности голодной смерти «писателя» при помощи семафоров. Возможные сценарии исполнения алгоритма блокировок чтения/записи. Проблемы, вызываемые блокировками. Условия возникновения мёртвой блокировки, методы решения проблемы мёртвой блокировки. Безопасные с точки зрения блокировок состояния.

7. *Межпроцессное взаимодействие.* Межпроцессный обмен дескрипторами в Windows. Совместное использование объектов синхронизации несколькими процессами в интерфейсах Windows API и POSIX. Использование разделяемой памяти в интерфейсах Windows API и POSIX. Использование именованных каналов в интерфейсах Windows API и POSIX. Реализация конвейера командной оболочки и функций `open()`/`pclose()` в POSIX. Использование именованных каналов.
8. *Управление памятью.* Задачи управления памятью. Модель однородного распределения памяти, пример: компьютер ZX Spectrum 48. Организация памяти в виде банков, пример: компьютер ZX Spectrum 128. Блок управления памятью. Классификация видов управления памятью. Виды фрагментации. Техника управления памятью «база и граница». Фиксированное распределение памяти. Фиксированное количество задач в системе OS/360. Версии ядра OS/360. Механизм ключей защиты. Сегмент. Реализация сегментной памяти в архитектуре 8086. Короткая и длинная адресация. Загрузка исполняемого модуля в MS-DOS. Режимы адресации в процессоре 80206. Селектор. Сегментные регистры. Процесс трансляции адресов в 80286. Поля дескриптора сегмента, виды сегментов. Алгоритм работы блока сегментации. Использование сегментации в архитектурах 80386 и x86-64. Использование сегментации в Linux, Windows 3.1 и Windows NT. Достоинства и недостатки сегментной памяти. Страница, страничный кадр. Процесс трансляции адресов в 80386. Поля таблицы страниц. Алгоритм работы блока управления страницами. Буфер ассоциативной трансляции. Режим расширения размера страниц. Механизм расширения физических адресов. Структура логического адреса в архитектуре x86-64. Использование страничной адресации в Windows 3.1. Состояния страниц в Windows NT. Копирование при записи. Последовательность изменения состояния страницы. Структура адресного пространства процесса. Основные виды секций модуля, их отображение в адресное пространство.

Организация виртуального адресного пространства процесса в 32-битной и 64-битной версии Windows. Виды данных в адресном пространстве. Использование больших страниц в Windows. Организация виртуального адресного пространства процесса в Linux. Виды оперативной памяти ядра, правила удовлетворения запросов на выделения памяти со стороны ядра и пользователя. Причины возникновения исключений при обращении к памяти. Алгоритм обработки исключения при обращении к странице. «Жадная» и «ленивая» реализация выделения памяти при создании процесса. Выделение страниц по требованию. Утилизация страничных кадров. Виды страниц с точки зрения алгоритма утилизации.

9. *Физические файловые системы.* Сектор, кластер, метаданные. Адресация секторов на физическом уровне. Разделы дисков. Главная загрузочная запись, загрузочная запись тома, расширенная загрузочная запись. Таблица разделов GUID. Версии файловой системы FAT. Организация метаданных в FAT. Запись таблицы размещения файлов. Запись каталога. Удаление файла в FAT, условия возможности восстановления файла. Загрузочный сектор. Параметр блока BIOS. Сектор информации о файловой системе. Информация в первых двух записях таблицы размещения файлов. Поддержка длинных имён файлов в FAT. Преимущества и недостатки системы FAT. Проблемы внешней фрагментации и способы её избежания в различных файловых системах. Основные возможности, поддерживаемые системой NTFS. Журналируемая файловая система.
10. *Безопасность.* Идентификаторы пользователя и группы в Linux. Виды идентификаторов. Получение и установка идентификаторов. Изменение системных файлов обычными пользователями при помощи атрибута `setuid`. Права пользователей на выполнение общесистемных операций в Linux. Способности процессов. Модули безопасности Linux. Перехватчик. Контекст безопасности. Маркер доступа. Право. Привилегия. Домен Windows. Идентификаторы безопасности, объекты, идентифицируемые ими. Агент, виды агентов. Структура идентификатора безопасности, правила его назначения. Общеизвестные RID и SID. Монитор состояния защиты. Олицетворение. Основной маркер. Маркер олицетворения. Функции Получения маркера пользователя, олицетворения, запуска процесса от имени пользователя. Элемент управления доступом. Список управления доступом. Список управления избирательным доступом. Системный список управления доступом. Дескриптор защиты. Уровни целостности. Условия и правила создания ограниченного маркера. Настройки манифеста приложения, влияющие на запрос повышения прав доступа. Алгоритм вывода запроса повышения прав доступа. Функция запуска программы, выводящая запрос повышения прав.