

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

Факультет компьютерных наук  
Департамент программной инженерии

**ПРОГРАММА УЧЕБНОЙ ДИСЦИПЛИНЫ**  
Научно-исследовательский семинар "Системное программирование"

для образовательной программы «Системное программирование»  
направления подготовки 09.04.04 «Программная инженерия»  
уровень - магистр

Разработчики программы

Аветисян А.И., член-корр. РАН, профессор, д.ф.-м.н., arut@ispras.ru

Петренко А.К., профессор, д.ф.-м.н., petrenko@ispras.ru

Одобрена на заседании департамента программной инженерии «\_\_»\_\_\_\_\_ 2017 г.

Руководитель департамента Авдошин С.М. \_\_\_\_\_

Утверждена Академическим советом образовательной программы «\_\_»\_\_\_\_\_ 2017 г.,

№ протокола \_\_\_\_\_

Академический руководитель образовательной программы Петренко А.К. \_\_\_\_\_

Москва, 2017

*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 «Системное программирование» направления подготовки 09.04.04 «Программная инженерия», изучающих дисциплину "Системное программирование".

Программа разработана в соответствии с образовательным стандартом Национального исследовательского университета «Высшая школа экономики» по направлению 09.04.04 «Программная инженерия».

## **2 Цели освоения дисциплины**

Целями освоения дисциплины "Системное программирование" являются:

- обеспечить студентов базовыми знаниями в области системного программирования;
- заложить основы для последующих курсов, посвященных созданию современных средств системного программирования;
- обучить студентов применению современных интегрированных инструментальных средств, предназначенных для разработки системного программного обеспечения (ПО);
- привить студентам навыки исследовательской работы, предполагающей самостоятельное изучение специфических инструментов и средств, необходимых для решения именно той конкретной проблемы, которая в качестве задачи поставлена перед ними..

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

В результате освоения дисциплины студент должен:

- Знать:
  - основные языки системного программирования;
  - основные принципы архитектурного дизайна системного ПО;
  - основные программные платформы.
- Уметь:
  - разрабатывать системные программы с помощью инструментальных средств;
  - отлаживать и тестировать создаваемые программы, используя диагностические возможности среды разработки;
  - самостоятельно находить новые знания и решения, необходимые для реализации функциональных требований, сформулированных в техническом задании на программу.
- Иметь навыки (приобрести опыт):
  - в решении типовых задач системного программирования с применением современных языков программирования и передовых инструментальных средств;
  - проектирования и программирования системных программ;
  - в применении библиотек используемых платформ и свободно распространяемых библиотек.

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

**Системные компетенции:**

- Способен рефлексировать (оценивать и перерабатывать, анализировать и синтезировать) освоенные научные методы и способы деятельности для применения на практике (СК-М1).
- Способен предлагать концепции, модели, создавать и апробировать новые способы и инструменты профессиональной деятельности для применения на практике (СК-М2).
- Способен к самостоятельному освоению новых методов исследований, изменению научного и производственного профиля своей деятельности (СК-М3).
- Способен анализировать, верифицировать, оценивать полноту информации, найденной и полученной из различных источников в ходе профессиональной деятельности, при необходимости восполнять и синтезировать недостающую информацию (СК-М6).

#### **Инструментальные компетенции:**

- Способен проводить анализ, синтез, оптимизацию решений с целью обеспечения качества объектов профессиональной деятельности (ИК-М1.2.НИД (ПИ)).
- Способен планировать, управлять и контролировать выполнение требований (ИК-М2.1.АД (ПИ)).
- Способен выполнять проектную деятельность в области программной инженерии на основе системного подхода, уметь строить и использовать модели для описания и прогнозирования различных явлений, осуществлять их качественный и количественный анализ (ИК-М3.1.ПД (ПИ)).
- Способен оценить и выбрать методологию проектирования объектов профессиональной деятельности (ИК-М3.3.ПД (ПИ)).
- Способен применять современные технологии разработки программных комплексов с использованием автоматизированных систем планирования и управления, осуществлять контроль качества разрабатываемых программных продуктов (ИК-М4.1.ПТД\_ПИ2 (ПИ)).

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

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

- «Дискретная математика»,
- «Программирование»,
- «Построение и анализ алгоритмов»,
- «Архитектура вычислительных систем»,
- «Операционные системы».

## **5 Тематический план учебной дисциплины**

№	Название раздела	Всего часов	Аудиторные часы			Самостоятельная работа
			Лекции	Семинары	Практические занятия	
1	Предмет и методы системного программирования	45		15		30
2	Вычислительные машины	45		15		30
3	Инструменты поддержки жизненного цикла ПО. Управление требованиями. Методы проектирования программ на основе моделей	44		22		22

4	Методы проектирования программ на основе моделей. Методы верификации. Тестирование на основе моделей	50		20		30
5	Языки и компиляторы	60		20		40
6	Операционные системы	60		20		40
	<b>Итого:</b>	<b>304</b>		<b>112</b>		<b>192</b>

## 6 Формы контроля знаний студентов

Тип контроля	Форма контроля	3 год			
		Модуль 1	Модуль 2	Модуль 3	Модуль 4
Текущий (неделя)	Доклад		*	*	*
Итоговый	Экзамен				*

### 6.1 Критерии оценки знаний, навыков

Оценки по всем формам текущего контроля выставляются по 10-ти балльной шкале.

### 6.2 Порядок формирования оценок по дисциплине

Преподаватель оценивает работу студентов на семинарских занятиях: Оценки за работу на семинарских занятиях преподаватель выставляет в рабочую ведомость. Накопленная оценка по 10-ти балльной шкале за работу на семинарских занятиях определяется перед промежуточным или итоговым контролем -  $O_{\text{ауд}}$ .

Преподаватель оценивает самостоятельную работу студентов. Оценки за самостоятельную работу студента преподаватель выставляет в рабочую ведомость. Накопленная оценка по 10-ти балльной шкале за самостоятельную работу определяется перед итоговым контролем –  $O_{\text{сам. работа}}$ .

Накопленная оценка за текущий контроль учитывает результаты студента по текущему контролю следующим образом:

$$O_{\text{накопленная}} = 0,5 * O_{\text{ауд}} + 0,5 * O_{\text{сам. работа}}$$

На передаче студенту не предоставляется возможность получить дополнительный балл для компенсации оценки за текущий контроль.

На экзамене студент может получить дополнительный вопрос (дополнительную практическую задачу, решить к передаче домашнее задание), ответ на который оценивается в 1 балл.

В диплом выставляет результирующая оценка по учебной дисциплине, которая формируется по следующей формуле:

$$O_{\text{результ}} = 0,5 * O_{\text{накопл}} + 0,5 * O_{\text{экза}}.$$

## 7 Содержание дисциплины

№ п/п	Разделы и темы НИС	
1	Предмет и методы системного программирования	Понятие системного слоя программного обеспечения. Особенности разработки и эксплуатации системного ПО. Виды системного ПО и виды инструментов разработки и анализа системного ПО.
	Вычислительные машины	Многопроцессорные и многомашинные вычислительные системы. СуперЭВМ. Локальные и глобальные сети ЭВМ. Базовая эталонная модель взаимодействия открытых систем
2	Инструменты поддержки жизненного цикла ПО. Управление требованиями. Методы проектирования программ на основе моделей	Принципы системного проектирования комплексов программ. Системная и программная инженерия, процессы жизненного цикла сложных технических систем и программных комплексов. Декомпозиция требований, функций, процессов проектирования компонентов и комплексов программ. Повторное использование готовых компонентов при проектировании программных комплексов.
3	Методы проектирования программ на основе моделей. Методы верификации. Тестирование на основе моделей	Виды моделей и парадигмы моделирования. Исполнимые (явные) модели, модели ограничений, алгебраические модели. Подходы к разработке и верификации программ на основе моделей (MDA, MDSE, MBT, SBT). Процессы верификации компонентов и комплексов программ. Трассирование взаимодействия требований к компонентам в комплексах программ. Организация процессов тестирования компонентов и комплексов программ. Процессы и методы тестирования программных модулей и компонентов. Функциональная декомпозиция системы тестирования на основе моделей. Примеры известных приложений подхода к тестированию на основе моделей.
4	Языки и компиляторы	Языки и грамматики. Синтаксис и семантика языков программирования. Формальное определение грамматики и языка. Ассемблеры и загрузчики. Команды и псевдокоманды, символические адреса, адресные выражения. Характерные особенности языков программирования: С, С++, Java, С#, Python. Основные этапы работы компилятора: лексический анализ, синтаксический анализ и генерация промежуточного кода, генерация объектного кода, оптимизация кода. Промежуточное представление. Польская запись, тетрады, триады и деревья. Оптимизация программ. Основные методы оптимизации. Машиннозависимая и машиннонезависимая оптимизации. Генерация объектного кода. Нейтрализация семантических и синтаксических ошибок.
5	Операционные системы	Функции и основные понятия операционной системы. Процессы. Реализация процессов. Взаимодействие процессов. Параллельные процессы. Взаимное исключение и взаимная синхронизация. Примитивы синхронизации: события, семафоры, мониторы

		Хоара, почтовые ящики. Распределение времени процессора. Мультипрограммирование. Методы планирования в мультипрограммных системах. Управление памятью. Распределение памяти и организация доступа к памяти в ЭВМ с различной структурой памяти. Виртуальная память. Стратегии и методы замещения страниц. Ввод-вывод в файлы. Базисная и логическая системы управления файлами. Методы доступа к файлам. Планирование заданий.
6	Системы управления базами данных	Физическая организация данных. Перемешанные файлы. Индексированные файлы. В-деревья. Модели баз данных и их особенности: реляционная, сетевая и иерархическая модели. Управление данными в реляционной модели. Реляционная алгебра. Реляционное исчисление. Защита базы данных. Целостность. Секретность. Организация мультидоступа к базе данных. Транзакция. Синхрозахваты

## 8 Оценочные средства для текущего контроля и аттестации студента

### 8.1 Тематика заданий текущего контроля

Текущий контроль проводится в форме оценки докладов на семинаре.

По окончании учебного периода проводится экзамен в форме собеседования.

## 9 Учебно-методическое и информационное обеспечение дисциплины

### 9.1 Базовый учебник

### 9.2 Основная литература

1. Танненбаум Э. Архитектура компьютера. СПб.: Питер, 2006.
2. Танненбаум Э., Уэзеролл Дэвид. Компьютерные сети. СПб.: Питер, 2003.
3. В.В.Кулямин, Н.В.Пакулин, О.Л.Петренко, А.А.Сортов, А.В.Хорошилов. Формализация требований на практике. Препринт Института системного программирования РАН, №13, 2006.
4. В.В. Кулямин. Технологии программирования. Компонентный подход. М. Интернет-университет информационных технологий - БИНОМ. Лаборатория знаний, 2007.
5. В. В. Кулямин. Методы верификации программного обеспечения. Конкурс обзорно-аналитических статей по направлению «Информационно-телекоммуникационные системы», 2008.
6. Кнут Д. Искусство программирования. Том 1. Основные алгоритмы. Вильямс, 2010.
7. Ахо Альфред В., Хопкрофт Джон Ульман, Джеффри, Д. Структуры данных и алгоритмы: Пер. с англ. : Уч. пос. - М. : Издательский дом "Вильямс", 2000.
8. Кормен Т., Лейзерсон Ч., Ривест Р. Алгоритмы, построение и анализ. - М.: МЦНМО, 2000.
9. Стивенс Р., Раго С. UNIX. Профессиональное программирование. - СПб.: Символ-Плюс, 2007.

10. Соломон Д., Руссинович М. Внутреннее устройство Microsoft Windows: Windows Server 2003, Windows XP, Windows 2000. - СПб.: Питер, 2005.
11. Кузнецов С.Д. Базы данных: языки и модели. Учебник. М.: Бином-Пресс, 2008.

### 9.3 Дополнительная литература

12. Буздалов Д.В., Корныхин Е.В., Панфёров А.А., Петренко А.К., Хорошилов А.В. Практикум по дедуктивной верификации программ. М.-МАКС Пресс, 2014.
13. Липаев В.В. Программная инженерия сложных заказных программных продуктов // Учебное пособие. – М.: МАКС Пресс, 2014. – 312 с.  
(<http://www.ispras.ru/lipaev/books/Software%20Engineering%20of%20Complex%20Custom%20Software.pdf> и [http://www.directmedia.ru/book\\_278971\\_programmnaya\\_injeneriya\\_slojnyih\\_zakaznyih\\_programmnyih\\_produkтов/](http://www.directmedia.ru/book_278971_programmnaya_injeneriya_slojnyih_zakaznyih_programmnyih_produkтов/)).
14. М.У. Мандрыкин, В.С. Мутилин, А.В. Хорошилов. Введение в метод SEGAR — уточнение абстракции по контрпримерам. Труды Института системного программирования РАН Том 24. 2013 г. Стр. 219-292.
15. И.С. Захаров, М.У. Мандрыкин, В.С. Мутилин, Е.М. Новиков, А.К. Петренко, А.В. Хорошилов. Конфигурируемая система статической верификации модулей ядра операционных систем. Труды Института системного программирования РАН Том 26. Выпуск 2. 2014 г. Стр. 5-42.