**Санкт-Петербургский государственный университет**

**Р А Б О Ч А Я П Р О Г Р А М М А**

**УЧЕБНОЙ ДИСЦИПЛИНЫ**

Системное программирование для современных платформ

System Programming for Modern Platforms

**Язык(и) обучения**

русский

Трудоемкость в зачетных единицах: 2

Регистрационный номер рабочей программы: 003734

**Раздел 1. Характеристики учебных занятий**

**1.1. Цели и задачи учебных занятий**

Дисциплина «Системное программирование для современных платформ»

является одной из важных дисциплин цикла, формирующего подготовку специалиста как в области системного, так и прикладного программирования. Она представляет собой комплекс знаний, умений и навыков, позволяющих овладеть основными принципами лежащими в основе современных многозадачных, многопользовательских операционных систем.  
Таким образом, целью учебных занятий является:

• формирование у слушателей общего представление о содержании, задачах и методах современного системного программирования как самостоятельной научной и инженерной дисциплины, о диапазоне и разнообразии ее типичных приложений;

• дать обучающемуся общее представление об архитектуре современных ОС, основных принципах, используемых в системном программировании, а также показать их типичные применения;

• обеспечить формирование принципов системного, аналитического и алгоритмического принципов мышления и соответствующих навыков для работы в области системного программирования, необходимых для решения различных научных и практических задач;

• дать навыки постановки и решения задачи или проекта, отбора необходимых технических средств, обеспечения информационной безопасности программного обеспечения, а также формирование соответствующих компетенций, в том числе навыков работы в коллективе.

Поставленные цели достигаются путём решения следующих задач:

• изучение общих структур и подходов в предметных областей основных разделов системного программирования;

• ознакомление с методологиями и структурами данных соответствующих разделов системного программирования на примерах математических моделей и их приложений;

• развитие навыков самостоятельного решения практических задач.

**1.2. Требования подготовленности обучающегося к освоению содержания учебных занятий (пререквизиты)**

Программа курса предназначена для обучающихся 4 курса и рассчитана на слушателей, владеющих базовыми навыками работы с компьютером на уровне квалифицированного пользователя, знания языков С и С++, основных парадигм современного программирования.  
Максимальная эффективность программы будет обеспечена при условии, что обучающийся:

• владеет навыками администрирования Windows(R)- и UNIXTM-подобных операционных систем;

• имеет базовые представления об архитектуре процессоров семейства х86 и x64;

• владеет представлениями о базовых принципах организации локальных вычислительных сетей.

Дисциплина «Системное программирование для современных платформ» является базовым основным курсом в подготовке профессионального математика-программиста и служит основой для изучения других специальных математических дисциплин отделения прикладной математики и информатики.

**1.3. Перечень результатов обучения (learning outcomes)**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| № | Наименование категории (группы) компетенций | Код и наименование компетенции | Планируемые результаты обучения, обеспечивающие формирование компетенции | Код индикатора и индикатор достижения универсальной компетенции |
|  | 1 | 2 | 3 | 4 |
| 1 | Общепрофессиональные компетенции | ОПК-1 – способен применять естесвеннонаучные и общеинженерные знания, методы математического анализа и моделирования, теоретического и экспериментального исследования в профессиональной деятельности | знание содержания «Системное программирование для современных платформ» и наличие достаточных представлений о возможностях применения его разделов в различных прикладных областях | ОПК-1.1 Уметь идентифицировать возможные проблемы и пути их решения |
| 2 | Общепрофессиональные компетенции | ОПК-2 Способен понимать принципы работы современных информационных технологий и программных средств, в том числе отечественного производства, и использовать их при решении задач профессиональной деятельности | Знание принципов организации программных комплексов: СУБД, операционных систем, информационных систем; принципов взаимодействия их внутренних механизмов | ОПК-2.2 Уметь проводить формализацию и алгоритмизацию поставленных задач |
| 3 | Общепрофессиональные компетенции | ОПК-5 – способен инсталировать программное и аппаратное обеспечение для информационных и автоматизированных систем; | Навыки применения в профессиональной деятельности современных языков программирования и языков баз данных, операционных систем, электронных библиотек и пакетов программ, сетевых технологий | ОПК-5.1 Установка и настройка системного и прикладного ПО, необходимого для функционирования ИС в соответствии с трудовым заданием |
| 4 | Общепрофессиональные компетенции | ОПК-8 – способен осуществить поиск, хранение, обработку и анализ информации из различных источников и баз данных, представлять ее в требуемом формате с использованием информационных, компьютерных и сетевых технологий; | Умение использовать в научной и познавательной деятельности, а также в социальной сфере профессиональные навыки работы с информационными и компьютерными технологиями | ОПК-8.1 Оценка и выбор технологии доступа к данным |
| 5 | Профессиональные компетенции | ПКП-1 – способен проектировать программные системы; | Навыки применения в профессиональной деятельности современных языков программирования и СУБД, методологии системной инженерии, систем автоматизации проектирования, электронных библиотек и коллекций, сетевых технологий, библиотек и пакетов программ, современных профессиональных стандартов информационных технологий | ПКП-1.1 Уметь разрабатывать технические спецификации на программные компоненты и их взаимодействие |
| 6 | Профессиональные компетенции | ПКП-2 – способен использовать основные модели информационных технологий и способы их применения для решения задач в предметных областях; | умение решать основные задачи системного программирования, включая этапы постановки и решения задачи, отбора необходимых технических средств, а также навыков работы в коллективе | ПКП-2.1 Уметь описывать алгоритмы компонентов, включая методы и схемы |
| 7 | Профессиональные компетенции | ПКП-3 – способен разрабатывать моделирующие алгоритмы и реализации их на базе языков и пакетов прикладных программ моделирования; | Умение профессионально решать задачи производственной и технологической деятельности с учетом современных достижений науки и техники, включая: разработку алгоритмических и программных решений в области системного и прикладного программирования; разработку математических, информационных и имитационных моделей по тематике выполняемых исследований; создание информационных ресурсов глобальных сетей, образовательных контентов, прикладных баз данных; разработку тестов и средств тестирования систем и средств на соответствие стандартам и исходным требованиям; разработку эргономичных человеко-машинных интерфейсов; | ПКП-3.1 Разработка модели бизнес-процессов заказчика |
| 8 | Профессиональные компетенции | ПКП-4 – способен выбирать архитектуру и комплексирование современных компьютеров, систем, комплексов и сетей системного администрирования; | Навыки осуществлять целенаправленный поиск информации о новейших научных и технологических достижениях в сети Интернет, способность взаимодействовать и сотрудничать с профессиональными сетевыми сообществами и международными консорциумами, отслеживать динамику развития выбранных направлений области информационных технологий | ПКП-4.1 Оценка и выбор варианта архитектуры программного средства |
| 9 | Профессиональные компетенции | ПКП-6 – способен формировать суждения о проблемах современной информатики, ее категорий и связей с другими научными дисциплинами; | знакомство с базовыми представлениями о жизненном цикле программ, технологии программирования | ПКП-6.1 Уметь разрабатывать процедуры проверки работоспособности и измерения характеристик программного обеспечения |
| 10 | Профессиональные компетенции | ПКП-7 – способен систематизировать и применять знания о содержании основных этапов и тенденций развития программирования, математического обеспечения и информационных технологий; | Знание современных тенденций развития программного обеспечения широкого диапазона типов вычислительных систем, в том числе суперкомпьютерных комплексов | ПКП-7.1 Умеет проводить технические и управленческие ревизии создаваемого программного средства |
| 11 | Универсальные компетенции | УК-1 – способен осуществлять поиск, критический анализ и синтез информации, применять системный подход для решения поставленных задач; | Нвыки работы с информацией из различных источников, включая сетевые ресурсы Интернет для решения профессиональных задач | УК 1.5. Грамотно, логично, аргументированно формирует собственные суждения, решения и оценки. |
| 12 | Универсальные компетенции | УК-2 – способен определять круг задач в рамках поставленной цели и выбирать оптимальные способы их решения, исходя из действующих правовых норм, имеющихся ресурсов и ограничений; | Умение критически переосмысливать свой опыт, адаптироваться к различным ситуациям, проявлять творческий подход, инициативу и настойчивость в достижении целей профессиональной деятельности | УК-2.4. Планирует реализацию задач в зоне своей ответственности с учетом имеющихся ресурсов и ограничений, действующих правовых норм; |
| 13 | Универсальные компетенции | УКБ-1 – способен участвовать в разработке и реализации проектов, в т.ч. предпринимательских; | Умение реализовывать решения, направленные на поддержку социально значимых проектов, на повышение электронной грамотности населения, обеспечения общедоступности информационных услуг | УКБ-1.5. Выполняет задачи в зоне своей ответственности в соответствии с запланированными результатами и точками контроля; |
| 14 | Универсальные компетенции | УКБ-3 - Способен понимать сущность и значение информации в развитии общества, использовать основные методы получения и работы с информацией с учетом современных технологий цифровой экономики, искусственного интеллекта и науки о данных, а также информационной безопасности | знание сущности и значения информации в развитии современного информационного общества; | УКБ-3.4. Проверяет достоверность собранной информации. |

**1.4. Перечень и объём активных и интерактивных форм учебных занятий**

Лекции – 10 ак. ч.

Часть материала может подаваться в форме интерактивной дискуссии между обучающимися под контролем преподавателя и на заданную им тему, направленной на самостоятельный поиск тех или иных решений актуальных вопросов системного программирования.

Также часть материала подаётся в виде готового к компиляции программного кода (входит в состав методических материалов, подготовленных преподавателем). Часть этого программного кода разбирается преподавателем, часть может быть предоставлена обучающимся на самостоятельное рассмотрение.

**Раздел 2. Организация, структура и содержание учебных занятий**

**2.1. Организация учебных занятий**

**2.1.1 Основной курс**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Трудоёмкость, объёмы учебной работы и наполняемость групп обучающихся | | | | | | | | | | | | | | | | | | | | | |
| Код модуля в составе дисциплины,  практики и т.п. | Контактная работа обучающихся с преподавателем | | | | | | | | | | | | | Самостоятельная работа | | | | | Объём активных и интерактивных  форм учебных занятий | | Трудоёмкость |
| лекции | семинары | консультации | практические  занятия | лабораторные работы | контрольные работы | коллоквиумы | текущий контроль | промежуточная  аттестация | итоговая аттестация | под руководством преподавателя | в присутствии  преподавателя | сам. раб. с использованием  методических материалов | | текущий контроль (сам.раб.) | промежуточная аттестация (сам.раб.) | итоговая аттестация  (сам.раб.) |  | |  | |
| ОСНОВНАЯ ТРАЕКТОРИЯ | | | | | | | | | | | | | | | | | | | | | |
| Форма обучения: очная | | | | | | | | | | | | | | | | | | | | | |
| Семестр 7 | 60 |  |  |  |  |  |  |  | 2 |  |  |  |  | |  | 10 |  | 10 | | 2 | |
|  | 2-20 |  |  |  |  |  |  |  | 2-20 |  |  |  |  | |  | 1-1 |  |  | |  | |
| ИТОГО | 60 |  |  |  |  |  |  |  | 2 |  |  |  |  | |  | 10 |  |  | | 2 | |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Виды, формы и сроки текущего контроля успеваемости и промежуточной аттестации | | | | | | |
| Код модуля в составе дисциплины, практики и т.п. | Формы текущего контроля успеваемости | | Виды промежуточной аттестации | | Виды итоговой аттестации  (только для программ итоговой аттестации и дополнительных образовательных программ) | |
| Формы | Сроки | Виды | Сроки | Виды | Сроки |
| ОСНОВНАЯ ТРАЕКТОРИЯ | | | | | | |
| Форма обучения: очная | | | | | | |
| Семестр 7 |  |  | зачёт, устно, традиционная форма | по графику промежуточной аттестации |  |  |

**2.2. Структура и содержание учебных занятий**

**2.1.1 Основной курс**

|  |  |  |  |
| --- | --- | --- | --- |
| Период обучения: **Семестр 7** | | | |
| **№ п.п.** | **Наименование темы (раздела, части)** | **Вид учебных занятий** | **Кол-во часов** |
| 1 | **Тема 1. Введение. Понятие о системном программировании** | лекции | 4 |
| 2 | **Тема 2. Особенности архитектуры современных ОС** | лекции | 8 |
| 3 | **Тема 3. Сигналы** | лекции | 6 |
| 4 | **Тема 4. Низкоуровневая подсистема ввода-вывода.** | лекции | 12 |
| 5 | **Тема 5. Ввод-вывод через прямое отображение в память** | лекции | 4 |
| 6 | **Тема 6. Программный механизм сокетов.** | лекции | 6 |
| 7 | **Тема 7. System V interprocess communication.** | лекции | 6 |
| 8 | **Тема 8. Remote Procedure Call framework.** | лекции | 6 |
| 9 | **Тема 9. Многопотоковое программирование** | лекции | 8 |
| 10 | **П****ромежуточн****а****я** **аттестации** | Самостоятельная работа | 10 |
| Зачёт | 2 |
|  | **Всего** | Лекции | 60 |
| Самостоятельная работа | 10 |
| Промежуточная аттестация | 2 |

На лекционных занятиях преподаватель представляет материал курса согласно следующему содержанию:

**I. «Введение. Понятие о системном программировании»:**

Особенности системного программирования в Windows(R)- и UNIXTM-подобных ОС. Стандарты системного программирования. Компиляторы и среды разработки. Основные средства управления программными проектами в UNIX(R)- подобных ОС. Среда разработки gcc.

**II. «Особенности архитектуры современных ОС»:**

Основные принципы программирования в многозадачной рабочей среде. Процессы. Уровни привелегий. Понятия «ядерного» и пользовательского кода. Основные компоненты ядра, обеспечивающие многозадачность: планировщик задач и виртуальная машина. Взаимодействие пользовательских процессов с ядром. Системные вызовы. Понятие взаимодействия между процессами. Порождение процессов. Семейства системных вызовов *fork* и *exec*. Родительские и дочерние процессы. Понятие потока управления. Жизненный цикл процесса. Процесс как конечный автомат.

**III. «Сигналы»:**

Предназначение сигналов. Системный вызов *kill.* «Старый» и «новый» механизмы регистрации обработчиков сигналов.

**IV. «Низкоуровневая подсистема ввода-вывода»:**

Файловые системы. Права доступа. ACL и супербиты. Отказоустойчивые файловые системы. Сравнение файлового ввода-вывода средствами стандартной библиотеки языка С и низкоуровневого интерфейса программирования ядра. Файловые дескрипторы. Системные вызовы низкоуровневого доступа к файлам. Использование специальных файлов для организации ввода-вывода данных периферийных устройств. Программирование ввода-вывода и использованием каналов (FIFO-файлов). Программный интерфейс работы с каталогами файловых систем.

**V. «Ввод-вывод через прямое отображение в память»:**

Применение файлового ввода-вывода посредством файлов, непосредственно отображаемых в память. API непосредственного отображения в память.

**VI. «Программный механизм сокетов»:**

Сетевые соединения. Модель ISO/OSI. Понятие сокета. Модель взаимодействия «клиент-сервер». Интерфейс программирования сокетов. TCP- и UDP-сокеты. Особенности программирования сокетов и часто встречающиеся ошибки.

**VII. «System V interprocess communication»:**

Цели создания IPC. Задачи, решаемые с его помощью. Объекты IPC: сообщения, семафоры, разделяемая память. Связь с сигналами и вводом-выводом с отображением в память.

**VIII. «Remote Procedure Call framework»:**

Предназначение и задачи, решаемые посредством механизма RPC. XDR-формат. Уровни абстракции RPC. IPC-препроцессор, машинно-независимое представление данных в RPC. Пример RPC программы клиент-сервер.

**IX. «Многопотоковое программирование»:**

Сравнение многозадачного и многопотокового подходов в программировании в многозадачной рабочей среде. posix- и solaris-подобные многопотоковые библиотеки. Создание, присоединение и отсоединение потоков. Блокировка данных в многопотоковой среде. Мьютексы, семафоры и условные переменные.

**Раздел 3. Обеспечение учебных занятий**

**3.1. Методическое обеспечение**

**3.1.1 Методические указания по освоению дисциплины**

Успешное освоение дисциплины возможно благодаря посещению лекций, участию в обсуждении вопросов, подготовленных к занятию, самостоятельной работе, включающей в себя чтение специальной литературы по разделам темы, использование презентаций по тематике курса, подготовленных преподавателем.

**3.1.2 Методическое обеспечение самостоятельной работы**

Самостоятельная работа обучающихся в рамках данной дисциплины является важным компонентом обучения, предусмотренным компетентностно-ориентированным учебным планом и рабочей программой учебной дисциплины.

Одна из форм самостоятельной работы – это подготовка докладов по тематике курса и источникам, указанным в обязательной, дополнительной литературе и интернет-источниках, указанных с данной программе. Доклад читается на зачёте и является его альтернативной формой.

**3.1.3 Методика проведения текущего контроля успеваемости и промежуточной аттестации и критерии оценивания**

Промежуточная аттестация представляет собой зачёт в устной форме. Билет к зачёту состоит из двух вопросов. Время подготовки ответа на вопросы билета составляет не менее 1 академического часа. Использование конспектов и учебников, а также электронных устройств хранения, обработки или передачи информации при подготовке и ответе на вопросы зачёта запрещено.

После ответа на основные вопросы билета, преподаватель вправе задать уточняющие вопросы по услышанному. Затем преподаватель задаёт дополнительные вопросы по любой теме из списка вопросов, вынесенных на зачёт. В качестве дополнительных используются вопросы, не требующие длительного ответа, в том числе основные определения и понятия. Рекомендуется задавать 2-3 дополнительных вопроса на билет и не более 1-2 уточняющих вопросов. Преподаватель вправе увеличить количество вопросов в случае, если у него возникает обоснованное сомнение, освоил обучающийся материал учебного курса или нет.

В качестве альтернативной формы зачёта допускается по согласованию с преподавателем подготовка обучающимся доклада по тематике курса, включающем материал из дополнительных источников, не раскрытый на лекциях. В этом случае на доклад отводится 15-20 мин. Обучающемуся разрешается использовать собственный конспект доклада.

В любой момент обучающийся имеет право отказаться от сдачи зачёта с выставлением оценки «незачёт».

По желанию преподавателя на зачёт допустимо приглашение других преподавателей с квалификацией не ниже, изложенной в п. 3.2.1, как для независимого оценивания ответов обучающихся, так и для коллегиального. В последнем случае оценка за зачёт ставится на основании голосования простого большинства. В спорных ситуациях преподаватель, ведущий дисциплину, принимает окончательное решение.

Критерии выставления оценки:

|  |  |  |
| --- | --- | --- |
| Дан полный ответ на вопросы по билету, приведены обоснования необходимых утверждений, обнаружено всестороннее, глубокое и систематическое знание учебного материала. Обучающийся усвоил взаимосвязь основных понятий курса, проявил творческие способности в понимании и использовании учебного материала. | A | зачёт |
| Допущены неточности в ответе, непринципиальные ошибки, исправленные самостоятельно после наводящих вопросов, обоснования утверждений приведены схематично. Обучающийся способен к самостоятельному пополнению и обновлению знаний по курсу в ходе дальнейшей учебы и профессиональной деятельности. | B |
| Допущены неточности в ответе, обоснования утверждений приведены схематично. Обучающийся способен к самостоятельному пополнению и обновлению знаний по курсу в ходе дальнейшей учебы и профессиональной деятельности. | C |
| Допущены существенные ошибки, но обучающийся обладает необходимыми знаниями для их устранения под руководством преподавателя. Знание учебного материала в объеме, необходимом для дальнейшей учебы и профессиональной деятельности (основные определения, формулы, формулировки утверждений). Обучающийся знаком с основной литературой, рекомендованной программой. | D |
| Допущены грубые ошибки. Знание учебного материала в объеме, необходимом для дальнейшей учебы и профессиональной деятельности (основные определения, формулы, формулировки утверждений). | E |
| В остальных случаях | F | незачёт |

**3.1.4 Методические материалы для проведения текущего контроля успеваемости и промежуточной аттестации (контрольно-измерительные материалы, оценочные средства)**

Примерный краткий перечень вопросов к зачёту:

1. сегментная и страничная организация памяти;
2. архитектура современных ОС, основные компоненты ядра ОС;
3. форматы исполняемых файлов COFF и ELF, струкутура образа процесса;
4. механизм взаимодействия между пользовательскими приложениями и ядром ОС, системные вызовы;
5. права доступа файловой системы, ACL и супербиты;
6. системные вызовы *fork и exec;*
7. использование низкоуровневого ввода-вывода для организации взаимодействия между процессами;
8. системные вызовы *open, create close;*
9. системные вызовы *read, write, lseek;*
10. системные вызовы *pipe, dup, dup2;*
11. системные вызовы *fcntl, access;*
12. программный интерфейс работы с каталогами файловых систем;
13. файловый ввод-вывод с отображением в память;
14. System V IPC. Идеология. Пространство имён;
15. System V IPC объекты: IPC-сообщения, IPC-семафоры, IPC-разделяемая память;
16. Framework RPC, уровни абстракции RPC;
17. RPC-компиллятор *rcgen*, XDR-формат представления данных;
18. пример RPC-программы;
19. сигналы как механизм взаимодействия между процессами, системный вызов *kill;*
20. «классический» метод обработки сигналов;
21. posix-стандартизированный метод обработки сигналов;
22. сравнение модели сетевого стека ISO/OSI и TCP/IP;
23. программный интерфейс сокетов, TCP- и UDP-сокеты;
24. многозадачная и многопотоковая парадигмы программирования;
25. системные вызовы pthread\_create, pthread\_exit, pthread\_detach, pthread\_join;
26. сигналы в многопотоковых процессах, системные вызовы pthread\_sigmask, pthread\_kill;
27. механизмы синхронизации в многопотоковом программировании: мьютексы, семафоры и условные переменные.

|  |  |  |
| --- | --- | --- |
| № | Код индикатора и индикатор достижения универсальной компетенции | Контрольно-измерительные материалы (КИМ) (тестовые вопросы, контрольные задания, кейсы и пр.) |
|  | 1 | 2 |
| 1 | ОПК-1.1 Уметь идентифицировать возможные проблемы и пути их решения | ответы на вопросы зачёта и на дополнительные вопросы оцениваются по шкале от 0 (нет ответа) до 10 (очень хорошо), далее оценка усредняется и переводится в диапазон от 0 до 100. |
| 2 | ОПК-2.2 Уметь проводить формализацию и алгоритмизацию поставленных задач | обучающемуся предлагают спроектировать многопоточную систему с нетривиальным вводом-выводом и выбрать подходящий API из рассмотренных в курсе. Результат оценивается по шкале от 0 (не сделано) до 100 (очень хорошо). |
| 3 | ОПК-5.1 Установка и настройка системного и прикладного ПО, необходимого для функционирования ИС в соответствии с трудовым заданием | обучающемуся предлагают установить среду разработки и продемонстрировать использование API на практике. Результат оценивается по шкале от 0 (не сделано) до 100 (очень хорошо). |
| 4 | ОПК-8.1 Оценка и выбор технологии доступа к данным | обучающемуся предлагают установить среду разработки и продемонстрировать использование API на практике. Результат оценивается по шкале от 0 (не сделано) до 100 (очень хорошо). |
| 5 | ПКП-1.1 Уметь разрабатывать технические спецификации на программные компоненты и их взаимодействие | обучающемуся предлагают спроектировать многопоточную систему с нетривиальным вводом-выводом и выбрать подходящий API из рассмотренных в курсе. Результат оценивается по шкале от 0 (не сделано) до 100 (очень хорошо). |
| 6 | ПКП-2.1 Уметь описывать алгоритмы компонентов, включая методы и схемы | обучающемуся предлагают спроектировать многопоточную систему с нетривиальным вводом-выводом и выбрать подходящий API из рассмотренных в курсе. Результат оценивается по шкале от 0 (не сделано) до 100 (очень хорошо). |
| 7 | ПКП-3.1 Разработка модели бизнес-процессов заказчика | обучающемуся предлагают установить среду разработки и продемонстрировать использование API на практике. Результат оценивается по шкале от 0 (не сделано) до 100 (очень хорошо). |
| 8 | ПКП-4.1 Оценка и выбор варианта архитектуры программного средства | обучающемуся предлагают спроектировать многопоточную систему с нетривиальным вводом-выводом и выбрать подходящий API из рассмотренных в курсе. Результат оценивается по шкале от 0 (не сделано) до 100 (очень хорошо). |
| 9 | ПКП-6.1 Уметь разрабатывать процедуры проверки работоспособности и измерения характеристик программного обеспечения | ответы на вопросы зачёта и на дополнительные вопросы оцениваются по шкале от 0 (нет ответа) до 10 (очень хорошо), далее оценка усредняется и переводится в диапазон от 0 до 100. |
| 10 | ПКП-7.1 Умеет проводить технические и управленческие ревизии создаваемого программного средства | обучающемуся предлагают установить среду разработки и продемонстрировать использование API на практике. Результат оценивается по шкале от 0 (не сделано) до 100 (очень хорошо). |
| 11 | УК 1.5. Грамотно, логично, аргументированно формирует собственные суждения, решения и оценки. | обучающемуся предлагают установить среду разработки и продемонстрировать использование API на практике. Результат оценивается по шкале от 0 (не сделано) до 100 (очень хорошо). |
| 12 | УК-2.4. Планирует реализацию задач в зоне своей ответственности с учетом имеющихся ресурсов и ограничений, действующих правовых норм; | обучающемуся предлагают спроектировать многопоточную систему с нетривиальным вводом-выводом и выбрать подходящий API из рассмотренных в курсе. Результат оценивается по шкале от 0 (не сделано) до 100 (очень хорошо). |
| 13 | УКБ-1.5. Выполняет задачи в зоне своей ответственности в соответствии с запланированными результатами и точками контроля; | ответы на вопросы зачёта и на дополнительные вопросы оцениваются по шкале от 0 (нет ответа) до 10 (очень хорошо), далее оценка усредняется и переводится в диапазон от 0 до 100. |
| 14 | УКБ-3.4. Проверяет достоверность собранной информации. | ответы на вопросы зачёта и на дополнительные вопросы оцениваются по шкале от 0 (нет ответа) до 10 (очень хорошо), далее оценка усредняется и переводится в диапазон от 0 до 100. |

**3.1.5 Методические материалы для оценки обучающимися содержания и качества учебного процесса**

Для оценки обучающимися содержания и качества учебного процесса применяется анкетирование в соответствии с методикой и графиком, утвержденными в установленном порядке.

**3.2. Кадровое обеспечение**

**3.2.1 Образование и (или) квалификация штатных преподавателей и иных лиц, допущенных к проведению учебных занятий**

К чтению лекций привлекаются преподаватели, имеющие базовое образование и/или ученую степень соответствующую профилю преподаваемой дисциплины.

**3.2.2 Обеспечение учебно-вспомогательным и (или) иным персоналом**

не требуется.

**3.3. Материально-техническое обеспечение**

**3.3.1 Характеристики аудиторий (помещений, мест) для проведения занятий**

Учебные аудитории для проведения учебных занятий, оснащенные стандартным оборудованием, используемым для обучения в СПбГУ в соответствии с требованиями материально-технического обеспечения.

**3.3.2 Характеристики аудиторного оборудования, в том числе неспециализированного компьютерного оборудования и программного обеспечения общего пользования**

Стандартное оборудование, используемое для обучения в СПбГУ. MS Windows, MS Office, Mozilla FireFox, Google Chrome, Acrobat Reader DC, WinZip, Антивирус Касперского

**3.3.3 Характеристики специализированного оборудования**

Специализированное оборудование не требуется.

**3.3.4 Характеристики специализированного программного обеспечения**

Специализированное программное обеспечение не требуется.

**3.3.5 Перечень и объёмы требуемых расходных материалов**

Для аудиторий с маркерными досками необходимы стирающиеся маркеры в объеме, достаточном для проведения курса. Для аудиторий с меловыми досками необходим мел в объеме, достаточном для проведения курса. Канцелярские принадлежности в объеме, достаточном для проведения курса.

**3.4. Информационное обеспечение**

**3.4.1 Список литературы**

1. **Таненбаум, Эндрю . Современные операционные системы** / Э. Таненбаум. - 3-е изд. - М. ; СПб. ; Нижний Новгород : Питер, 2007-2013. - 1115 с.

2. [**Олифер, Виктор Григорьевич**](https://litkarta.kraslib.ru/cgi-bin/irbis64r/irbis64r_91/cgiirbis_64.exe?LNG=&Z21ID=&I21DBN=EKU&P21DBN=EKU&S21STN=1&S21REF=2&S21FMT=fullwebr&C21COM=S&S21CNR=328&S21P01=0&S21P02=1&S21P03=A=&S21STR=Олифер%20Виктор%20Григорьевич). Компьютерные сети. Принципы, технологии, протоколы: [учебное пособие для студентов высших учебных заведений] / В. Г. Олифер, Н. А. Олифер. - 3-е издание. - Санкт-Петербург: Питер, 2006-2011. - 957 с.: ил. - (Учебник для вузов).

3. **Брукс Ф. Мифический человеко-месяц или как создаются программные системы.**- СПб, 2007. – 298 с.

**3.4.3 Перечень иных информационных источников**

1. Документация среды разработки GNU ([http://www.gnu.org](http://www.gnu.org/))
2. <http://citforum.ru/operating_systems/sos/contents.shtml> Н. А. Олифер, В. Г. Олифер. «Сетевые операционные системы»,
3. <http://www.intuit.ru/department/se/baseppvs2010> / А. А. Алексеев. «Основы параллельного программирования с использованием Visual Studio 2010»
4. <http://www.intuit.ru/department/hardware/paralltech/> А. Б. Барский. «Архитектура параллельных вычислительных систем»  
   **- ЭР открытого доступа в сети Интернет**
5. Сайт Научной библиотеки им. М. Горького СПбГУ: <http://www.library.spbu.ru/>
6. Электронный каталог Научной библиотеки им. М. Горького СПбГУ: <http://www.library.spbu.ru/cgi-bin/irbis64r/cgiirbis_64.exe?C21COM=F&I21DBN=IBIS&P21DBN=IBIS>
7. Перечень электронных ресурсов, находящихся в доступе СПбГУ: <http://cufts.library.spbu.ru/CRDB/SPBGU/>
8. Перечень ЭБС, на платформах которых представлены российские учебники, находящиеся в доступе СПбГУ: <http://cufts.library.spbu.ru/CRDB/SPBGU/browse?name=rures&resource%20type=8>

**Раздел 4. Разработчики программы**

Шилов Сергей Юрьевич, старший преподаватель кафедры системного программирования. s.shilov@spbu.ru