**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: 002292

Санкт-Петербург

2020

**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 областях 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, 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)**

В процессе изучения дисциплины «Системное программирование в современных ОС» обучаемые приобретают следующие знания:

* знание содержания «Системное программирование для современных платформ» и наличие достаточных представлений о возможностях применения его разделов в различных прикладных областях;
* знакомство с базовыми представлениями о жизненном цикле программ, технологии программирования;
* знание сущности и значения информации в развитии современного информационного общества;
* знание способов и средств получения, хранения, переработки информации, сознавать опасности и угрозы, возникающие в этом процессе, соблюдать основные требования информационной безопасности, в том числе защиты государственной тайны;
* современных тенденций развития программного обеспечения широкого диапазона типов вычислительных систем, в том числе суперкомпьютерных комплексов;
* современных методов анализа и синтеза сложных проектов и проектирования программных средств для решения современных задач в различных прикладных областях;
* современных парадигм программирования, языков программирования и базовых алгоритмов для реализации сложных проектов;
* принципов организации программных комплексов: СУБД, операционных систем, информационных систем; принципов взаимодействия их внутренних механизмов.

Умения:

* умение решать основные задачи системного программирования, включая этапы постановки и решения задачи, отбора необходимых технических средств, а также навыков работы в коллективе;
* умение формализовать поставленные задачи и реализовывать сложные программные комплексы при работе в коллективе разработчиков;
* владеть основными методами, способами и средствами получения, хранения, переработки информации, иметь навыки работы с компьютером как средством управления информацией, в том числе в глобальных компьютерных сетях;
* соблюдать основные требования информационной безопасности, в том числе защиты государственной тайны;
* реализовывать решения, направленные на поддержку социально значимых проектов, на повышение электронной грамотности населения, обеспечения общедоступности информационных услуг;
* использовать в научной и познавательной деятельности, а также в социальной сфере профессиональные навыки работы с информационными и компьютерными технологиями;
* использовать нормативные правовые документы в своей деятельности, действовать в условиях гражданского общества;
* критически переосмысливать свой опыт, адаптироваться к различным ситуациям, проявлять творческий подход, инициативу и настойчивость в достижении целей профессиональной деятельности;
* делать анализ и грамотную оценку эффективности разрабатываемых алгоритмов;
* понимать и анализировать мировоззренческие, социально и личностно значимые философские проблемы, имеющие отношение к заявленной предметной области.

Навыки:

* работы с информацией из различных источников, включая сетевые ресурсы Интернет для решения профессиональных задач;
* осуществления целенаправленного поиска информации о технологических достижениях в сети Интернет и из других источников;
* 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;
* 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;
* 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;
* владеть способностью применять на практике международные и профессиональные стандарты информационных технологий, современные парадигмы и методологии, инструментальные и вычислительные средства;
* применения в профессиональной деятельности современных языков программирования и языков баз данных, операционных систем, электронных библиотек и пакетов программ, сетевых технологий;
* владеть способностью формировать суждения о значении и последствиях своей профессиональной деятельности с учетом социальных, профессиональных и этических позиций;
* взаимодействия с коллегами, работы в коллективе.

Курс способствует развитию следующих компетенций:

* ОПК-1: способен применять фундаментальные знания, полученные в области математических и (или) естественных наук, и использовать их в профессиональной деятельности;
* ОПК-3: способен применять современные информационные технологии, в том числе отечественные, при создании программных продуктов и программных комплексов различного назначения;
* ОПК-4: способен участвовать в разработке технической документации программных продуктов и программных комплексов;
* ОПК-5: способен инсталлировать и сопровождать программное обеспечение для информационных систем и баз данных, в том числе отечественного производства;
* ПКА-1: способен демонстрировать базовые знания математических и естественных наук, программирования и информационных технологий;
* ПКП-1: способность проводить под научным руководством исследование на основе существующих методов в конкретной области профессиональной деятельности;
* ПКП-2: способен решать задачи в области развития науки, техники и технологии с учетом нормативного правового регулирования в сфере интеллектуальной собственности;
* ПКП-3: способен преподавать математику и информатику в средней школе, специальных учебных заведениях на основе полученного фундаментального образования и научного мировоззрения;
* ПКП-4: способен применять современные информационные технологии при проектировании, реализации, оценке качества и анализа эффективности программного обеспечения для решения задач в различных предметных областях;
* ПКП-5: способен использовать основные методы и средства автоматизации проектирования, реализации, испытаний и оценки качества при создании конкурентоспособного программного продукта и программных комплексов, а также способен использовать методы и средства автоматизации, связанные с сопровождением, администрированием и модернизацией программных продуктов и программных комплексов;
* ПКП-8: способен использовать современные методы разработки и реализации конкретных алгоритмов математических моделей на базе языков программирования и пакетов прикладных программ моделирования;
* 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 и информационной безопасности.

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

Лекции – 4 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 орм учебных занятий | Трудоёмкость в зачетных ед. |
| 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 | 30 |  |  |  |  |  |  |  | 2 |  |  |  | 34 |  | 6 |  | 4 | 2 |
|  | 2-100 |  |  |  |  |  |  |  | 2-12 |  |  |  | 1-1 |  | 1-1 |  | 2-100 |  |
| ИТОГО | 30 |  |  |  |  |  |  |  | 2 |  |  |  | 34 |  | 6 |  | 4 | 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**

Базовый курс Основная траектория Очная форма обучения

Период обучения: **Семестр 7**

|  |  |  |  |
| --- | --- | --- | --- |
| № п.п. | Наименование темы (раздела, части) | Вид учебных занятий | Кол-во часов |
| 1 | Введение. Понятие о системном программировании | лекции | 2 |
| 2 | Особенности архитектуры современных ОС | лекции | 4 |
| 3 | Сигналы | лекции | 3 |
| 4 | Низкоуровневая подсистема ввода-вывода. | лекции | 6 |
| 5 | Ввод-вывод через прямое отображение в память | лекции | 2 |
| 6 | Программный механизм сокетов. | лекции | 3 |
| 7 | System V Interprocess communication. | лекции | 3 |
| 8 | Remote Procedure Call framework. | лекции | 3 |
| 9 | Многопотоковое программирование | лекции | 4 |
| 10 | Промежуточная аттестации | Самостоятельная работа | 6 |
| Зачёт | 2 |
|  | Всего | Лекции | 30 |
| Самостоятельная работа в т.ч. с использованием учебно-методич. материалов | 34 |
| Промежуточная аттестация | 8 |
| **Итого** | | | **72** |

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 UNIXTM- 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. API SysVRPC.

**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.

**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. Слайды к лекциям размещены на сайте Кафедры системного программирования по адресу: <https://oops.math.spbu.ru/SE/Members/ssh/ssh>.

**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.

Тематика возможных докладов совпадает с формулировками вопросов к зачёту.

**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 преподаванию дисциплины допускаются преподаватели, имеющие диплом о высшем образовании, соответствующий профилю преподаваемой дисциплины.

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

Оператор проекционного оборудования.

**3.3. Материально-техническое обеспечение**

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

Стандартно оборудованные лекционные аудитории для проведения интерактивных лекций: видеопроектор, экран, др. оборудование.

Аудитории должны соответствовать требованиям СанПиН 2.2.2/2.4.1340-03 в редакции от 21 июня 2016 года.

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

В аудиториях, где проводятся лекционные занятия, необходимо наличие досок и средств письма на них. Для показа слайдов необходим компьютер с установленным программным обеспечением для работы со слайдами в форматах PDF, PPT, PPTX и подключенный к нему мультимедиа проектор с экраном.

**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. Робачевский А.М., Операционная система Unix. - СПб.: BHV- Санкт-Петербург, 1998.-528 с., ил - ISBN 5-8206-0030-4
2. «Unix и Linux. Руководство системного администратора», 4-е издание, Эви Немет, Трент Хейн, Гарт Снайдер, Бэн Уэйли, 1312 стр., ISBN 978-5-8459-1740-9, «ВИЛЬЯМС», 2012
3. Харт Д. Системное программирование в среде Windows, 3-е издание. - Пер. с англ. - М.: Издательский дом "Вильямс", 2005. - 592 с.: ил. - ISBN 5-8459-0879-5.
4. Собелл Марк. Практическое руководство по Red Hat Linux: Fedora Core и Red Hat Enterprise Linux: руководство / М. Г. Собелл. - 2-е изд. - М.; СПб.; Киев: Вильямс, 2005. - 1071 с.
5. Таненбаум, Эндрю. Современные операционные системы / Э. Таненбаум. - 3-е изд. - М.; СПб.; Нижний Новгород: Питер, 2007-2013. - 1115 с.
6. [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. - 957 с.: ил. - (Учебник для вузов).
7. Брукс Ф. Мифический человеко-месяц или как создаются программные системы. - СПб, 2007. – 298 с.

**3.4.3 Перечень иных информационных источников**

1. http://www.scert.ru/files/1500d.pdf В. В. Воеводин, Вл. В. Воеводин. Параллельные вычисления.
2. http://microsat.sm.bmstu.ru/e-library/Books/TheMythicalManMonth\_rus/The%20Mythical%20Man-Month.pdf Ф.П. Брукс «Мифический человеко-месяц» (электронный текст книги)
3. Linux documentation project. ([http://www.linuxdoc.org](http://www.linuxdoc.org/))
4. Документация среды разработки GNU ([http://www.gnu.org](http://www.gnu.org/))
5. http://citforum.ru/operating\_systems/sos/contents.shtml Н. А. Олифер, В. Г. Олифер. «Сетевые операционные системы»,
6. http://www.intuit.ru/department/se/baseppvs2010/ А. А. Алексеев. «Основы параллельного программирования с использованием Visual Studio 2010»
7. http://www.intuit.ru/department/hardware/paralltech/ А. Б. Барский. «Архитектура параллельных вычислительных систем»

**Раздел 4. Разработчики программы**

Шилов Сергей Юрьевич, старший преподаватель кафедры системного программирования, mailto:s.shilov@spbu.ru (serguei.shilov@gmail.com), тел. +7-921-7957384