



РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ
Б1.О.02.08 Современные операционные системы

Направление подготовки: 01.04.02 Прикладная математика и информатика

Направленность (профиль): Информационное обеспечение экономической деятельности

Квалификация (степень): *магистр*

Форма обучения: *очная*

Институт: математики, естествознания и техники

Кафедра: математического моделирования и компьютерных технологий

	очная форма	очно-заочная форма	заочная форма
Курс	1		
Семестр/триместр	1, 2		
Лекций	18		
Лабораторных занятий	36		
Практических (семинарских) занятий	18		
Консультации	-		
Форма(ы) промежуточной аттестации	Зачет в 1 семестре Экзамен во 2 семестре – 0.3		
Контроль	9		
Иные формы работы	-		
Самостоятельная работа	206.7		

Всего часов: **288**

Трудоемкость: **8** зачетных единиц.

I. ОРГАНИЗАЦИОННО-МЕТОДИЧЕСКИЙ РАЗДЕЛ

Цель изучения дисциплины:

Целью освоения дисциплины Б1.О.02.08 «Современные операционные системы» является изучение организации и принципов построения современных операционных систем и системных программ; формирование представлений об общей методологии разработки системно-ориентированных программ с использованием современных алгоритмических языков и систем программирования; углубленная подготовка магистров в области применения аппаратных и программных средств современных процессоров, предназначенных для поддержки многозадачных операционных систем.

Задачи изучения дисциплины:

Основной задачей изучения дисциплины Б1.О.02.08 «Современные операционные системы» является формирование у студентов представлений об архитектурном строении современных операционных систем и получение практических навыков работы с ними. В результате изучения курса студенты должны знать теоретические концепции, состав и взаимодействие компонент современных операционных систем, а также иметь практические навыки работы с ними.

Место дисциплины в структуре ОПОП:

Дисциплина Б1.О.02.08 «Современные операционные системы» реализуется в рамках Б1.О.02 Модуль 2 "Предметно-содержательный" обязательной части ОПОП.

Планируемые результаты обучения по дисциплине, соотнесенные с индикаторами достижения компетенций:

Код компетенции	Индикатор компетенции	Планируемые результаты обучения по дисциплине
ОПК-4	Знать: – языки программирования, языки баз данных, алгоритмы, библиотеки и пакеты программ.	Знает: – языки программирования, языки баз данных, алгоритмы, библиотеки и пакеты программ.
	Уметь: – применять в профессиональной деятельности современные информационно-коммуникационные технологии, современные системы программирования, операционные системы, продукты системного и прикладного программного обеспечения, сетевые технологии с учетом требований информационной безопасности.	Умеет: – применять в профессиональной деятельности современные информационно-коммуникационные технологии, современные системы программирования, операционные системы, продукты системного и прикладного программного обеспечения, сетевые технологии с учетом требований информационной безопасности.
	Владеть: – способностью реализовывать решения научно-исследовательских задач с	Владеет: – способностью реализовывать решения научно-исследовательских задач с применением современных информационно-

	применением современных информационно-коммуникационных технологий с учетом требований информационной безопасности.	коммуникационных технологий с учетом требований информационной безопасности.
ПКС-1	Знать: <ul style="list-style-type: none"> – методы и приемы организации процесса разработки баз данных ИС; – методологии разработки баз данных ИС и технологии программирования; – языки формализации функциональных спецификаций; – методы и приемы формализации задач; – методы и средства проектирования программного обеспечения, программных интерфейсов и баз данных ИС. 	Знает: <ul style="list-style-type: none"> – методы и приемы организации процесса разработки программного обеспечения; – методологии разработки программного обеспечения и технологии программирования; – языки формализации функциональных спецификаций; – методы и приемы формализации задач; – методы и средства проектирования программного обеспечения, программных интерфейсов.
	Уметь: <ul style="list-style-type: none"> – использовать существующие типовые решения проектирования баз данных ИС; – применять методы и средства проектирования структур данных, баз данных, программных интерфейсов; – осуществлять коммуникации с заинтересованными сторонами; – выбирать средства реализации требований к базам данных ИС; – вырабатывать варианты реализации баз данных ИС и требований к нему; – проводить анализ исполнения требований. 	Умеет: <ul style="list-style-type: none"> – использовать существующие типовые решения проектирования программного обеспечения; – применять методы и средства проектирования структур данных, баз данных, программных интерфейсов; – осуществлять коммуникации с заинтересованными сторонами; – выбирать средства реализации требований к программному обеспечению; – вырабатывать варианты реализации программного обеспечения и требований к нему; – проводить анализ исполнения требований.
	Владеть: <ul style="list-style-type: none"> – методологией организации процесса разработки баз данных ИС; – методологией и технологиями проектирования программных интерфейсов, структур и баз данных ИС в соответствии с установленными 	Владеет: <ul style="list-style-type: none"> – методологией организации процесса разработки программного обеспечения; – методологией и технологиями проектирования программных интерфейсов, структур и баз данных ИС в соответствии с установленными требованиями; – действиями по разработке и согласованию технических

	требованиями; – действиями по разработке и согласованию технических спецификаций на программные компоненты; – действиями по согласованию требований к базам данных ИС с заинтересованными сторонами, распределению заданий между программистами в соответствии с техническими спецификациями, осуществлению контроля выполнения заданий, формированию отчетности в соответствии с установленными регламентами.	спецификаций на программные компоненты; – действиями по согласованию требований к программному обеспечению с заинтересованными сторонами, распределению заданий между программистами в соответствии с техническими спецификациями, осуществлению контроля выполнения заданий, формированию отчетности в соответствии с установленными регламентами.
--	--	--

II. СОДЕРЖАНИЕ И ОБЪЕМ ДИСЦИПЛИНЫ

с указанием количества часов, выделенных на контактную работу обучающихся с преподавателем (по видам учебных занятий) и на самостоятельную работу

Очная форма обучения

№ п/п	Наименование разделов и тем	Всего	Аудиторные занятия			
			ЛК	ПЗ	ЛБ	Сам. раб.
1	Тема 1. Базовые понятия и теоретические основы технологии применения сетевых операционных систем.	52	6	6	6	34
2	Тема 2. Обзор современных сетевых операционных систем и операционных оболочек.	52	6	6	6	34
3	Тема 3. Мультипрограммирование в системах пакетной обработки, системах разделения времени, системах реального времени. Мультипроцессирование.	51	6	6	6	33
	<i>Контроль</i>	-				
	<i>Консультация</i>	-				
	<i>Зачет</i>	-				
	<i>Итого за 1 семестр</i>	155	18	18	18	101
4	Тема 4. Защитные механизмы операционных систем.	41	-	-	6	35
5	Тема 5. Безопасность операционных	41	-	-	6	35

	систем и сетей.					
6	Тема 6. Операционные системы для облачных вычислений.	41.7	-	-	6	35.7
	Контроль	9				
	Консультация	-				
	Экзамен	0.3				
	Итого за 2 семестр	133	-	-	18	105.7
	ИТОГО:	288	18	18	36	206.7

Очно-заочная форма обучения (не реализуется)

Заочная форма обучения (не реализуется)

III. ОЦЕНОЧНЫЕ МАТЕРИАЛЫ ДЛЯ ПРОВЕДЕНИЯ ТЕКУЩЕЙ И ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ ОБУЧАЮЩИХСЯ ПО ДИСЦИПЛИНЕ

Текущая аттестация проводится в форме контрольной работы.

Типовая контрольная работа

Создать один или несколько пакетных файлов, которые содержат следующие компоненты:

- управление эхо-выводом пакетного файла на экран;
- передача нескольких параметров;
- проверка числа параметров;
- превращение пакетного файла в интерактивный;
- применение в пакетных файлах условий if/then;
- выполнение одного пакетного файла из другого;
- создание меню в пакетных файлах.

Рубежный контроль

1. В некоторой операционной системе, похожей на UNIX, существует единственный способ порождения нового процесса, который будет являться дубликатом родительского процесса по регистровому и пользовательскому контекстам, с помощью системного вызова fork(). Неопытный программист написал следующую программу:

```
void main()
{
    int i;
    for (i = 0; i < n; i++){
        fork();
    }
    while(1);
}
```

где n - некоторая положительная константа. Сколько процессов будет запущено в операционной системе в результате ее выполнения? Дайте обоснование ответа.

2. Пусть в вычислительную систему поступают пять процессов различной длительности со статическими приоритетами по следующей схеме:

Номер процесса	Момент поступления в систему	Время исполнения	Приоритет
1	3	10	1
2	6	4	0
3	0	4	3
4	2	1	4
5	4	3	2

• Вычислите среднее время между стартом процесса и его завершением (turnaroud time) и среднее время ожидания процесса (waiting time) для каждого из трех алгоритмов планирования: FCFS (First Come First Served), RR (Round Robin) и вытесняющее приоритетное планирование. При вычислениях считать, что процессы не совершают операций ввода-вывода, величину кванта времени принять равной 3, временем переключения контекста пренебречь. Для алгоритма RR принять, что вновь прибывший процесс попадает в САМЫЙ конец очереди готовых (после процесса, отработавшего свой квант). Наивысшим приоритетом является приоритет 0.

3. Три процедуры А, В и С, расположенные в разных файлах, были объединены в одну программу с помощью linker'a и загружены в память. Размеры их составляют 900, 1000 и 1300 машинных слов соответственно. Рассмотрим следующие схемы управления памятью:

- Страничная организация (без сегментации). Размер страницы - 1000 слов, таблица страниц занимает одну страницу.
- Сегментная организация (без страниц). Для каждой процедуры используется свой сегмент памяти. Таблица сегментов занимает 1000 слов.
- Сегментно-страничная организация. Комбинация информации из пунктов а) и б).

Для всех трех схем организации определите количество памяти, требующееся программе для ее полной загрузки в оперативную память (т.е. для загрузки процедур и всех необходимых таблиц).

4. В вычислительной системе моделируется отдых пассажиров на прогулочном катере. Катер вмещает N человек и совершает кольцевые прогулки по расписанию через равные промежутки времени. Если к моменту очередного отправления пассажиров нет, катер остается на месте до следующего времени отправления. По окончании прогулки все пассажиры покидают катер. Пассажиры не садятся на катер до выхода прибывших пассажиров. Каждый пассажир моделируется процессом

Процесс i-й пассажир:

```
while (1) {
    <зайти на катер>
    <сойти с катера>
}
```

Катер представляется процессом:

```
<приплыть из гавани>
```

```
while(1){
    do {
```

```

    <ждать время T>
    } while(число пассажиров == 0);
<совершить прогулку>
}

```

Опишите схему организации прогулок, используя семафоры Дейкстры и разделяемые переменные.

5. В вычислительной системе с сегментной организацией памяти из 32-х бит адреса старшие 14 его бит отводятся для номера сегмента.

а. Какое максимальное количество сегментов может иметь процесс? Каков максимальный размер сегмента?

б. Для некоторого процесса таблица сегментов в этой системе имеет вид:

Номер сегмента	Адрес начала сегмента	Длина сегмента
1	0x00000	0x8000
2	0x20000	0x2000
3	0x10000	0x10000
5	0x30000	0x0f000

1. Каким физическим адресам соответствуют адреса 0x45678, 0x170201, 0x1300de?

6. В вычислительной системе со страничной организацией памяти время доступа процессора к оперативной памяти составляет 100 нс, а время доступа к ассоциативной памяти составляет 15 нс. Частота попаданий в ассоциативную память при обращении к данным (hit ratio) составляет 80%. Оцените среднее время доступа к одному данному.

Вопросы к зачету (1 семестр, очная форма обучения)

1. Структура вычислительной системы.
2. История эволюции вычислительных систем.
3. Основные понятия, концепции ОС.
4. Архитектурные особенности ОС.
5. Классификация ОС.
6. Понятие процесса.
7. Состояния процесса.
8. Операции над процессами и связанные с ними понятия.
9. Уровни планирования.
10. Критерии планирования и требования к алгоритмам.
11. Параметры планирования.
12. Вытесняющее и невытесняющее планирование.

13. Алгоритмы планирования First-Come, First-Served (FCFS), Round Robin (RR).
14. Алгоритмы планирования Shortest-Job-First (SJF). Гарантированное планирование.
15. Приоритетное планирование.
16. Многоуровневые очереди. Многоуровневые очереди с обратной связью.
17. Взаимодействующие процессы.
18. Категории средств обмена информацией.
19. Логическая организация механизма передачи информации.
20. Особенности передачи информации с помощью линий связи.
21. Надежность средств связи. Завершение связи.
22. Нити исполнения.
23. Активность, состояние состязания и взаимоисключения.
24. Критическая секция.
25. Требования, предъявляемые к алгоритмам организации взаимодействия процессов. Запрет прерываний. Переменная-замок.
26. Строгое чередование. Флаги готовности. Алгоритм Петерсона.
27. Алгоритм булочной. Аппаратная поддержка взаимоисключений.
28. Команда Test-and-Set. Команда Swap.
29. Семафоры.
30. Мониторы. Сообщения.
31. Реализация мониторов и передачи сообщений с помощью семафоров.
32. Реализация семафоров и передачи сообщений с помощью мониторов. Реализация семафоров и мониторов с помощью очередей сообщений.
33. Условия возникновения тупиков. Основные направления борьбы с тупиками. Игнорирование проблемы тупиков.
34. Способы предотвращения тупиков.
35. Обнаружение тупиков. Восстановление после тупиков.
36. Физическая организация памяти компьютера. Логическая память.
37. Связывание адресов. Функции системы управления памятью.
38. Простейшие схемы управления памятью.
39. Страничная память.
40. Сегментная и сегментно-страничная организация памяти.

Вопросы к экзамену (2 семестр, очная форма обучения)

Теоретическая часть

1. Понятие виртуальной памяти.
2. Страничная виртуальная память, Сегментно-страничная организации виртуальной памяти.
3. Структура таблицы страниц. Ассоциативная память.

4. Инвертированная таблица страниц. Размер страницы.
5. Исключительные ситуации при работе с памятью. Стратегии управления страничной памятью.
6. Алгоритмы замещения страниц.
7. Управление количеством страниц, выделенных процессу. Модель рабочего множества.
8. Страничные демоны. Программная поддержка сегментной модели памяти процесса.
9. Отдельные аспекты функционирования менеджера памяти.
10. Общие сведения о файлах.
11. Организация файлов и доступ к ним.
12. Операции над файлами.
13. Директории. Логическая структура файлового архива.
14. Операции над директориями. Защита файлов.
15. Общая структура файловой системы.
16. Методы выделения дискового пространства.
17. Управление свободным и занятым дисковым пространством.
18. Структура файловой системы на диске.
19. Реализация директорий.
20. Монтирование файловых систем.
21. Связывание файлов.
22. Кооперация процессов при работе с файлами.
23. Надежность файловой системы.
24. Производительность файловой системы.
25. Реализация некоторых операций над файлами.
26. Современные архитектуры файловых систем.
27. Общие сведения об архитектуре компьютера. Структура контроллера устройства.
28. Опрос устройств и прерывания. Исключительные ситуации и системные вызовы.
29. Прямой доступ к памяти (Direct Memory Access – DMA). Логические принципы организации ввода-вывода.
30. Систематизация внешних устройств и интерфейс между базовой подсистемой ввода-вывода и драйверами.
31. Функции базовой подсистемы ввода-вывода.
32. Алгоритмы планирования запросов к жесткому диску.
33. Сетевые и распределенные операционные системы. Взаимодействие удаленных процессов как основа работы вычислительных сетей.
34. Основные вопросы логической организации передачи информации между удаленными процессами.
35. Понятие протокола.

36. Многоуровневая модель построения сетевых вычислительных систем.
37. Проблемы адресации в сети.
38. Проблемы маршрутизации в сетях.
39. Связь с установлением логического соединения и передача данных с помощью сообщений. Синхронизация удаленных процессов.
40. Угрозы безопасности.
41. Формализация подхода к обеспечению информационной безопасности.
42. Криптография как одна из базовых технологий безопасности ОС.
43. Шифрование с использованием алгоритма RSA. Теорема Эйлера.
44. Идентификация и аутентификация.
45. Авторизация. Разграничение доступа к объектам ОС.
46. Выявление вторжений. Аудит системы защиты.
47. Анализ некоторых популярных ОС с точки зрения их защищенности.

Практическая часть

1. Найдите сумму и разность 2-х чисел: 1-е число - номер в группе (переведенное в шестнадцатиричную форму), 2-е - число, противоположное номеру первой буквы фамилии в алфавите (отрицательное число в дополнительном коде). Сумму и разность переведите в десятичную форму.
2. Просмотрите содержимое регистров микропроцессора, а также флагов и выпишите их в протокол.
3. Запишите в регистр AX первое число (из задания 1), а в регистр BX - второе (из задания 1). Введите в оперативную память в сегмент кода (смещение 100) машинную команду сложения регистров AX и BX. Просмотрите на экране ее ассемблерную форму. Выполните эту команду, результат переведите в десятичную форму.
4. Введите в оперативную память в сегмент кода (смещение 100) набор команд ассемблера для распечатки символа на экране - первой буквы вашей фамилии. Проверьте программу в DEBUG. Затем запишите ее на диск в виде COM-файла. Запустите ее на выполнение из DOS.

IV. ПЕРЕЧЕНЬ ЛИТЕРАТУРЫ, НЕОБХОДИМОЙ ДЛЯ ОСВОЕНИЯ ДИСЦИПЛИНЫ

4.1. Основная литература

1. Гостев И.М. Операционные системы: учебник и практикум для академического бакалавриата / И.М. Гостев. — 2-е изд., испр. и доп. — М.: Издательство Юрайт, 2019. — 164 с. — (Серия: Бакалавр. Академический курс). — ISBN 978-5-534-04520-8. — Режим доступа: www.biblio-online.ru/book/1147C542-A66A-4AF1-AA6C-EC76A70E52CB.

4.2. Дополнительная литература

1. Партыка Т.Л. Операционные системы, среды и оболочки / Т.Л. Партыка, И.И. Попов. – изд. 3-е испр. и доп. – М.:ФОРУМ, 2010
2. Синицын С.В. Операционные системы: учебник для студ. высш. учеб. заведений. / С.В. Синицын, А.В. Батаев, Н.Ю. Налютин – М.: Издательский центр «Академия», 2010.
3. Стахнов А. Linux в подлиннике. – BHV, 2010.

V. ПЕРЕЧЕНЬ РЕСУРСОВ ИНФОРМАЦИОННО-ТЕЛЕКОММУНИКАЦИОННОЙ СЕТИ «ИНТЕРНЕТ», НЕОБХОДИМЫХ ДЛЯ ОСВОЕНИЯ ДИСЦИПЛИНЫ

№ пп	Ссылка на информационный ресурс	Наименование разработки в электронной форме	Доступность
1.	http://www.biblioclub.ru	Электронно-библиотечная система (ЭБС) Университетская библиотека онлайн	Свободный доступ
2.	http://edu.ru/	Российское образование: Федеральный портал. Включает ссылки на порталы и сайты образовательных учреждений; государственные образовательные стандарты; нормативные документы; каталог экскурсий и обучающих программ.	Свободный доступ
3.	www.elibrary.ru	Российский информационный портал в области науки, технологии, медицины и образования	Свободный доступ

VI. СОВРЕМЕННЫЕ ПРОФЕССИОНАЛЬНЫЕ БАЗЫ ДАННЫХ И ИНФОРМАЦИОННЫЕ СПРАВОЧНЫЕ СИСТЕМЫ

1.	http://ilib.mccme.ru	ЭБ с книгами по математике	Свободный доступ
2.	https://e.lanbook.com/	ЭБС Лань	Регистрация через компьютер Научной библиотеки ЕГУ. Доступ с компьютеров библиотеки.

VII. ЛИЦЕНЗИОННОЕ И СВОБОДНО РАСПРОСТРАНЯЕМОЕ ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ

При реализации учебной дисциплины применяется следующее лицензионное и свободно распространяемое программное обеспечение: Microsoft Windows 7 Profes-

sional 64-bit, Kaspersky Endpoint Security 11, Smart Notebook 17, а также свободным программным обеспечением: LibreOffice 6.0.

VIII. ОБОРУДОВАНИЕ И ТЕХНИЧЕСКИЕ СРЕДСТВА ОБУЧЕНИЯ, НЕОБХОДИМЫЕ ДЛЯ ОСУЩЕСТВЛЕНИЯ ОБРАЗОВАТЕЛЬНОГО ПРОЦЕССА ПО ДИСЦИПЛИНЕ

Учебные занятия проводятся в аудиториях, укомплектованных специализированной мебелью, в том числе стационарными или переносными техническими средствами обучения (проектор, экран, компьютер/ноутбук).

Лабораторные занятия, групповые и индивидуальные консультации, текущая и промежуточная аттестации проводятся в специализированных классах, оснащенных автоматизированными рабочими местами с компьютерами.

Самостоятельная работа проводится в кабинетах, оснащенных компьютерной техникой с возможностью подключения к сети «Интернет» и обеспечением доступа в электронную информационно-образовательную среду университета.