МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ «РЯЗАНСКИЙ ГОСУДАРСТВЕННЫЙ РАДИОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ ИМЕНИ В.Ф. УТКИНА»

Кафедра «Вычислительная и прикладная математика»

ОЦЕНОЧНЫЕ МАТЕРИАЛЫ ПО ДИСЦИПЛИНЕ

«Операционные системы»

Направление подготовки
09.03.04 «Программная инженерия»

Направленность (профиль) подготовки
«Программная инженерия систем искусственного интеллекта»

Уровень подготовки – бакалавриат

Квалификация выпускника – бакалавр

Форма обучения – очная

Срок обучения – 4 года

Рязань

1. ОБЩИЕ ПОЛОЖЕНИЯ

Оценочные материалы – это совокупность учебно-методических материалов и процедур, предназначенных для оценки качества освоения обучающимися данной дисциплины как части основной образовательной программы.

Цель – оценить соответствие знаний, умений и уровня приобретенных компетенций, обучающихся целям и требованиям основной образовательной программы в ходе проведения текущего контроля и промежуточной аттестации.

Основная задача – обеспечить оценку уровня сформированности компетенций и индикаторов их достижения, приобретаемых обучающимся в соответствии с этими требованиями.

Контроль знаний обучающихся проводится в форме текущего контроля и промежуточной аттестации.

Текущий контроль успеваемости и промежуточная аттестация проводятся с целью определения степени усвоения учебного материала, своевременного выявления и устранения недостатков в подготовке обучающихся, организации работы обучающихся в ходе учебных занятий и оказания им индивидуальной помощи.

К контролю текущей успеваемости относятся проверка знаний, умений и навыков обучающихся на практических занятиях по результатам выполнения и защиты обучающимися индивидуальных заданий, по результатам выполнения контрольных работ и тестов, по результатам проверки качества конспектов лекций и иных материалов.

В качестве оценочных средств на протяжении семестра используется устные и письменные ответы студентов на индивидуальные вопросы, письменное тестирование по теоретическим разделам курса, реферат. Дополнительным средством оценки знаний и умений студентов является отчет о выполнении практических заданий и его защита.

По итогам курса обучающиеся сдают экзамен. Форма проведения – устный ответ с письменным подкреплением по утвержденным билетам, сформулированным с учетом содержания дисциплины. В билет для экзамена включается два теоретических вопроса и задача. В процессе подготовки к устному ответу студент должен составить в письменном виде план ответа.

2. Перечень компетенций с указанием этапов их формирования

При освоении дисциплины формируются следующие компетенции: ОПК-2 (индикаторы ОПК-2.1, ОПК-2.2), ОПК-5 (индикаторы ОПК-5.1, ОПК-5.2, ОПК-5.3).

Указанные компетенции формируются в соответствии со следующими этапами:

– формирование и развитие теоретических знаний, предусмотренных указанными компетенциями (лекционные занятия, самостоятельная работа студентов);

– приобретение и развитие практических умений предусмотренных компетенциями (практические занятия, самостоятельная работа студентов);

– закрепление теоретических знаний, умений и практических навыков, предусмотренных компетенциями, в ходе решения конкретных задач на занятиях, выполнения индивидуальных заданий на практических занятиях и их защиты, а так же в процессе сдачи экзамена.

3. Показатели и критерии оценивания компетенций (*результатов*) на различных этапах их формирования, описание шкал оценивания

Сформированность каждой компетенции в рамках освоения данной дисциплины оценивается по трехуровневой шкале:

– пороговый уровень является обязательным для всех обучающихся по завершении освоения дисциплины;

– продвинутый уровень характеризуется превышением минимальных характеристик сформированности компетенций по завершении освоения дисциплины;

– эталонный уровень характеризуется максимально возможной выраженностью компетенций и является важным качественным ориентиром для самосовершенствования.

При достаточном качестве освоения более 80% приведенных знаний, умений и навыков преподаватель оценивает освоение данной компетенции в рамках настоящей дисциплины на эталонном уровне, при освоении более 60% приведенных знаний, умений и навыков – на продвинутом, при освоении более 40% приведенных знаний умений и навыков – на пороговом уровне. При освоении менее 40% приведенных знаний, умений и навыков компетенция в рамках настоящей дисциплины считается неосвоенной.

Уровень сформированности каждой компетенции на различных этапах ее формирования в процессе освоения данной дисциплины оценивается в ходе текущего контроля успеваемости и представлено различными видами оценочных средств.

Оценке сформированности в рамках данной дисциплины подлежат компетенции/индикаторы:

Показатели достижения планируемых результатов обучения и критерии их оценивания на разных уровнях формирования компетенций приведены в таблице 1.

Таблица 1. Показатели достижения индикаторов компетенции

1	2	3	4
Компетенция: код по ФГОС 3++, формулировка	**Индикаторы**	**Этап**	**Наименование оценочного средства**
ОПК-2 (09.03.04) Способен понимать принципы работы современных информационных технологий и программных средств, в том числе отечественного производства, и использовать их при решении задач профессиональной деятельности	ЗНАТЬ - современные информационные технологии и программные средства отечественного и иностранного производства, используемые при проектировании и реализации информационных систем различного назначения или их компонентов УМЕТЬ - использовать современные информационные технологии и программные средства отечественного и иностранного производства при решении задач профессиональной деятельности ВЛАДЕТЬ - методиками применения современных информационных технологий и программных средств отечественного и иностранного производства при решении задач профессиональной деятельности	1, 2	Экзамен.
ОПК-5 (09.03.04) Способен инсталлировать программное и аппаратное обеспечение для информационных и автоматизиро-	ЗНАТЬ - порядок и правила инсталляции отечественного и иностранного программного обеспечения для информационных и автоматизированных систем	1, 2	Экзамен.

1	2	3	4
ванных систем	УМЕТЬ - инсталлировать отечественное и иностранное программное обеспечение для информационных и автоматизированных систем ВЛАДЕТЬ - навыками инсталляции программного обеспечения для информационных и автоматизированных систем		

Преподавателем оценивается содержательная сторона и качество материалов, приведенных в отчетах студента по практическим занятиям. Кроме того, преподавателем учитываются ответы студента на вопросы по соответствующим видам занятий при текущем контроле:

– контрольные опросы;

– задания для практических занятий.

Принимается во внимание **знания** обучающимися:

- Базовых принципов построения операционных систем. Основных понятий и абстракции, структуры данных. Событий, переводящие систему в режим ядра, переключение контекста. Управлении памятью: задачи, способы, виртуальная память.Управление памятью Intel;

- Взаимодействий параллельных процессов: проблемы взаимодействия – «гонки» и тупики, решения – способы, алгоритмы. Подсистема ввода-вывода: аппаратные прерывания. Классификация ядер ОС: микро ядерная архитектура;

- ОС UNIX/LINUX. Файловой системы VFS. Структуры VFS и их взаимодействие. Файлы и открытые файлы;

- Ядро ОС Linux: доступа к привелегированной информации, виртуальной файловая системы /proc. Взаимоисключение в ядре. Прерываний, обработчики аппаратных прерываний (верхняя и нижняя половины).

умений:

- программировать демонов в ОС Linux. Написать программу, запускающую на выполнение процесс-демон.

обладание навыками**:**

- Реализации задачи «читатели-писатели» с использованием семафоров и разделяемой памяти;

- Реализации задачи «производство-потребление» с использованием семафоров и разделяемой памяти(программа).

Критерии оценивания уровня сформированности компетенции в процессе выполнения практических работ:

41%-60% правильных ответов соответствует пороговому уровню сформированности компетенции на данном этапе ее формирования;

61%-80% правильных ответов соответствует продвинутому уровню сформированности компетенции на данном этапе ее формирования;

81%-100% правильных ответов соответствует эталонному уровню сформированности компетенции на данном этапе ее формирования.

Сформированность уровня компетенций не ниже порогового является основанием для допуска обучающегося к промежуточной аттестации по данной дисциплине.

Формой промежуточной аттестации по данной дисциплине является экзамен, оце-

ниваемый по принятой в ФГБОУ ВО «РГРТУ» четырехбалльной системе: «неудовлетворительно», «удовлетворительно», «хорошо» и «отлично».

Критерии оценивания промежуточной аттестации представлены в таблице.

Шкала оценивания	Критерии оценивания
«отлично»	**студент должен**: продемонстрировать глубокое и прочное усвоение знаний материала; исчерпывающе, последовательно, грамотно и логически стройно изложить теоретический материал; правильно формулировать определения; уметь сделать выводы по излагаемому материалу; безупречно ответить не только на вопросы билета, но и на дополнительные вопросы в рамках рабочей программы дисциплины; выполнить все практические задания, предусмотренные программой
«хорошо»	**студент должен:** продемонстрировать достаточно полное знание материала; продемонстрировать знание основных теоретических понятий; достаточно последовательно, грамотно и логически стройно излагать материал; уметь сделать достаточно обоснованные выводы по излагаемому материалу; ответить на все вопросы билета; выполнить все практические задания, предусмотренные программой.
«удовлетворительно»	**студент должен:** продемонстрировать общее знание изучаемого материала; знать основную рекомендуемую программой дисциплины учебную литературу; уметь строить ответ в соответствии со структурой излагаемого вопроса; показать общее владение понятийным аппаратом дисциплины; уметь устранить допущенные погрешности в ответе на теоретические вопросы; выполнить все практические задания, предусмотренные программой.
«неудовлетворительно»	**ставится в случае:** невыполнения практических занятий; незнания значительной части пройденного материала; не владения понятийным аппаратом дисциплины; существенных ошибок при изложении учебного материала; неумения строить ответ в соответствии со структурой излагаемого вопроса; неумения делать выводы по излагаемому материалу. Как правило, оценка «неудовлетворительно» ставится студентам, которые не могут продолжить обучение по образовательной программе без дополнительных занятий по соответствующей дисциплине (формирования и развития компетенций, закрепленных за данной дисциплиной). Оценка «неудовлетворительно» выставляется также, если студент после начала экзамена отказался его сдавать или нарушил правила сдачи экзамена (списывал, подсказывал, обманом пытался получить более высокую оценку и т.д.).

4. Типовые контрольные задания или иные материалы

ФОС по дисциплине содержит следующие оценочные средства, позволяющие оценить знания, умения и уровень приобретенных компетенций при текущем контроле и промежуточной аттестации, разбитые по модулям дисциплины:

- перечни экзаменационных вопросов;
- макеты билетов к экзамену.

Средства для оценки различных уровней формирования компетенций по категориям «знать», «уметь», «владеть» обеспечивают реализацию основных принципов контроля, таких, как объективность и независимость, практико-ориентированность, междисциплинарность.

С учетом этого, контрольные вопросы (задания, задачи,) входящие в ФОС, для различных категорий и уровней освоения компетенций имеют следующий вид:

Уровень ЗНАТЬ

Дескрипторы	Пример задания из оценочного средства
современные информационные технологии и программные средства отечественного и иностранного производства, используемые при проектировании и реализации информационных систем различного назначения или их компонентов	1. Дать определение операционной системы, процесса и потока. Перечислить особенности процесса и потока. 2. События, переводящие систему в режим ядра: назначение, особенности. Действия, по управлению процессами, выполняющиеся в режиме ядра. 3. Выбрать и обосновать средства реализации верхней половины обработчика прерывания от клавиатуры в Linux, изменяющие ее функциональность, например, сопровождение нажатия клавишей звуком.
порядок и правила инсталляции отечественного и иностранного программного обеспечения для информационных и автоматизированных систем	

Уровень УМЕТЬ

Дескрипторы	Пример задания из оценочного средства
использовать современные информационные технологии и программные средства отечественного и иностранного производства при решении задач профессиональной деятельности	1. Разбирать коды операционных систем и составлять алгоритмы работы системных функций. 2. Разрабатывать программное обеспечение нулевого уровня привилегий: переводящее компьютер из реального режима в защищенный. Формировать системные таблицы. 3. Использовать средства взаимоисключения для обеспечения корректного взаимодействия параллельных процессов. 4. Разрабатывать обработчики аппаратных прерываний и отложенных действий.
инсталлировать отечественное и иностранное программное обеспечение для информационных и автоматизированных систем	

Уровень ВЛАДЕТЬ

Дескрипторы	Пример задания из оценочного средства
методиками применения современных информационных технологий и программных	1. Разрабатывать управляющую программу нулевого уровня привилегий. Создавать системные таблицы GDT и IDT.

средств отечественного и иностранного производства при решении задач профессиональной деятельности	2. Разрабатывать программы, в которых выполняется взаимодействие параллельных процессов методами взаимоисключения. 3. Разрабатывать обработчики аппаратных прерываний и отложенных действий: тасклеты и очереди работ. Обосновывать их выбор.

Перечни вопросов к экзамену и макеты экзаменационного билета

5 семестр

1. Классификация операционных систем.
2. Виртуальная и иерархическая машины.
3. Операционная система – определение, место ОС в системе программного обеспечения ЭВМ.
4. Ресурсы ВС. Режимы ядра и задачи. Процесс, как единица декомпозиции ВС. Потоки.
5. Прерывания: классификация. Аппаратные прерывания в последовательности обработки запроса ввода-вывода приложения.
6. Понятие процесса – процесс как единица декомпозиции системы. Диаграмма состояния процесса. Переключение контекста.
7. Планирование и диспетчеризация. Классификация алгоритмов планирования.
8. Виртуальная память: управление памятью страницами по запросам, использование гиперстраниц – обоснование.
9. Виртуальная память: управление памятью сегментами по запросам: схема преобразования, способы организации таблиц сегментов.
10. Взаимоисключение и синхронизация процессов и потоков. Семафоры: определение, типы, примеры использования.
11. Управление памятью в архитектуре X86.
12. Аппаратные прерывания: задачи обработчика прерывани я от системного таймера в защищенном режиме.
13. Классификация ядер операционных систем. Особенности ОС с микроядром. Три состояния процесса при передаче сообщения.
14. Виртуальная память: сегментно-страничное распределение памяти по запросам, схема преобразования адреса.
15. Unix: концепция процессов.
16. Процессы в OSUnix: диаграмма состояния процесса, планирование процессов.
17. Процессы Unix: системные вызовы fork(), exec(), wait(). Дескриптор процесса.
18. Взаимодействие процессов в распределенных системах: особенности. RPC: методология, механизмы.
19. Виртуальная память: алгоритмы замещения страниц, рабочее множество, локальное и глобальное замещение.
20. Синхронизация и взаимоисключение в распределенных системах: алгоритмы.

Макет экзаменационного билета

Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение высшего образования «Рязанский государственный радиотехнический университет»
(РГРТУ)

Экзаменационный билет № 3

по дисциплине «Операционные системы – 1 часть»

1. Классификация операционных систем.
2. Управление памятью в архитектуре X86.

Билеты рассмотрены и утверждены на заседании кафедры «___ _____» 20__ г.

6 семестр

1. Взаимодействие процессов в распределенных системах: особенности. RPC: методология, механизмы.
2. Виртуальная память: алгоритмы замещения страниц, рабочее множество, локальное и глобальное замещение.
3. Синхронизация и взаимоисключение в распределенных системах: алгоритмы.
4. Средства взаимодействия параллельных процессов в UnixSystemV.
5. Параллельные процессы: монопольный доступ и взаимоисключение, программная реализация, алгоритм Деккера.
6. Защищенный режим: перевод компьютера из реального режима в защищенный.
7. Взаимодействие параллельных процессов: монитор Хоара «читатели-писатели».
8. Процессы в Unix: запуск новой программы. Процессы – «сироты», «зомби», «демоны».
9. Параллельные процессы: взаимоисключение – аппаратный способ и взаимоисключение с помощью семафоров.
10. Процессы: бесконечное откладывание, тупик, зависание – три негативные ситуации в системе на примере задачи об «обедающих философах».
11. Параллельные процессы: взаимоисключение с помощью мониторов. Определение монитора. Примеры: простой монитор, монитор «кольцевой буфер», «читатели-писатели».
12. Три режима работы компьютера на базе процессоров Intel. Адресация аппаратных прерываний в защищенном режиме.
13. Тупики: условия возникновения тупиков и стратегия Хавендера, исключающая возможность возникновения тупиков.
14. Тупики: обход тупиков с помощью алгоритма Банкира, аппроксимации алгоритма Банкира.
15. Тупики: обнаружение тупиков с помощью модели на основе двудольного графа Холта.
16. Тупики: восстановление работоспособности системы после обнаружения тупика.
17. Система прерываний: типы прерываний и их особенности. Быстрые и медленные прерывания. Обработчики прерываний: деление на верхнюю и нижнюю половины.

18. Обработчики аппаратных прерываний – регистрация в системе, разделение линии IRQ и отложенные действия: softirq, tasklet, workqueue.
19. Файловая подсистема /proc – назначение, особенности, файлы, поддиректории, структура proc_dir_entry.
20. Управление устройствами: абстракция устройств, типы устройств и их идентификация в Linux.
21. Драйверы устройств и обработчики прерываний в Linux. Структура USB-драйвера – structusb_driver, таблица id_table, точки входа в драйвер.
22. Файловая система LinuxVFS. Основные структуры файловой системы VFS и их назначение.

Макет экзаменационного билета

Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение высшего образования «Рязанский государственный радиотехнический университет»
(РГРТУ)

Экзаменационный билет № 1

по дисциплине «Операционные системы – 2 часть»

1. Защищенный режим: перевод компьютера из реального режима в защищенный.

2. Управление устройствами: абстракция устройств, типы устройств и их идентификация в Linux.

Билеты рассмотрены и утверждены на заседании кафедры «___ _____» 20__ г.

Перечень лабораторных работ

*Лабораторная работа 1.1*Дизассемблирование прерывания от системного таймера в режиме V86.

Цель работы. Знакомство со средствами дизассемблирования. Анализ программного кода работающей ОС и функциями int8h.

Задания:

Дизассемблировать код 8-го прерывания, изучить его функции и особенности реализации в конкретной ОС. Познакомиться с префиксной командой lock. Составить алгоритм работы 8-го прерывания от системного таймера в графическом виде про ГОСТу.

*Лабораторная работа 1.2.*Алгоритм работы обработчика прерывания int8h в графическом виде.

Цель работы: Составление по коду обработчика прерывания алгоритма работы аппаратного прерывания в структурированном виде. Структуризация алгоритма в соответствии с теоремой о структурном подходе к программированию Бема и Якопини.

*Задания:*Проанализировать использование префиксной команды lock; необходимость сброса контролера прерывания, установку флага if.

*Лабораторная работа 1.3.*Функции обработчика прерываний от системного таймера в защищенном режиме.

Цель работы Изучение функций обработчиков прерываний от системного таймера для ОС Windows и Unix/Linux.

Задания: определить функции обработчика от системного таймера по тику, по главному тику и по кванту для ОС Windows и Unix/Linux.

*Лабораторная работа 1.4.*Пересчет динамических приоритетов в ОС Windows и Unix.

Цель работы: Изучение способов пересчета динамических для ОС Windows и Unix/Linux.

Задания: определить и описать как пересчитываются динамические приоритеты в ОС Windows и Unix/Linux; показать, как изменяется приоритет процесса при блокировке на различных внешних устройствах и объектах ядра.

*Лабораторная работа 1.5.*Управляющая программа (нулевой уровень привилегий), переводящая систему из реального режима в защищенный и обратно.

Цель работы: Изучение особенностей работы компьютера в защищенном режиме.

Задания: объявить структуру, описывающую дескриптор сегмента в таблице GDT; заполнить таблицу GDT дескрипторами сегментов, которые нужны для работы программы; подсчитать размер доступного адресного пространства.

*Лабораторная работа 1.6.*Управляющая программа (нулевой уровень привилегий), обеспечивающая работу с аппаратными прерываниями от системного таймера и клавиатуры в защищенном режиме.

Цель работы: Изучение особенностей работы компьютера в защищенном режиме.

Задания: объявить структуру, описывающую дескриптор прерывания в таблице IDT; заполнить таблицу IDT дескрипторами прерываний, которые нужны для работы программы – 32 заглушки для обработчиков исключений, дескриптор аппаратных прерываний от системного таймера и клавиатуры; перепрограммировать базовый вектор контроллера прерывания, чтобы обеспечить адресацию дескрипторов аппаратных прерываний в защищенном режиме.

*Лабораторная работа 1.7.*Командная строка UNIX, информация о процессах и файлах.

Цель работы. Освоение основных команд командного процессора: ls, ps, mknod, ln. Изучение управляющих параметров, влияющих на объем информации, выводимой на экран, например, о процессах и файлах. Знакомство с системными демонами, файловой подсистемой.

Задания: Используя помощник man определить, какие опции надо указать в команде, чтобы она выводила на экран требуемую информацию. Запустить параллельные процессы и с помощью команды kill смоделировать ситуацию возникновения процесса-«сироты» и процесса-«зомби»; изучить информацию, которая при этом появляется на экране.

Лабораторная работа 1.8. Работа со специальными файлами: линки, именованные программные каналы- передача сообщений между консолями.

*Цель работы.*подсистемойext*, типами файлов, поддерживаемыми файловой системой, жесткими и гибкими ссылками на файлы, дескриптором файла.

Задания. Создать именованный программный канал и передать с помощью него сообщение между консолями.

*Лабораторная работа 1.9.*Системные вызовы UNIX: fork(), exec(), wait().

Цель работы: Разработка программы, в которой создаются параллельные процессы.

Задания. Написать программу, в которой создаются параллельные процессы, на экран выводятся идентификаторы процесса, процесса-предка, группы процесса. При завершении процесса-предка процесс-потомок «усыновляется» терминальным процессом. Модифицировать программу так, чтобы процесс-предок ждал завершения своих потомков. Модифицировать программу с системным вызовом wait() так, чтобы в ветвях процессов-потомков запускались новые программы.

*Лабораторная работа 2.1.*Взаимодействие параллельных процессов: pipe(), signal().

Цель работы: Знакомство с неименованными программными каналами, обеспечивающими симплексную связь, и с техникой сигналов и возможностью изменять ход выполнения процесса с помощью сигнала.

Задания. Написать программу, в которой создаются параллельные процессы, которые передают сообщения через неименованный программный канал. Модифицировать программу так, чтобы можно было изменять ход выполнения программы с помощью сигнала, например, SIGINT.

*Лабораторная работа 2.2*Реализация задачи «читатели-писатели» с использованием семафоров и разделяемой памяти.

Цель работы: Знакомство и отработка техники использования семафоров для реализации взаимоисключения и разделяемой памяти для передачи сообщений.

Задания. Написать программу, в которой параллельные процессы взаимодействуют по алгоритму Хоара «Читатели-писатели» с использованием набора семафоров и разделяемой памяти.

*Лабораторная работа 2.3*Реализация задачи «производство-потребление» с использованием семафоров и разделяемой памяти(программа).

Цель работы: Знакомство и отработка техники использования семафоров для реализации взаимоисключения и разделяемой памяти для передачи сообщений.

Задания. Написать программу, в которой параллельные процессы взаимодействуют по алгоритму Дейкстры «Производство-потребление» с тремя семафорами.

*Лабораторная работа 2.4*Задачи «читатели-писатели» реализация в ОС Windows с использованием потоков и событий (программа).

Цель работы. Знакомство и отработка техники взаимоисключения с помощью объектов ядра ОС Windows.

Задания. Написать программу, в которой параллельные потоки взаимодействуют по алгоритму монитора Хоара «Читатели-писатели» с использованием таких объектов ядра, как события (event).

*Лабораторная работа 2.5*Правила программирования демонов в ОС Linux. Написать программу, запускающую на выполнение процесс-демон.

Цель работы. Знакомство с процессами – демонами и правилами «демонизации» процесса.

Задания. Написать программу, запускающую процесс – демон. Демон может выводить в системный файл информацию по выбору разработчика.

Лабораторная работа 2.6 Написание программы, выводящей на экран дерево каталогов в ОС Linux: рекурсивно или с помощью итераций.

Цель работы. Освоение системных вызовов работы с директориями, chdir(), stat() и т.п.

Задания. Разработать программу или рекурсивную, или итеративную, которая выводит на экран терминала дерево каталогов. В программе следует использовать короткие имена.

*Лабораторная работа 2.7***Разработка загружаемых модулей ядра.**

*Цель работы***.** Получение навыков разработки загружаемых модулей ядра и техники работы с ними. Знакомство с их структурой и функциями ядра.

Задания. Написать загружаемый модуль ядра «Hello!», загрузить в ядро, показать, что модуль загружен, выгрузить модуль и показать, что его нет в ядре.

Написать три загружаемых модуля ядра, которые взаимодействуют друг с другом, отработать особенности взаимодействия на примере третьего загружаемого модуля, обосновать появление ошибки.

Лабораторная работа 2.8 Разработка виртуальной файловой системы, ее регистрация, дерегистрация, создание корневого каталога и монтирование.

Цель работы. Получение навыков работы с файловыми системами: их объявлением, регистрацией в системе и дерегистрацией, создания каталога и монтирования.

Задания. Написать виртуальную файловую систему. В программе файловая система объявляется, вызывается системные вызовы регистрации файловой системы и дерегистрации файловой системы, создается inode корневого каталога, разрабатываются функции необходимые для монтирования файловой системы.

*Лабораторная работа 4.1*Открытые файлы: анализ буферизованного и не буферизованного ввода-вывода.

Цель работы. Знакомство со структурами, описывающими открытые файлы, проблемами буферизованного ввода-вывода.

Задания: Написать программы, осуществляющие как буферизованный, так и не буферизованный ввод – вывод, проанализировать полученные результаты и их прокомментировать с использование схем, демонстрирующих дескрипторы соответствующих таблиц открытых файлов и их взаимосвязь.

Лабораторная работа 4.2 Виртуальная файловая система proc: создание файла, директории, передача данных из режима пользователя в ядро и из ядра в режим пользователя.

Цель работы. Знакомство в виртуальной файловой системой /proc.

Задания. Написать загружаемый модуль ядра, в котором с помощью структур и функций ядра реализуется работа с виртуальной файловой системой /proc. Модуль должен обеспечивать взаимодействие с режимом пользователя.

*Лабораторная работа 4.3*Аппаратные прерывания: нижняя половина прерывания - тасклеты.

Цель работы. Знакомство с программированием аппаратных прерываний, их регистрацией в системе и нижней половины прерывания – тасклетом.

Задания. Написать загружаемый модуль ядра, объявляющий аппаратное прерывание, регистрирующее его обработчик на разделяемой линии IRQ. Обработчик аппаратного прерывания ставит в очередь на выполнение свою нижнюю половину – тасклет.

*Лабораторная работа 4.4*Аппаратные прерывания: нижняя половина прерывания – очереди работ.

Цель работы. Знакомство с программированием аппаратных прерываний, их регистрацией в системе и нижней половины прерывания – очередью работ.

Задания. Написать загружаемый модуль ядра, объявляющий аппаратное прерывание, регистрирующее его обработчик на разделяемой линии IRQ. Обработчик аппаратного прерывания ставит в очередь на выполнение свою нижнюю половину – очередь работ.

Лабораторная работа 4.5 Взаимодействие параллельных процессов в UnixBSD: сокеты семейства AF_UNIX и SOCK_DGRAM.

Цель работы. Изучить средства взаимодействия параллельных процессов в UnixBSD – sockets.

Задания. Написать программу по модели клиент – сервер, обеспечивающую взаимодействие параллельных процессов на отдельно стоящей машине (мейнфрейме) с помощью сокетов Беркли: семейство Unix, тип DGRAM.

*Лабораторная работа 4.6*Взаимодействие параллельных процессов в UnixBSD для мейнфрейма: сокеты семейства AF_INET и SOCK_STREAM.

Цель работы. Изучить средства взаимодействия параллельных процессов в UnixBSD для распределенных систем – sockets.

Задания. Написать программу по модели клиент – сервер, обеспечивающую взаимодействие параллельных процессов в распределенной системе с помощью сокетов Берк-

ли: семейство INET, тип STREAM.

*Лабораторная работа 4.7*Дизассемблирование прерывания от системного таймера в режиме V86.

Цель работы. Знакомство со средствами дизассемблирования. Анализ программного кода работающей ОС и функциями int8h.

Задания:

Дизассемблировать код 8-го прерывания, изучить его функции и особенности реализации в конкретной ОС. Познакомиться с префиксной командой lock. Составить алгоритм работы 8-го прерывания от системного таймера в графическом виде про ГОСТу.

Лабораторная работа 5.1 Алгоритм работы обработчика прерывания int8h в графическом виде.

Цель работы Составление по коду обработчика прерывания алгоритма работы аппаратного прерывания в структурированном виде. Структуризация алгоритма в соответствии с теоремой о структурном подходе к программированию Бема и Якопини.

Задания: проанализировать использование префиксной команды lock; необходимость сброса контролера прерывания, установку флага if.

*Лабораторная работа 5.2*Функции обработчика прерываний от системного таймера в защищенном режиме.

Цель работы Изучение функций обработчиков прерываний от системного таймера для ОС Windows и Unix/Linux.

Задания: определить функции обработчика от системного таймера по тику, по главному тику и по кванту для ОС Windows и Unix/Linux.

*Лабораторная работа 5.3*Пересчет динамических приоритетов в ОС Windows и Unix.

Цель работы: Изучение способов пересчета динамических для ОС Windows и Unix/Linux.

Задания: определить и описать как пересчитываются динамические приоритеты в ОС Windows и Unix/Linux; показать, как изменяется приоритет процесса при блокировке на различных внешних устройствах и объектах ядра.

*Лабораторная работа 5.4*Управляющая программа (нулевой уровень привилегий), переводящая систему из реального режима в защищенный и обратно.

Цель работы: Изучение особенностей работы компьютера в защищенном режиме.

Задания: объявить структуру, описывающую дескриптор сегмента в таблице GDT; заполнить таблицу GDT дескрипторами сегментов, которые нужны для работы программы; подсчитать размер доступного адресного пространства.

*Лабораторная работа 5.5*Управляющая программа (нулевой уровень привилегий), обеспечивающая работу с аппаратными прерываниями от системного таймера и клавиатуры в защищенном режиме.

Цель работы: Изучение особенностей работы компьютера в защищенном режиме.

Задания: объявить структуру, описывающую дескриптор прерывания в таблице

IDT; заполнить таблицу IDT дескрипторами прерываний, которые нужны для работы программы – 32 заглушки для обработчиков исключений, дескриптор аппаратных прерываний от системного таймера и клавиатуры; перепрограммировать базовый вектор контроллера прерывания, чтобы обеспечить адресацию дескрипторов аппаратных прерываний в защищенном режиме.

Лабораторная работа 6.1 Взаимодействие параллельных процессов в UnixBSD: сокеты семейства AF_UNIX и SOCK_DGRAM.

Цель работы. Освоение основных команд командного процессора: ls, ps, mknod, ln. Изучение управляющих параметров, влияющих на объем информации, выводимой на экран, например, о процессах и файлах. Знакомство с системными демонами, файловой подсистемой.

Задания: Используя помощник man определить, какие опции надо указать в команде, чтобы она выводила на экран требуемую информацию. Запустить параллельные процессы и с помощью команды kill смоделировать ситуацию возникновения процесса-«сироты» и процесса-«зомби»; изучить информацию, которая при этом появляется на экране.

Лабораторная работа 6.2 Взаимодействие параллельных процессов в UnixBSD: сокеты семейства AF_UNIX и SOCK_DGRAM..

Цель работы: Разработка программы с.подсистемой ext*, типами файлов, поддерживаемыми файловой системой, жесткими и гибкими ссылками на файлы, дескриптором файла.

Задания. Создать именованный программный канал и передать с помощью него сообщение между консолями.

Лабораторная работа 6.3 Мультиплексирование ввода-вывода: сокеты семейства AF_INET и SOCK_STREAM и мультиплексирование

Цель работы: Разработка программы, в которой создаются параллельные процессы.

Задания. Написать программу, в которой создаются параллельные процессы, на экран выводятся идентификаторы процесса, процесса-предка, группы процесса. При завершении процесса-предка процесс-потомок «усыновляется» терминальным процессом. Модифицировать программу так, чтобы процесс-предок ждал завершения своих потомков. Модифицировать программу с системным вызовом wait() так, чтобы в ветвях процессов-потомков запускались новые программы.