

МИНОБРНАУКИ РОССИИ
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ
ВЫСШЕГО ПРОФЕССИОНАЛЬНОГО ОБРАЗОВАНИЯ
**«БЕЛГОРОДСКИЙ ГОСУДАРСТВЕННЫЙ
ТЕХНОЛОГИЧЕСКИЙ УНИВЕРСИТЕТ им. В.Г.ШУХОВА»**
(БГТУ им. В.Г. Шухова)



РАБОЧАЯ ПРОГРАММА
дисциплины

Операционные системы

направление подготовки

09.03.02 Информационные системы и технологии

профиль программы

Информационные системы и технологии

Квалификация

бакалавр

Форма обучения

очная

Институт: Информационных технологий и управляющих систем

Кафедра: Информационных технологий

Белгород – 2015

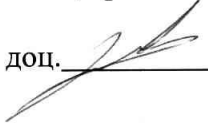
Рабочая программа составлена на основании требований:

- Федерального государственного образовательного стандарта высшего образования по направлению подготовки 09.03.02 Информационные системы и технологии (уровень бакалавриата), утвержденного Приказом Министерства образования и науки Российской Федерации. от 12 марта 2015 г. N 219
- плана учебного процесса БГТУ им. В.Г. Шухова, введенного в действие в 2015 году.

Составитель: ст. преп.  (А.В. Глухоедов)


Рабочая программа обсуждена на заседании кафедры информационных технологий

« 15 » 04 2015 г., протокол № 5

Зав. кафедрой: канд. техн. наук, доц.  (И.В. Иванов)

Рабочая программа одобрена методической комиссией института ИТУС

« 23 » 04 2015 г., протокол № 8/12

Председатель: доц.  (Ю. И. Солопов)

ПЛАНИРУЕМЫЕ РЕЗУЛЬТАТЫ ОБУЧЕНИЯ ПО ДИСЦИПЛИНЕ

Формируемые компетенции			Требования к результатам обучения
№	Код компетенции	Компетенция	
Общепрофессиональные			
	ОПК-6	способность выбирать и оценивать способ реализации информационных систем и устройств (программно-, аппаратно- или программно-аппаратно-) для решения поставленной задачи	<p>В результате освоения дисциплины обучающийся должен:</p> <p>Знать:</p> <ul style="list-style-type: none">– назначение, функции, классификацию и архитектуру операционных систем. <p>Уметь:</p> <ul style="list-style-type: none">– использовать полученные знания по операционным системам для создания программных приложений для современных операционных систем. <p>Владеть:</p> <ul style="list-style-type: none">– навыками, необходимыми для эффективной работы в современных операционных системах.
Профессиональные			
	ПК-15	способность участвовать в работах по доводке и освоению информационных технологий в ходе внедрения и эксплуатации информационных систем	<p>В результате освоения дисциплины обучающийся должен:</p> <p>Знать:</p> <ul style="list-style-type: none">– понятие процесса и потока, управление процессами и потоками, планирование и синхронизацию;– понятие виртуальной памяти;– понятие файла и файловой системы. <p>Уметь:</p> <ul style="list-style-type: none">– использовать полученные знания по операционным системам для создания программных приложений для современных операционных систем; <p>Владеть:</p> <ul style="list-style-type: none">– навыками, необходимыми для эффективной работы в современных операционных системах.

1. МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ОБРАЗОВАТЕЛЬНОЙ ПРОГРАММЫ

Содержание дисциплины основывается и является логическим продолжением следующих дисциплин:

№	Наименование дисциплины (модуля)
1	Технологии программирования (Все разделы)
2	Информатика (Все разделы)
3	Визуальное программирование (Все разделы)

Содержание дисциплины служит основой для изучения следующих

дисциплин:

№	Наименование дисциплины (модуля)
1	Информационная безопасность (Безопасность в операционных системах)
2	Администрирование информационных систем (Обзор современных операционных систем. Файлы и файловые системы. Безопасность в операционных системах)

2. ОБЪЕМ ДИСЦИПЛИНЫ

Общая трудоемкость дисциплины составляет 4 зач. единицы, 144 часов.

Вид учебной работы	Всего часов	Семестр № 5
Общая трудоемкость дисциплины, час	144	144
Контактная работа (аудиторные занятия), в т.ч.:	51	51
лекции	17	17
лабораторные	34	34
практические		
Самостоятельная работа студентов, в том числе:	93	93
Курсовой проект		
Курсовая работа		
Расчетно-графическое задания		
Индивидуальное домашнее задание		
<i>Другие виды самостоятельной работы</i>	53	53
Форма промежуточная аттестация (зачет, экзамен)	40 Экзамен	40 Экзамен

3. СОДЕРЖАНИЕ ДИСЦИПЛИНЫ

4.1 Наименование тем, их содержание и объем

Курс 3 Семестр 5

№ п/п	Наименование раздела (краткое содержание)	Объем на тематический раздел по видам учебной нагрузки, час			
		Лекции	Практические занятия	Лабораторные з анятия	Самостоятельная работа
1. Раздел 1. Основные функции операционных систем					
	Определение операционной системы. Исполнение программ. Управление памятью. Управление устройствами. Управление файлами. Обеспечение пользовательского интерфейса. Обеспечение безопасности	1		12	12
2. Раздел 2. Классификация операционных систем					
	Классификация по типу аппаратного обеспечения. Классификация по числу одновременно выполняемых задач. Классификация по числу одновременно работающих пользователей. Классификация по количеству поддерживаемых процессоров	2			2
3. Раздел 3. Архитектура операционной системы					
	Монолитные операционные системы. Микроядерные операционные системы. Гибридные операционные системы	1			2
4. Раздел 4. Обзор современных операционных систем					
	Linux. Oracle Solaris. Apple OS X и iOS. Microsoft Windows. IBM AIX	1			2
5. Раздел 5. Процессы и потоки					
	Определение процесса и потока. Алгоритмы планирования. Синхронизация	3		8	15
6. Раздел 6. Виртуальная память					
	Страничная организация виртуальной памяти. Сегментная организация виртуальной памяти	3		2	4
7. Раздел 7. Файлы и файловые системы					
	Определение файла и файловой системы. Файловая система FAT. Файловая система NTFS	3		6	8
8. Раздел 8. Безопасность в операционных системах					
	Модель безопасности UNIX. Модель безопасности Windows	3		6	8
	ВСЕГО	17		34	53

4.2. Содержание практических (семинарских) занятий

Не предусмотрено

4.3. Содержание лабораторных занятий

№ п/п	Наименование раздела дисциплины	Тема лабораторного занятия	К-во часов	К-во часов СРС
семестр № 5				
1	Основные функции операционных систем	Введение в Windows API	6	6
2		Оконные приложения Windows	6	6
3	Процессы и потоки	Процессы и потоки в Windows	6	6
4	Файлы и файловые системы	Файлы и каталоги. Системный реестр Windows	6	6
5	Безопасность в операционных системах	Безопасность в Windows	6	6
6	Процессы и потоки. Виртуальная память	Межпроцессное взаимодействие. Службы Windows	4	9
ИТОГО:			34	39
			ВСЕГО:	73

4. ФОНД ОЦЕНОЧНЫХ СРЕДСТВ ДЛЯ ПРОВЕДЕНИЯ ТЕКУЩЕГО КОНТРОЛЯ УСПЕВАЕМОСТИ, ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ ПО ИТОГАМ ОСВОЕНИЯ ДИСЦИПЛИНЫ

5.1. Перечень контрольных вопросов (типовых заданий)

Контрольные вопросы для текущего контроля

- 1) Что такое Windows API?
- 2) Чем Win32 API отличается от Win64 API?
- 3) В каких библиотеках находятся основные функции Win32 API?
- 4) Какие типы данных определены в Win32 API? Что такое дескриптор объекта?
- 5) Каким образом осуществляется обработка ошибок в Win32 API?
- 6) Каким образом осуществляется работа с символами и строками в Win32 API? Что такое безопасные строковые функции?
- 7) Какие в Win32 API имеются дополнительные возможности при работе со строками?
- 8) Какие функции Win32 API следует использовать для того, чтобы получить имя компьютера?
- 9) Какие функции Win32 API следует использовать для того, чтобы получить имя пользователя текущего сеанса?
- 10) Какие функции Win32 API следует использовать для того, чтобы получить пути к различным каталогам Windows?
- 11) Какую функцию Win32 API следует использовать для того, чтобы определить версию Windows?
- 12) Какие функции Win32 API следует использовать для того, чтобы получить значения различных метрик и параметров Windows?
- 13) Какие функции Win32 API следует использовать при работе с системной датой?

и временем?

- 14) Что такое консольное приложение Win32? Как в Visual C++ создать проект консольного приложения Win32?
- 15) Для чего в консольном приложении Win32 предназначена функция main? Какие существуют варианты функции main?
- 16) Что такое точка входа в консольное приложение Win32?
- 17) Что называют статической библиотекой? Как в Visual C++ создать проект статической библиотеки?
- 18) Как в Visual C++ подключить статическую библиотеку к проекту Win32?
- 19) Что называют динамически подключаемой библиотекой (DLL)? Как в Visual C++ создать проект библиотеки DLL?
- 20) Как в Visual C++ подключить библиотеку DLL к проекту Win32? Что такое явное и неявное подключение DLL?
- 21) Что такое точка входа в DLL?
- 22) Что такое отложенная загрузка DLL? Как в Visual C++ применить отложенную загрузку DLL?
- 23) Что в Windows называют окном?
- 24) Что такое окно рабочего стола Windows?
- 25) Что такое главное окно приложения Windows? Сколько таких окон может быть у приложения?
- 26) Что в Windows называют диалоговое окно? Какие различают виды диалоговых окон?
- 27) Что в Windows называют элементами управления?
- 28) Что такое клиентская и не клиентская область окна? Что они в себя включают?
- 29) Какие системы координат используются в функциях Win32 API, работающих с окнами?
- 30) Какие функции Win32 API следует использовать для преобразования из одной системы координат в другую?
- 31) Что такое оконный класс? Как создать новый оконный класс?
- 32) Что такое предопределенные оконные классы элементов управления общего пользования? Какие предопределенные оконные классы существуют?
- 33) Какие функции Win32 API следует использовать для того, чтобы создать окно?
- 34) Какие функции Win32 API следует использовать для создания различных диалоговых окон?
- 35) Какие функции Win32 API возвращают дескриптор окна?
- 36) Какие функции Win32 API изменяют отображение окна?
- 37) Какие функции Win32 API возвращают различные характеристики окна или его оконного класса?
- 38) Что в Windows называют оконным сообщением?
- 39) Что такое очередь сообщений?
- 40) Что такое цикл обработки сообщений? Как обрабатываются сообщения диалоговых окон?
- 41) Что такое оконная процедура? Для чего применяется оконная процедура?
- 42) Что такое оконная процедура диалогового окна? Чем она отличается от обычной оконной процедуры?
- 43) Какие оконные сообщения используются наиболее часто?
- 44) Чем отличаются оконные сообщения WM_CREATE и WM_INITDIALOG?

- 45) Чем отличаются оконные сообщения WM_DESTROY и WM_CLOSE?
- 46) Чем отличаются оконные сообщения WM_COMMAND и WM_NOTIFY?
- 47) В чем разница между синхронными и асинхронными оконными сообщениями?
- 48) Какую функцию Win32 API следует использовать для того, чтобы отправить синхронное оконное сообщение?
- 49) Какую функцию Win32 API следует использовать для того, чтобы отправить асинхронное оконное сообщение?
- 50) Зачем отправлять оконные сообщения элементам управления? Какие функции Win32 API следует для этого использовать?
- 51) Какие существуют макросы для отправки оконных сообщений элементам управления? В чем заключается удобство использования этих макросов?
- 52) Какую функцию Win32 API следует использовать для того, чтобы отправить сообщение WM_QUIT? Почему нельзя использовать другие функции отправки оконных сообщений?
- 53) Как и для чего применяют распаковщики сообщений?
- 54) Что такое приложение Win32? Как в Visual C++ создать проект приложения Win32?
- 55) Для чего в приложении Win32 предназначена функция WinMain? Какие существуют варианты функции WinMain?
- 56) Что такое точка входа в приложение Win32?
- 57) Что такое и для чего применяется манифест приложения?
- 58) Для чего предназначена утилита Spy++?
- 59) Что в Windows называют объектом ядра? Какие объекты ядра позволяют создавать Windows?
- 60) Какие объекты ядра могут находиться в свободном или занятом состоянии?
- 61) Для чего предназначена утилита WinObj?
- 62) Как создаются и удаляются объекты ядра? Что такое счетчик использования объекта ядра?
- 63) Для чего используется функция CloseHandle?
- 64) Для чего применяют наследование дескрипторов объектов ядра? Как создать наследуемый дескриптор объекта ядра?
- 65) Какую функцию Win32 API следует использовать для дублирования дескриптора объекта ядра? Для чего применяют дублирование дескрипторов объектов ядра?
- 66) Что такое именованные объекты ядра? Для чего применяют именованные объекты ядра?
- 67) Что в Windows называют процессом?
- 68) Для чего предназначены утилиты Task Manager и Process Explorer?
- 69) Какую функцию Win32 API следует использовать для того, чтобы создать процесс? Что такое обособленный процесс?
- 70) Каким образом завершается работа процесса? Как определить код завершения процесса? Какие функции Win32 API применяются для принудительного завершения работы процесса?
- 71) Что такое идентификатор процесса? Как определить идентификатор текущего процесса?
- 72) Что такое псевдо дескриптор процесса? Как определить псевдо дескриптор текущего процесса?

- 73) Какие функции Win32 API следует использовать для того, чтобы получить дескриптор процесса, зная его идентификатор или псевдо дескриптор?
- 74) Что такое класс приоритета процесса? Какие классы приоритета существуют в Windows?
- 75) Какие функции Win32 API следует использовать для работы с классом приоритета процесса?
- 76) Какую функцию Win32 API следует использовать для того, чтобы получить список идентификаторов процессов?
- 77) Какие функции Win32 API можно использовать для того, чтобы получить список модулей, загруженных процессом?
- 78) Какие функции Win32 API следует использовать для того, чтобы получить имя модуля, загруженного процессом?
- 79) Что в Windows называют заданием?
- 80) Какую функцию Win32 API следует использовать для того, чтобы создать задание?
- 81) Какие функции Win32 API используются для работы с ограничениями, накладываемыми на процессы в задании?
- 82) Как включить процесс в задание? Какую функцию Win32 API для этого следует использовать?
- 83) Можно ли включить процесс одновременно в несколько заданий?
- 84) Можно ли исключить процесс из задания?
- 85) Всегда ли дочерние процессы принадлежат к тому же заданию, что и родительский процесс?
- 86) Какую функцию Win32 API следует использовать для того, чтобы принудительно завершить работу всех процессов, включенных в одно задание?
- 87) Как получить список идентификаторов процессов, включенных в задание?
- 88) Что в Windows называют потоком? Что такое главный поток?
- 89) Какая функция Win32 API используется для создания потока?
- 90) Каким образом завершается работа потока? Как определить код завершения потока?
- 91) Какие функции Win32 API применяются для принудительного завершения работы потока?
- 92) Для чего предназначена функция RtlUserThreadStart?
- 93) Какие функции Visual C++ следует использовать для создания и завершения потока? Почему для этого не следует использовать функции Win32 API?
- 94) Что такое идентификатор потока? Как определить идентификатор текущего потока?
- 95) Что такое псевдо дескриптор потока? Как определить псевдо дескриптор текущего потока?
- 96) Какие функции Win32 API следует использовать для того, чтобы получить дескриптор потока, зная его идентификатор или псевдо дескриптор?
- 97) Что такое счетчик приостановок потока?
- 98) Какие функции Win32 API применяются для приостановки и возобновления работы потока?
- 99) Что такое относительный приоритет потока? Каким образом вычисляется уровень приоритета потока?
- 100) Какие функции Win32 API следует использовать для работы с относительным

приоритетом потока?

- 101) Для чего применяется синхронизация потоков?
- 102) Что такое функции взаимоблокировки? Как и для чего применяют функции взаимоблокировки?
- 103) Что такое критические секции? Как и для чего применяют критические секции?
- 104) Что такое тонкая блокировка чтения и записи? Как и для чего применяют тонкую блокировку чтения и записи?
- 105) Что такое функции ожидания? Как и для чего применяют функции ожидания?
- 106) Что такое мьютекс? Как и для чего применяют мьютекс?
- 107) Что такое семафор? Как и для чего применяют семафор?
- 108) Что такое событие? Как и для чего применяют событие?
- 109) Какие существуют правила именования файлов и каталогов в Windows?
- 110) Какую функцию Win32 API следует использовать для создания/открытия файла?
- 111) Как открыть файл только для чтения или только для записи? Как открыть файл для чтения и записи?
- 112) Как организовать совместный доступ к открытому файлу? Для чего нужен совместный доступ к файлу?
- 113) Как создать скрытый файл, временный файл и файл доступный только для чтения?
- 114) Какую функцию Win32 API следует использовать для закрытия файла?
- 115) Как создать файл, который будет удален после закрытия?
- 116) Какие функции Win32 API следует использовать для чтения и записи данных в файлах?
- 117) Что такое буферизация данных при чтении и записи данных? Как отключить буферизацию данных?
- 118) Для чего предназначено упреждающее чтение данных? Как отключить упреждающее чтение данных?
- 119) Что такое асинхронные операции чтения и записи данных? Как разрешить асинхронные операции чтения и записи данных?
- 120) Какие функции Win32 API следует использовать для асинхронных операций чтения и записи данных?
- 121) Какую функцию Win32 API следует использовать для отмены асинхронных операций чтения и записи данных?
- 122) Какую функцию Win32 API следует использовать для изменения указателя в файле?
- 123) Какую функцию Win32 API следует использовать для принудительного указания конца файла?
- 124) Какие функции Win32 API следует использовать для создания каталогов?
- 125) Какие функции Win32 API следует использовать для удаления файлов и каталогов?
- 126) Какие функции Win32 API следует использовать для перемещения или переименования файлов и каталогов?
- 127) Какую функцию Win32 API следует использовать для копирования файлов?
- 128) Какую функцию Win32 API следует использовать для открытия каталогов и логических дисков?

- 129) Что такое текущий каталог? Какие функции Win32 API следует использовать для определения и изменения текущего каталога?
- 130) Какими атрибутами обладают файлы и каталоги? Какие функции Win32 API следует использовать для определения и изменения атрибутов файлов и каталогов?
- 131) Какие функции Win32 API следует использовать для поиска файлов и каталогов? Что такое шаблон поиска?
- 132) Что такое файл инициализации? Какие функции Win32 API следует использовать для работы с файлами инициализации?
- 133) Что такое системный реестр Windows?
- 134) Какие функции Win32 API следует использовать для создания/открытия ключей реестра?
- 135) Какую функцию Win32 API следует использовать для закрытия ключей реестра?
- 136) Какие функции Win32 API следует использовать для удаления ключей реестра?
- 137) Какие функции Win32 API следует использовать для перечисления вложенных ключей и параметров в ключе реестра?
- 138) Какие функции Win32 API следует использовать для работы с параметрами в ключах реестра?
- 139) Какие объекты Windows относятся к защищаемым объектам?
- 140) Что в модели безопасности Windows называют доверенным объектом?
- 141) Что такое идентификатор безопасности?
- 142) Какие существуют общеизвестные идентификатор безопасности?
- 143) Какие функции Win32 API следует использовать для получения идентификатора безопасности учетной записи пользователя, группы или компьютера?
- 144) Что такое дескриптор безопасности? Из каких компонентов состоит дескриптор безопасности?
- 145) Что такое владелец и основная группа объекта?
- 146) Что такое списки контроля доступа? Какие бывают виды списков контроля доступа?
- 147) Что такое элемент контроля доступа? Какие различают типы элементов контроля доступа?
- 148) В каком порядке должны располагаться элементы контроля доступа?
- 149) Что такое абсолютный и относительный формат дескриптора безопасности? В чем заключаются различия между ними?
- 150) Какие существуют управляющие флаги дескриптора безопасности?
- 151) Какие функции Win32 API следует использовать для работы с управляющими флагами дескриптора безопасности?
- 152) Какие функции Win32 API следует использовать для работы с владельцем и основной группой в дескрипторе безопасности?
- 153) Какие функции Win32 API следует использовать для работы со списками контроля доступа в дескрипторе безопасности?
- 154) Какие функции Win32 API следует использовать для работы с дескриптором безопасности?
- 155) Что такое контекст защиты?
- 156) Что такое маркер доступа?

- 157) Что такое привилегии безопасности?
- 158) В каких состояниях могут находиться привилегии безопасности?
- 159) Какие функции Win32 API следует использовать для работы привилегиями безопасности в маркере доступа?
- 160) Какую функцию Win32 API можно использовать для получения маркера доступа?
- 161) Что называют межпроцессным взаимодействием?
- 162) Какие механизмы межпроцессного взаимодействия поддерживаются в Windows?
- 163) Что такое буфер обмена Windows?
- 164) Какие функции Win32 API следует использовать для того, чтобы открыть или закрыть буфер обмена? Для чего необходимо открывать и закрывать буфер обмена?
- 165) Какие функции Win32 API следует использовать для передачи данных через буфер обмена?
- 166) Какие форматы текстовых данных поддерживаются в буфере обмена? Как создать собственный формат данных в буфере обмена?
- 167) Что в Windows называют спроецируемым в память файлом?
- 168) Как спроецировать файл на адресное пространство процесса? Какие функции Win32 API следует для этого использовать?
- 169) Как удалить спроецированный файл из адресного пространства процесса? Какие функции Win32 API следует для этого использовать?
- 170) Как использовать спроецированные в память файлы для передачи данных между процессами?
- 171) Какие оконные сообщения можно использовать для передачи данных между процессами? Какие функции Win32 API следует для этого использовать?
- 172) Каким образом осуществляется отправка оконных сообщений окнам, созданным в другом процессе?
- 173) Как осуществляется передача текстовых данных с помощью оконных сообщений WM_SETTEXT и WM_GETTEXT? Чем отличаются эти оконные сообщения?
- 174) Как осуществляется передача данных с помощью оконного сообщения WM_COPYDATA?
- 175) Что в Windows называют почтовым ящиком?
- 176) Какие функции Win32 API следует использовать для создания и открытия почтового ящика? Какие существуют правила именования почтовых ящиков?
- 177) Какие функции Win32 API следует использовать для передачи данных через почтовый ящик?
- 178) Как организовать двунаправленный обмен данными между двумя процессами при помощи почтовых ящиков?
- 179) Что в Windows называют анонимным каналом?
- 180) Какую функцию Win32 API следует использовать для создания анонимного канала?
- 181) Какие функции Win32 API следует использовать для передачи данных через анонимный канал?
- 182) Что в Windows называют именованным каналом?
- 183) Какие функции Win32 API следует использовать для создания и открытия

- именованного канала? Какие существуют правила именования каналов?
- 184) Какие функции Win32 API следует использовать для передачи данных через именованный канал?
- 185) Чем отличаются механизмы передачи данных через анонимный канал и через именованный канал?
- 186) Что такое службы Windows?
- 187) Для чего предназначен диспетчер управления службами?
- 188) Какие функции Win32 API следует использовать для создания, открытия и удаления служб Windows?
- 189) Какие функции Win32 API следует использовать для управления службами Windows?
- 190) Как в Visual C++ создать проект приложения, реализующего службу Windows?
- 191) Что такое точка входа в службу Windows?
- 192) Как осуществляется обработка уведомлений, поступающих в службу Windows?
- 193) Как и для чего применяется изоляция служб в Windows?

Экзаменационные вопросы

№ п/п	Наименование раздела дисциплины	Содержание вопросов (типовых заданий)
1	Основные функции операционных систем	Определение операционной системы.
2		Определение операционной системы. Основные функции операционных систем. Исполнение программ.
3		Основные функции операционных систем. Управление памятью.
4		Основные функции операционных систем. Управление устройствами ввода-вывода.
5		Основные функции операционных систем. Управление файлами.
6		Основные функции операционных систем. Обеспечение пользовательского интерфейса.
7		Основные функции операционных систем. Обеспечение безопасности.
8	Классификация операционных систем	Классификация операционных систем. По типу аппаратного обеспечения.
9		Классификация операционных систем. По числу одновременно выполняемых задач.
10		Классификация операционных систем. По числу одновременно работающих пользователей.
11		Классификация операционных систем. По количеству поддерживаемых процессоров.
12	Архитектура операционной системы	Архитектура операционной системы. Монолитные операционные системы.
13		Архитектура операционной системы. Микроядерные операционные системы.
14		Архитектура операционной системы. Гибридные операционные системы.
15	Современные операционные системы	Современные операционные системы. Linux.
16		Современные операционные системы. Oracle Solaris, Apple OS X, Apple iOS и IBM AIX.
17		Современные операционные системы. Microsoft Windows.

18	Процессы и потоки	Определение и реализация процесса. Создание и завершение процесса.
19		Определение и реализация процесса. Состояния процесса.
20		Определение потока. Реализация потоков на уровне ядра.
21		Определение потока. Реализация потоков на уровне пользователя.
22		Определение потока. Смешанная реализация потоков.
23		Определение потока. Состояния потока.
24		Планирование процессов и потоков. Алгоритм FIFO.
25		Планирование процессов и потоков. Алгоритм наименьшего время выполнения.
26		Планирование процессов и потоков. Алгоритм циклического планирования.
27		Планирование процессов и потоков. Алгоритм приоритетного планирования.
28		Планирование процессов и потоков. Алгоритм справедливого планирования.
29		Синхронизация процессов и потоков. Критические секции и взаимное исключение. Запрещение прерываний.
30		Синхронизация процессов и потоков. Критические секции и взаимное исключение. Переменные блокировки.
31		Синхронизация процессов и потоков. Критические секции и взаимное исключение. Поочередный доступ к критической секции.
32		Синхронизация процессов и потоков. Критические секции и взаимное исключение. Алгоритм Петерсона.
33		Синхронизация процессов и потоков. Критические секции и взаимное исключение. Блокировка шины памяти.
34		Синхронизация процессов и потоков. Приостановка и активизация. Задача производителя и потребителя.
35		Синхронизация процессов и потоков. Приостановка и активизация. Семафоры.
36		Синхронизация процессов и потоков. Приостановка и активизация. Мьютексы.
37		Синхронизация процессов и потоков. Задача обедающих философов.
38		Синхронизация процессов и потоков. Задача читателей и писателей.
39		Синхронизация процессов и потоков. Задача спящего бравуря.
40	Виртуальная память	Виртуальная память. Страничная организация виртуальной памяти. Таблица страниц.
41		Виртуальная память. Страничная организация виртуальной памяти. Многоуровневые таблицы страниц.
42		Виртуальная память. Страничная организация виртуальной памяти. Буфер ассоциативной трансляции
43		Виртуальная память. Алгоритмы замещения страниц. Алгоритм NPU.
44		Виртуальная память. Алгоритмы замещения страниц. Алгоритм FIFO.
45		Виртуальная память. Алгоритмы замещения страниц. Алгоритм «вторая попытка».
46		Виртуальная память. Алгоритмы замещения страниц. Алгоритм «часы».

47	Виртуальная память	Виртуальная память. Сегментная и сегментно-страничная организация виртуальной памяти.
48	Файлы и файловые системы	Файлы и файловые системы. Файловая система FAT.
49		Файлы и файловые системы. Файловая система NTFS.
50	Основные функции операционных систем	Определение интерфейса прикладного программирования Windows. Win32 API.
51		Типы данных Win32 API.
52		Обработка ошибок в Win32 API.
53		Работа с символами и строками в Windows.
54		Работа с системной информацией Windows.
55		Консольные приложения Win32.
56		Статические и динамические библиотеки.
57		Окна и элементы управления в Windows. Оконный класс.
58		Окна и элементы управления в Windows. Оконные сообщения.
59		Приложения Win32.
60	Процессы и потоки	Определение объекта ядра Windows. Создание и закрытие объектов ядра.
61		Определение объекта ядра Windows. Наследование дескриптора объекта ядра.
62		Определение объекта ядра Windows. Дублирование дескриптора объекта ядра.
63		Определение объекта ядра Windows. Именованные объекты ядра.
64		Процессы в Windows.
65		Задания в Windows.
66		Потоки в Windows.
67		Приоритет процессов и потоков в Windows.
68		Синхронизация потоков в Windows. Синхронизация в пользовательском режиме.
69		Синхронизация потоков в Windows. Синхронизация с использованием объектов ядра.
70	Файлы и файловые системы	Правила именования файлов и каталогов в Windows.
71		Работа с файлами и каталогами в Windows.
72		Асинхронные операции ввода/вывода.
73		Файлы инициализации.
74		Системный реестр Windows.
75	Безопасность в операционных системах	Модель безопасности Windows. Учетные записи.
76		Модель безопасности Windows. Идентификаторы безопасности.
77		Модель безопасности Windows. Права и привилегии учетной записи.
78		Модель безопасности Windows. Дескриптор безопасности. Владелец и основная группа.
79		Модель безопасности Windows. Дескриптор безопасности. Списки контроля доступа.
80		Модель безопасности Windows. Маркер доступа.
81	Основные функции операционных систем. Процессы и потоки	Межпроцессное взаимодействие. Буфер обмена.
82		Межпроцессное взаимодействие. Проецируемые в память файлы.
83		Межпроцессное взаимодействие. Оконные сообщения.
84		Межпроцессное взаимодействие. Почтовые ящики.

5.2. Перечень тем курсовых проектов, курсовых работ, их краткое содержание и объем.

Не предусмотрено

5.3. Перечень индивидуальных домашних заданий, расчетно-графических заданий.

Не предусмотрено

4.4. Перечень контрольных работ.

Не предусмотрено

6. ОСНОВНАЯ И ДОПОЛНИТЕЛЬНАЯ ЛИТЕРАТУРА

6.1. Перечень основной литературы

1. Трухачев С.С. Операционные системы: лабораторный практикум Лабор. практ. БГТУ им. В.Г. Шухова 2012 <https://elib.bstu.ru/Reader/Book/-2271>
2. Олифер В. Г., Олифер Н. А. Сетевые операционные системы, 2-е изд. Учебник СПб.: Питер 2009
3. Гордеев А. В. Операционные системы, 2-е изд. Учебник СПб.: Питер 2007

6.2. Перечень дополнительной литературы

1. Партыка Т. Л., Попов И. И. Операционные системы, среды и оболочки Учебное пособие М.: Инфра-М 2006
2. Бэкон Д., Харрис Т. Операционные системы. Параллельные и распределенные системы Учебное пособие СПб. : Питер 2004
3. Гордеев А. В. Операционные системы, 2-е изд. Учебник СПб.: ПИТЕР 2004

6.3. Интернет-ресурсы

1. Access Control Model [Электронный ресурс] // Microsoft Developer Network. - Режим доступа: <http://msdn.microsoft.com/en-us/library/windows/desktop/aa374876/>.
2. Common Dialog Box Library [Электронный ресурс] // Microsoft Developer Network. - Режим доступа: <http://msdn.microsoft.com/en-us/library/windows/desktop/ms645524/>.
3. Control Library [Электронный ресурс] // Microsoft Developer Network. - Режим доступа: <http://msdn.microsoft.com/en-us/library/windows/desktop/bb773169/>.

4. Controls [Электронный ресурс] // Microsoft Developer Network. - Режим доступа: <http://msdn.microsoft.com/en-us/library/win-dows/desktop/aa511482/>.
5. Keyboard Accelerators [Электронный ресурс] // Microsoft Developer Network. - Режим доступа: <http://msdn.microsoft.com/en-us/library/windows/desktop/ms645526/>.
6. Keyboard and Mouse Input [Электронный ресурс] // Microsoft Developer Network. - Режим доступа: <http://msdn.microsoft.com/en-us/library/windows/desktop/ms632585/>.
7. Security and Identity [Электронный ресурс] // Microsoft Developer Network. - Режим доступа: <http://msdn.microsoft.com/en-us/library/windows/desktop/ee663293/>.
8. Services [Электронный ресурс] // Microsoft Developer Network. - Режим доступа: <http://msdn.microsoft.com/en-us/library/win-dows/desktop/ms685141/>.
9. Synchronization [Электронный ресурс] // Microsoft Developer Network. - Режим доступа: <http://msdn.microsoft.com/en-us/library/windows/desktop/ms686353/>.
10. Windows and Messages [Электронный ресурс] // Microsoft Developer Network. - Режим доступа: <http://msdn.microsoft.com/en-us/library/windows/desktop/ms632586/>.

7. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ

Учебные аудитории для проведения лекционных занятий, лабораторных занятий, групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации, а также помещения для самостоятельной работы обучающихся, оснащенные компьютерной техникой с возможностью подключения к сети «Интернет» и обеспечением доступа в электронную информационно-образовательную среду организации. Аудитории для лекционных занятий оборудованы специализированной мебелью, мобильным или стационарным мультимедийным проектором, переносным экраном, ноутбуком, или компьютерами на базе одно или двухъядерных процессоров с тактовой частотой не менее 2 ГГц, объемом оперативной памяти не менее 2 Гб и жесткого диска до 500 Гб; локальная сеть с пропускной способностью 100 Мбит/с; лазерные принтеры или многофункциональные устройства форматов А4, А3; планшетные сканеры (при отсутствии МФУ).

Для проведения лабораторных занятий могут использоваться компьютерные классы, оснащенные компьютерами с установленными программными продуктами:

Лицензионное ПО:

- Microsoft Office Professional
- Microsoft Windows
- Microsoft Visual Studio
- Система компьютерного тестирования знаний VeralTest

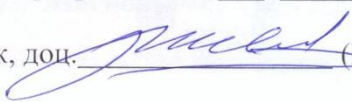
8. УТВЕРЖДЕНИЕ РАБОЧЕЙ ПРОГРАММЫ


Утверждение рабочей программы с изменениями, дополнениями

1. На титульном листе рабочей программы читать название «Федеральное государственное бюджетное образовательное учреждение высшего профессионального образования» как «Федеральное государственное бюджетное образовательное учреждение высшего образования»
2. Институт информационных технологий и управляющих систем был переименован 30.04.2016 г. в институт Энергетики, информационных технологий и управляющих систем на основании приказа № 4/52 от 29.02.2016 г.

Рабочая программа с изменениями, дополнениями утверждена на 2016/2017 учебный год.

Протокол № 7 заседания кафедры ИТ от «15» 06 2016 г.

Заведующий кафедрой: канд.техн. наук, доц.  (И.В. Иванов)

Директор института ЭИТУС: канд.техн. наук, доц.  (А.В. Белоусов)

Утверждение рабочей программы без изменений

Рабочая программа без изменений и дополнений утверждена на 20¹⁷/20¹⁸ учебный год.

Протокол № 12 заседания кафедры ИТ от «27» 06 20¹⁷ г.

Заведующий кафедрой: канд.техн. наук, доц. [signature] (И.В. Иванов)

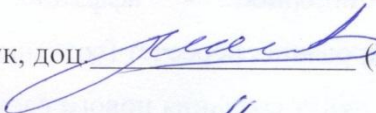
Директор института ЭИТУС: канд.техн. наук, доц. [signature] (А.В. Белоусов)


Утверждение рабочей программы с изменениями, дополнениями

1. Изменения в п. 6

Рабочая программа с изменениями, дополнениями утверждена на 20¹⁸/20¹⁹ учебный год.

Протокол № 6 заседания кафедры ИТ от «11» 04 20¹⁸ г.

Заведующий кафедрой: канд.техн. наук, доц.  (И.В. Иванов)

Директор института ЭИТУС: канд.техн. наук, доц.  (А.В. Белоусов)

6. ОСНОВНАЯ И ДОПОЛНИТЕЛЬНАЯ ЛИТЕРАТУРА

6.1. Перечень основной литературы

1. Глухоедов А.В. Операционные системы: лабораторный практикум: учебное пособие / А.В. Глухоедов. – Белгород: Изд-во БГТУ, 2017. – 386 с. ISBN 978-5-361-00393-8.
2. Олифер, В. Г. Сетевые операционные системы : учебник / В. Г. Олифер, Н. А. Олифер. - 2-е изд. - Санкт-Петербург : Питер, 2009. - 668 с.
3. Гордеев, А. В. Операционные системы : учебник / А. В. Гордеев. - 2-е изд. - Санкт-Петербург : Питер, 2007. - 415 с.

6.2. Перечень дополнительной литературы

1. Назаров, С. В. Современные операционные системы [Электронный ресурс] : учебное пособие / Назаров С. В. - Москва : Интернет-Университет Информационных Технологий (ИНТУИТ), 2016. - 279 с. <http://www.iprbookshop.ru/52176.html?replacement=1>
2. Журавлева, Т. Ю. Практикум по дисциплине «Операционные системы» [Текст] : автоматизированный практикум / Журавлева Т. Ю. - Саратов : Вузовское образование, 2014. - 40 с. <http://www.iprbookshop.ru/20692>
3. Основы операционных систем. Лекция 1. Презентация [Текст] . - Москва : Национальный Открытый Университет «ИНТУИТ», 2014. <http://biblioclub.ru/index.php?page=book&id=237121>
4. Партыка, Т. Л. Операционные системы, среды и оболочки : учеб. пособие / Т. Л. Партыка, И. И. Попов. - Москва : Инфра-М, 2006. - 399 с.
5. Бэкон, Д. Операционные системы. Параллельные и распределенные системы / Д. Бэкон, Т. Харрис. - Санкт-Петербург : Питер, 2004. - 799 с.
6. Гордеев, А. В. Операционные системы : учебник / А. В. Гордеев. - 2-е изд. - Санкт-Петербург : ПИТЕР, 2004. - 415 с.

Утверждение рабочей программы без изменений

Рабочая программа без изменений утверждена на 2019 /2020 учебный год.

Протокол № 9 заседания кафедры ИТ от «7» июня 2019 г.


И.о.зав. кафедрой ИТ: канд.техн. наук  (Д.Н. Старченко)

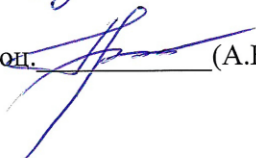
Директор института ЭИТУС: канд.техн. наук, доц.  (А.В. Белоусов)

Утверждение рабочей программы без изменений

Рабочая программа без изменений утверждена на 2020 /2021 учебный год.

Протокол № 6 заседания кафедры ИТ от «12» 05 2020 г.


И.о.зав. кафедрой ИТ: канд.техн. наук  (Д.Н. Старченко)

Директор института ЭИТУС: канд.техн. наук, доц.  (А.В. Белоусов)

Утверждение рабочей программы без изменений

Рабочая программа без изменений утверждена на 2021 /2022 учебный год.

Протокол № 6 заседания кафедры ИТ от «30» 04 2021 г.

И.о.зав. кафедрой ИТ канд.техн.наук  (Д.Н. Старченко)

Директор института ЭИТУС канд.техн.наук, доц.  (А.В. Белоусов)