

**Федеральное государственное автономное образовательное учреждение  
высшего образования  
"Национальный исследовательский университет  
"Высшая школа экономики"**

Факультет компьютерных наук  
Департамент программной инженерии

**Рабочая программа дисциплины  
Конструирование операционных систем**

для образовательной программы «Программная инженерия»  
направления подготовки 09.03.04 «Программная инженерия»  
уровень - бакалавр

Разработчик программы  
Хорошилов А.В, к.ф.-м.н., khoroshilov@ispras.ru

Одобрена на заседании департамента программной инженерии «\_\_»\_\_\_\_\_ 2016 г.  
Руководитель департамента Авдошин С.М. \_\_\_\_\_

Утверждена Академическим советом образовательной программы «\_\_»\_\_\_\_\_ 2016 г.,  
№ протокола \_\_\_\_\_

Академический руководитель образовательной программы Шилов В.В. \_\_\_\_\_

Москва, 2016

*Настоящая программа не может быть использована другими подразделениями университета и другими вузами без разрешения подразделения-разработчика программы.*

## 1. Область применения и нормативные ссылки

Настоящая программа учебной дисциплины устанавливает минимальные требования к знаниям и умениям студента и определяет содержание и виды учебных занятий и отчетности. Программа предназначена для преподавателей, ведущих данную дисциплину, учебных ассистентов и студентов образовательной программы «Программная инженерия» направления подготовки 09.03.04 «Программная инженерия», изучающих дисциплину "Конструирование операционных систем". Программа разработана в соответствии с образовательным стандартом Национального исследовательского университета «Высшая школа экономики» по направлению 09.03.04 «Программная инженерия»

## 2. Цели освоения дисциплины

*Цель курса* – Целью освоения дисциплины является изучение основных принципов внутреннего устройства ядра операционной системы, механизмов аппаратной поддержки работы ядра, а также получение навыков проектирования и программирования компонентов ядра операционной системы и отладки программ в привилегированном режиме работы процессора.

*Задачами данного курса являются:*

- освоение студентами базовых современных достижений в области проектирования операционных систем;
- формирование практических навыков проектирования и программирования компонентов ядра операционной системы и отладки программ в привилегированном режиме работы процессора.

## 3. Компетенции обучающегося, формируемые в результате освоения дисциплины

В результате освоения дисциплины студент должен:

### 1. Знать:

- фундаментальные понятия, теории современного системного программирования;
- принципы внутреннего устройства ядра операционной системы;
- механизмы аппаратной поддержки работы ядра;
- механизмы обеспечения защиты ядра операционной системы от приложений и приложений друг от друга;
- методы управления и распределения аппаратными ресурсам;
- методы и средства виртуализации аппаратных ресурсов.

### 2. Уметь:

- проектировать компоненты ядра операционной системы;
- программировать на языке Си и на языке ассемблера с использованием привилегированных инструкций процессора;
- отлаживать программы, работающие в привилегированном режиме работы процессора.

### 3. Иметь навыки (приобрести опыт):

- освоения большого объема информации;
- самостоятельной работы с документацией по архитектуре и машинным инструкциям процессора;
- культурой разработки и реализации системного программного обеспечения современных компьютеров;
- разработки и отладки программ, работающих в привилегированном режиме работы процессора.

**В результате освоения дисциплины студент должен обладать следующими компетенциями:**

Универсальные компетенции:

- Способен решать проблемы в профессиональной деятельности на основе анализа и синтеза (УК-3)

- Способен оценивать потребность в ресурсах и планировать их использование при решении задач в профессиональной деятельности (УК-4)
- Способен работать с информацией: находить, оценивать и использовать информацию из различных источников, необходимую для решения научных и профессиональных задач (в том числе на основе системного подхода) (УК-5)
- Способен вести исследовательскую деятельность, включая анализ проблем, постановку целей и задач, выделение объекта и предмета исследования, выбор способа и методов исследования, а также оценку его качества (УК-6)

Профессиональные компетенции:

- Способен использовать методы и инструментальные средства исследования объектов профессиональной деятельности (ПК-3)
- Способен обосновать принимаемые проектные решения, осуществлять постановку и выполнение экспериментов по проверке их корректности и эффективности (ПК-4)
- Способен проектировать, конструировать и тестировать программные продукты (ПК-10)
- Способен читать, понимать и выделять главную идею прочитанного исходного кода, документации (ПК-11)
- Способен использовать различные технологии разработки программного обеспечения (ПК-16)
- Способен применять основные методы и инструменты разработки программного обеспечения (ПК-17)

#### 4. Место дисциплины в структуре образовательной программы

Изучение данной дисциплины базируется на знаниях, полученных студентами при освоении учебных дисциплин:

- «Дискретная математика»,
- «Программирование»,
- «Алгоритмы и структуры данных»,
- «Архитектура вычислительных систем»,
- «Операционные системы».

#### 5. Тематический план учебной дисциплины

№	Название раздела	Всего часов	Аудиторные часы			Самостоятельная работа
			Лекции	Семинары	Практические занятия	
1.	Введение. Карта физической памяти x86. Процесс загрузки и инициализации PC. BIOS, инициализация основных устройств. Загрузчик JOS. Загрузка ядра.	13	1	2		10
2.	Устройство ядра JOS. Отладка кода ядра JOS. Компиляция первой собственной функции, вывод строк на консоль.	13	1	2		10
3.	Описатели процессов в JOS. Создание процессов в JOS, загрузка приложений в память из бинарных секций образа ядра. Переключение контекстов. Кооперативное разделение времени. Примитивный планировщик FIFO без приоритетов.	26	2	4		20

4.	Прерывания в x86. Инициализация IDT. Обработка прерываний таймера. Вытесняющее разделение времени. Прimitивный планировщик Round Robin без приоритетов.	46	2	4	40
5.	Обработка вложенных прерываний в x86. Средства синхронизации, состояние гонок, дeадлоки. Запрет прерываний, семафоры.	34	2	2	30
6.	Управление распределением физических страниц. Виртуальная память. Сегментная и страничная трансляция. Таблицы трансляции.	48	4	4	40
7.	Переключение между режимами работы процессора. Прерывания и системные вызовы. Вложенные прерывания. Изменения в создании процессов, переключении между контекстами. Передача данных между программой и ядром.	60	4	6	50
8.	Управление процессами. Системный вызов fork(). Механизмы межпроцессного взаимодействия.	50	4	6	40
9.	Примитивная файловая система. Реализация системных вызовов open(), close(), read(), write(), exec().	52	4	6	42
<b>Итого:</b>		<b>342</b>	<b>24</b>	<b>36</b>	<b>282</b>

## 6. Формы контроля знаний студентов

Тип контроля	Форма контроля	4 год		
		1 модуль	2 модуль	3 модуль
Текущий	Домашнее задание	*	*	*
	Контрольная работа			*
Итоговый	Экзамен			*

### Критерии оценки знаний, навыков

В рамках курса слушателям предлагается выполнить 12 домашних работ. За сдачу лабораторных работ набираются баллы (максимум 85 баллов). Оценки за контрольную работу и экзамен выставляются по 10-ти балльной шкале.

### Порядок формирования оценок по дисциплине

Оценка по курсу состоит из оценки за выполнение домашних работ  $O_{лаб}$  (85 баллов), контрольной работы  $O_{конт}$  (10 баллов) и оценки за итоговый устный экзамен (10 баллов). В диплом выставляется результирующая оценка по учебной дисциплине, которая формируется по следующей формуле:

$$O_{результ} = 0,1 * O_{лаб} + 0,05 * O_{конт} + 0,1 * O_{экс}$$

## 7. Содержание дисциплины

Тема 1. Введение. Карта физической памяти x86. Процесс загрузки и инициализации персональных ЭВМ архитектуры x86. Задачи, решаемые BIOS, инициализация основных устройств. Функции загрузчика. Загрузка ядра. Эмулятор ЭВМ Qemu.

Тема 2. Виды архитектур ядра ОС. Монолитные и микроядерные архитектуры. Устройство ядра JOS.

Тема 3. Описатели процессов в JOS. Создание процессов в JOS, загрузка приложений в память из бинарных секций образа ядра. Переключение контекстов. Кооперативное разделение времени. Прimitивный планировщик FIFO без приоритетов.

Тема 4. Прерывания в x86. Инициализация IDT. Обработка прерываний таймера. Вытесняющее разделение времени. Прimitивный планировщик Round Robin без приоритетов. Обзор алгоритмов планирования процессорного времени.

Тема 5. Обработка вложенных прерываний в x86. Средства синхронизации, состояние гонок, дедлоки. Запрет прерываний, спинлоки, мьютексы, семафоры. Read-Copy-Update.

Тема 6. Управление распределением физических страниц. Виртуальная память. Сегментная и страничная трансляция x86. Таблицы трансляции.

Тема 7. Переключение между режимами работы процессора. Прерывания и системные вызовы. AMD syscall/sysreturn и Intel sysenter/sysexit. Выполнение системных вызовов без переключения в привилегированный режим (VDSO - Virtual Dynamically linked Shared Objects).

Тема 8. Управление процессами. Системные вызовы fork() и exec(). Механизмы межпроцессного взаимодействия. Обзор механизмов межпроцессного взаимодействия в ОС Linux (сигналы, разделяемая память, семафоры, очереди сообщений, программные каналы, сетевые интерфейсы, взаимодействие на основе файловых систем, KDBUS).

Тема 9. Файловые системы. Основные задачи файловых систем. Организации работы файловых систем в ОС Linux: VFS, драйвера файловых систем, bio, кэширование. Обзор современных файловых систем ОС Linux (ext4, btrfs, XFS, jffs2, f2fs). Прimitивная файловая система JOS. Реализация системных вызовов open(), close(), read(), write(), exec().

## 8. Оценочные средства для текущего контроля и аттестации студента

### 8.1 Тематика заданий текущего контроля

1. Процесс загрузки и инициализации персональных ЭВМ архитектуры x86. Функции загрузчика. Загрузка ядра JOS.
2. Переключение контекстов. Кооперативное разделение времени.
3. Прерывания в x86. Инициализация таблицы дескрипторов прерываний IDT.
4. Обработка прерываний таймера. Вытесняющее разделение времени. Алгоритмы планирования процессорного времени.
5. Обработка вложенных прерываний в x86. Средства синхронизации, состояние гонок, взаимные блокировки. Запрет прерываний, спинлоки, мьютексы, семафоры. Read-Copy-Update.
6. Виртуальная память. Модели организации оперативной памяти. Сегментная и страничная трансляция x86. Таблицы трансляции.
7. Переключение между режимами работы процессора. Прерывания и системные вызовы. Выполнение системных вызовов без переключения в привилегированный режим.
8. Аппаратно-программные средства поддержки мультипрограммного режима – система прерываний, защита памяти, привилегированный режим.
9. Файловые системы. Основные задачи файловых систем.

### 8.2 Примеры контрольных вопросов для экзамена

1. Как обеспечивается защита памяти приложений друг от друга? других ресурсов? Как обеспечивается взаимодействие приложений друг с другом?

2. Как обеспечивается защита памяти ОС от приложений? других ресурсов? Как обеспечивается взаимодействие приложений с ОС?
3. Как возможно выполнить системный вызов без переключения в привилегированный режим?
4. Какие средства синхронизации применяются в операционных системах?

## **9. Учебно-методическое и информационное обеспечение дисциплины**

### **9.1 Базовый учебник**

1. Д.В. Ефремов, Н.Ю. Комаров, А.В. Хорошилов. "Конструирование ядра операционной системы", Издательский отдел факультета ВМК МГУ имени М.В. Ломоносова, Москва, 2015. ISBN: 978-5-89407-549-5

### **9.2 Основная литература**

1. Э. Таненбаум "Архитектура компьютера", Питер, 2012 г.
2. Э. Таненбаум "Современные операционные системы", Питер, 2011 г.
3. Э. Таненбаум "Операционные системы. Разработка и реализация", Питер, 2006 г.
4. А. Робачевский, С. Немнюгин, О. Стесик «Операционная система UNIX», БХВ Санкт-Петербург, 2010 г.

### **9.3 Дополнительная литература**

1. C. A. R. Hoare, Monitors: An Operating System Structuring Concept. Communications of the ACM, 17(10):549--557, 1974.
2. Thomas W. Doeppner "Operating Systems In Depth: Design and Programming" Wiley, 2010
3. Jonathan Corbet, Alessandro Rubini, Greg Kroah-Hartman. "Linux Device Drivers".
4. Ellen Siever, Stephen Figgins, Robert Love, Arnold Robbins. "Linux in a Nutshell". O'Reilly Media, 2009.
5. Скотт Шакон. «Про Git». Apress. Перевод на русский. (<http://git-scm.com/book/ru>).
6. Веб-страница дисциплины: <http://forge.ispras.ru/projects/oscourse-hse-YEAR/wiki>
7. IA-32 Intel Architecture Software Developer's Manual Volume 1: Basic Architecture. Basic 80x86 architecture and programming environment. <http://developer.intel.com>.
8. IA-32 Intel Architecture Software Developer's Manual Volume 3A: System Programming Guide. Operating system support, including segmentation, paging, tasks, interrupt and exception handling. <http://developer.intel.com>.
9. Tool Interface Standard Executable and Linking Format (ELF) Specification Version 1.2.