**Санкт-Петербургский государственный университет**

**Р А Б О Ч А Я П Р О Г Р А М М А**

**УЧЕБНОЙ ДИСЦИПЛИНЫ**

Операционные системы и реализация языков программирования

Operating Systems and Programming Languages Design and Implementation

**Язык(и) обучения**

русский

Трудоемкость в зачетных единицах: 3

Регистрационный номер рабочей программы: 003729

**Раздел 1. Характеристики учебных занятий**

**1.1. Цели и задачи учебных занятий**

Дисциплина «Операционные системы и реализация языков программирования» входит в вариативную часть учебного периода C6, формирующего подготовку бакалавра в области прикладной математики и информатики. Отдельные параметры курса могут варьироваться по степени сложности в зависимости от начальной подготовки обучающихся.  
Операционная система предоставляет программистам удобную абстракцию аппаратного обеспечения компьютера, а также управляет разделением ресурсов между пользователями. Темы данного курса затрагивают вопросы, влияющие на проектирование современных операционных систем. В течение многих лет операционные системы и их абстракции становились все более сложными по сравнению с обычными прикладными программами. Прежде чем перейти к изучению реализацию внутренних алгоритмов и структур данных, необходимо убедиться, что обучающиеся осознают, насколько широко в современном мире используются операционные системы. Поэтому данные темы связаны как с использованием операционных систем, так и с их проектированием и реализацией. Многие идеи, возникшие в процессе разработки операционных систем, нашли приложения в других областях информатики, например, в параллельном программировании. Изучение внутренней структуры операционных систем отражается на таких областях, как программирование с повышенными требованиями к надежности, проектирование и реализация алгоритмов, разработка современных устройств, создание виртуальных сред, кэширование документов в Internet, создание безопасных и защищенных систем, управление сетями и многих других.  
Реализация языков программирования наиболее эффективно рассматривается в контексте архитектур операционных систем, так как последние коренным образом влияют на специфику трансляторов языков программирования и порождаемого ими кода.  
Отдельные параметры курса могут варьироваться по степени сложности в зависимости от уровня подготовки обучающихся. Курс должен быть построен на принципах компетентностного, деятельностного подхода к операционным системам как к текстовым преобразователю, позволяющим получать эффективный обоснованного проведения различных расчетов, с применением высокопроизводительных компьютеров с общей памятью, что предполагает распределение содержания обучения по следующим видам деятельности: изучение основных концепций параллельного программирования, изучение средств распараллеливания, изучение элементов программирования, составление алгоритмов, проведение простейших вычислений, анализ ошибок, связанных с различными погрешностями, в том числе, с алгебраически допустимым изменением порядка вычислений.   
Основным методологическим принципом построения программы курса, является принцип поэтапного системного накопления знаний и формирования необходимых компетенций по модели: от простого и/или знакомого - к сложному и/или незнакомому, а основной методологической стратегией прохождения отдельных разделов программы является ступенчатость и цикличность, предусматривающие постепенный возврат к ранее усвоенному материалу на более высоком языковом и концептуальном уровне.  
Цель изучения дисциплины: ознакомление обучающихся с основными принципами и подходами в области реализации современных языков программирования.

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

Программа курса предназначена обучающимся 3 курса и рассчитана на обучающихся, изучавших основы информатики и программирования.  
Максимальная эффективность программы будет обеспечена при условии, что обучающийся:  
• владеет основами программирования;  
• понимает принципы применения языков программирования.

**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 естественнонаучные и общеинженерные знания, методы математического анализа и моделирования, теоретического и экспериментального исследования в профессиональной деятельности; | знание содержания дисциплины "Реализация языков программирования" и обладание достаточно полным представлением о возможностях применения разделов курса в различных прикладных областях науки и техники | 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 содержания дисциплины "Реализация языков программирования" и обладание достаточно полным представлением о возможностях применения разделов курса в различных прикладных областях науки и техники | 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; | знание содержания дисциплины "Реализация языков программирования" и обладание достаточно полным представлением о возможностях применения разделов курса в различных прикладных областях науки и техники | ОПК-5.1 Установка и настройка системного и прикладного ПО, необходимого для функционирования ИС в соответствии с трудовым заданием |
| 4 | Общепрофессиональные компетенции | ОПК-6 – способен разрабатывать алгоритмы и программы, пригодные для практического использования, применять основы информатики и программирования к проектированию, конструированию и тестированию программных продуктов; | знание содержания дисциплины "Реализация языков программирования" и обладание достаточно полным представлением о возможностях применения разделов курса в различных прикладных областях науки и техники | ОПК-6.1 Уметь проверять и отлаживать программный код |
| 5 | Общепрофессиональные компетенции | ОПК-7 – способен применять в практической деятельности основные концепции, принципы, теории и факты, связанные с информатикой | знание содержания дисциплины "Реализация языков программирования" и обладание достаточно полным представлением о возможностях применения разделов курса в различных прикладных областях науки и техники | ОПК-7.2 Оценка осуществимости функционирования и сопровождения программного средства |
| 6 | 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; | знание содержания дисциплины "Реализация языков программирования" и обладание достаточно полным представлением о возможностях применения разделов курса в различных прикладных областях науки и техники | ОПК-8.1 Оценка и выбор технологии доступа к данным |
| 7 | Профессиональные компетенции | ПКП-1 – способен проектировать программные системы; | знание содержания дисциплины "Реализация языков программирования" и обладание достаточно полным представлением о возможностях применения разделов курса в различных прикладных областях науки и техники | ПКП-1.1 Уметь разрабатывать технические спецификации на программные компоненты и их взаимодействие |
| 8 | Профессиональные компетенции | ПКП-2 – способен использовать основные модели информационных технологий и способы их применения для решения задач в предметных областях; | знание содержания дисциплины "Реализация языков программирования" и обладание достаточно полным представлением о возможностях применения разделов курса в различных прикладных областях науки и техники | ПКП-2.1 Уметь описывать алгоритмы компонентов, включая методы и схемы |
| 9 | Профессиональные компетенции | ПКП-3 – способен разрабатывать моделирующие алгоритмы и реализации их на базе языков и пакетов прикладных программ моделирования; | знание содержания дисциплины "Реализация языков программирования" и обладание достаточно полным представлением о возможностях применения разделов курса в различных прикладных областях науки и техники | ПКП-3.1 Разработка модели бизнес-процессов заказчика |
| 10 | Профессиональные компетенции | ПКП-4 – способен выбирать архитектуру и комплексирование современных компьютеров, систем, комплексов и сетей системного администрирования; | знание содержания дисциплины "Реализация языков программирования" и обладание достаточно полным представлением о возможностях применения разделов курса в различных прикладных областях науки и техники | ПКП-4.1 Оценка и выбор варианта архитектуры программного средства |
| 11 | Профессиональные компетенции | ПКП-5 – способен использовать современные системные программные средства: операционные системы, операционные и сетевые оболочки, сервисные программы; | знание содержания дисциплины "Реализация языков программирования" и обладание достаточно полным представлением о возможностях применения разделов курса в различных прикладных областях науки и техники | ПКП-5.1 Уметь выбирать технологии и средства разработки программного обеспечения |
| 12 | Профессиональные компетенции | ПКП-6 – способен формировать суждения о проблемах современной информатики, ее категорий и связей с другими научными дисциплинами; | знание содержания дисциплины "Реализация языков программирования" и обладание достаточно полным представлением о возможностях применения разделов курса в различных прикладных областях науки и техники | ПКП-6.1 Уметь разрабатывать процедуры проверки работоспособности и измерения характеристик программного обеспечения |
| 13 | Профессиональные компетенции | ПКП-7 – способен систематизировать и применять знания о содержании основных этапов и тенденций развития программирования, математического обеспечения и информационных технологий; | знание содержания дисциплины "Реализация языков программирования" и обладание достаточно полным представлением о возможностях применения разделов курса в различных прикладных областях науки и техники | ПКП-7.1 Умеет проводить технические и управленческие ревизии создаваемого программного средства |
| 14 | Универсальные компетенции | УК-1 – способен осуществлять поиск, критический анализ и синтез информации, применять системный подход для решения поставленных задач; | развитие способности дальнейшего освоения методов разработки и реализации языков программирования | УК 1.4. Оценивает достоинства, недостатки и последствия вариантов решения поставленных задач; |
| 15 | 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; | развитие способности дальнейшего освоения методов разработки и реализации языков программирования | УК-2.2. Предлагает способы решения поставленных задач; |
| 16 | Универсальные компетенции | УКБ-1 – способен участвовать в разработке и реализации проектов, в т.ч. предпринимательских; | развитие способности дальнейшего освоения методов разработки и реализации языков программирования | УКБ-1.5. Выполняет задачи в зоне своей ответственности в соответствии с запланированными результатами и точками контроля; |
| 17 | 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 | развитие способности дальнейшего освоения методов разработки и реализации языков программирования | УКБ-3.2. Точно определяет тип и форму необходимой информации. |

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

Аудиторная учебная работа: лекционные занятия в объеме 4 часов в неделю.

**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 6 | 60 |  | 2 |  |  |  |  |  | 2 |  |  |  | 8 |  | 36 |  | 10 | 3 |
|  | 2-25 |  | 2-25 |  |  |  |  |  | 2-25 |  |  |  | 1-1 |  | 1-1 |  |  |  |
| ИТОГО | 60 |  | 2 |  |  |  |  |  | 2 |  |  |  | 8 |  | 36 |  |  | 3 |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 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 6 |  |  | экзамен, устно, традиционная форма | по графику промежуточной аттестации |  |  |

**2.2. Структура и содержание учебных занятий**

**Основной курс Основная траектория Очная форма обучения**

Период обучения (модуль): Семестр 6

|  |  |  |  |
| --- | --- | --- | --- |
| № п/п | Наименование темы (раздела, части) | Вид учебных занятий | Количество часов |
| М1 | Основные понятия операционных систем | Лекционные занятия | 4 |
| М2 | Процессы и потоки | Лекционные занятия | 4 |
| М3 | Управление памятью | Лекционные занятия | 4 |
| М4 | Ввод-вывод | Лекционные занятия | 6 |
| М5 | Безопасность | Лекционные занятия | 8 |
| М6 | Введение в распределенные вычисления | Лекционные занятия | 6 |
| М7 | Языки сценариев | Лекционные занятия | 8 |
| М8 | Конкретные случаи | Лекционные занятия | 6 |
| М9 | Оценка производительности | Лекционные занятия | 4 |
| М10 | Реализация языков программирования | Лекционные занятия | 10 |

М1 Основные понятия операционных систем

1.1. Роль и задачи операционных систем

1.2. История развития операционных систем

1.3. Функциональность типичной операционной системы

1.4. Связь поколений компьютеров и поколений операционных систем

1.5. Вопросы, влияющие на устройство ОС

1.6. Методы структуризации

1.7. Абстракции, процессы и ресурсы

1.8. Создание программных интерфейсов приложений

1.9. Организация устройств

1.10. Прерывания

1.11 Переключения между режимами работы пользователя / супервизора

Модуль 2 «Процессы и потоки»

2.1. Состояния и диаграммы состояний

2.2. Структуры (готовности, блоки управления процессом, …)

2.3. Диспетчеризация и переключение контекстов

2.4. Роль прерываний

2.5. Параллельное исполнение: преимущества и недостатки

2.6. Проблемы " исключения " и некоторые решения

2.7. Тупики: причины, условия, способы предотвращения

2.8. Модели и механизмы (семафоры, мониторы, условные переменные, рандеву)

2.9. Проблема «читатель/писатель» и синхронизация

2.10. Вопросы, связанные с многопроцессорностью (спин-блокировка, реентерабельность)

2.11. Тупики: причины, условия, способы предотвращения

2.12. Проблема «читатель/писатель» и синхронизация

2.13. Вопросы, связанные с многопроцессорностью (спин-блокировка, реентерабельность)

2.14. Вытесняющее и невытесняющее планирование

2.15. Политики планирования

2.16. Процессы и нити

2.17. Особенности систем реального времени

М3 Управление памятью

3.1. Обзор физической памяти и управляющей аппаратуры

3.2. Оверлеи, подкачка и разделы

3.3. Алгоритмы динамического распределения памяти

3.4. Страничная организация памяти и сегментация

3.5. Стратегии подкачки и выгрузки страниц

3.6. Рабочие множества и пробуксовка

3.7. Кэширование

М4 Ввод-вывод

4.1. Характеристики последовательных и параллельных устройств

4.2. Абстрактные понятия различий устройств

4.3. Стратегии буферизации

4.4. Прямой доступ к памяти

4.5. Восстановление после сбоев

4.6. Основные понятия управления файлами (данные, метаданные, операции, организация, буферизация, последовательные файлы, файлы с непоследовательным размещением)

4.7. Содержание и структура каталогов

4.8. Методы работы файловой системы (сегментирование дисковой памяти, монтирование и демонтирование, виртуальные файловые системы)

4.9. Файлы, отображаемые в память

4.10. Файловые системы специального назначения

4.11. Именование, поиск и доступ

4.12. Стратегии резервного копирования

4.13. Сжатие файлов

М5 Безопасность

5.1. Обзор системы безопасности ОС

5.2. Разделение политики безопасно и механизма ее реализации

5.3. Методы и устройства обеспечения безопасности

5.4. Защита, доступ и аутентификация

5.5. Модели защиты

5.6. Защита памяти

5.7. Шифрование

5.8. Управление восстановлением

М6 Введение в распределенные вычисления

6.1. Основные задачи, стоящие перед системой распределенных вычислений

6.2. Сетевое аппаратное обеспечение

6.3. Концепции программных решений

6.4. Архитектура клиент/сервер (определение, классы приложений, трехуровневая архитектура)

6.5. Протоколы

6.6. Распределенная передача сообщений

6.7. Синхронизация часов

М7 Языки сценариев

7.1. Языки программирования систем и языки управления сценариями

7.2. Типы языков сценариев

7.3. Современные языки сценариев

7.4. Влияние языков сценариев на программирование

М8 Конкретные случаи

8.1. История операционных систем Microsoft

8.3. Структура операционной системы Windows 2000

8.4. Выполнение вызовов API в Windows 2000

8.5. Процессы и потоки в Windows 2000

8.6. Межпроцессное взаимодействие (каналы и именованные каналы, почтовые ящики, сокеты, вызов удаленной процедуры, совместно используемые файлы) в Windows 2000

8.7. Планирование в Windows 2000

8.8. Управление памятью в Windows 2000

8.9. Файловые системы в Windows 2000

8.10. Структура операционной системы Windows XP

8.11. Управление виртуальной памятью в Windows XP

8.12. Процессы, потоки и волокна в Windows XP

8.13. Организация сетевой поддержки в Windows XP

8.14. Управление файлами в Windows XP

М9 Оценка производительности

9.1. Основные подходы к оценке производительности операционной системы

9.2. Оценка объема работ

М10 Реализация языков программирования

* 1. . Языки программирования; синтаксис, семантика, прагматика.

10.2. Понятие смешанного вычислителя. Интерпретаторы, проекции Футамуры-Ершова-Турчина.

10.3. Общий план устройства компилятора. Просмотры, промежуточные представления.

10.4. Архитектура ЭВМ с точки зрения компилятора.

10.5. Представление программ на уровне ассемблера. Формат объектного кода, редактор связей.

10.6. Принципы устройства отладчиков и профилировщиков.

10.7. Реализация сложно устроенных объектов (массивов, структур).

10.8. Процедуры и функции. Протокол организации вызова, дисциплина передачи параметров и возврата результатов.

10.9. Реализация вложенных функций.

10.10. Нелокальные переходы, сравнение различных подходов и свойств их реализаций.

10.11. Объектно-ориентированные конструкции; устройство объектов, реализация позднего связывания.

10.12. Параметрический полиморфизм и принципы его реализации.

10.13. Классы типов Haskell. Сравнение с объектно-ориентированными конструкциями.

**Раздел 3. Обеспечение учебных занятий**

**3.1. Методическое обеспечение**

**3.1.1 Методические указания по освоению дисциплины**

Для освоения дисциплины обучающиеся должны посещать лекционные занятия. На них преподаватель рассказывает материал курса согласно содержанию в разбивке по разделам по п.2.2.

При подготовке к лекциям необходимо учитывать, что изложение материала должно быть от простого к сложному, от известного к неизвестному. Рекомендуется применять активные формы проведения занятий: проблемное изложение, дискуссии, диалог с целью активизации деятельности обучающихся. При проведении занятий необходимо использовать презентации.

При изложении материала важно помнить, что почти половина информации на семинаре передается через интонацию. Учитывать тот факт, что первый кризис внимания обучающихся наступает на 15-20-й минутах, второй – на 30-35-й минутах.

**3.1.2 Методическое обеспечение самостоятельной работы**

При самостоятельном изучении теоретического материала целесообразно использовать рекомендованную основную и дополнительную литературу. По согласованию с преподавателем обучающийся может осваивать теоретическую часть курса по одному из онлайн-курсов, представленных на образовательных платформах в п. 3.4.3. При этом преподаватель обязан сообщить обучающимся, какие разделы выбранного им онлайн-курса недостаточно полно раскрывают ту или иную тему курса и порекомендовать дополнительные источники по данной теме. Преподаватель вправе отказать обучающемуся в самостоятельном освоении теоретической части дисциплины по онлайн-курсу в случае нахождения в нем существенных расхождений с содержанием курса в разделе 2 и п. 3.1.1.

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

Экзамен проводится в устной форме по билетам. Список экзаменационных вопросов предоставляется обучающимся не позднее, чем за две недели до экзамена. Пример списка содержится в разделе 3.1.4.

Использование конспектов и учебников, а также электронных устройств хранения, обработки или передачи информации при подготовке и ответе на вопросы экзамена запрещено.

Билет содержит три теоретических вопроса. На подготовку отводится не менее 1 академического часа. По окончании подготовки к ответу обучающийся устно излагает содержание экзаменационного вопроса экзаменатору. После ответа на вопрос билета, преподаватель вправе задать уточняющие вопросы по услышанному. Затем преподаватель задает дополнительные вопросы по любой теме из списка вопросов, вынесенных на экзамен. В качестве уточняющих используются вопросы, не требующие длительного ответа, в том числе основные определения и понятия. Дополнительные вопросы должны строиться таким образом, чтобы с их помощью обучающийся мог продемонстрировать своё понимание или непонимание материала, а также способности к аргументированному обсуждению. Рекомендуется задавать 3-4 дополнительных вопроса на билет и не более 2-3 уточняющих вопросов. Преподаватель вправе увеличить количество вопросов в случае, если у него не возникает понимание, освоил обучающийся материал учебного курса или нет.

В любой момент обучающийся имеет право отказаться от ответа с выставлением оценки в 0 баллов.

За экзамен обучающийся может получить максимум 60 баллов. Ответ на каждый вопрос экзаменационного билета вместе с ответами на уточняющие вопросы по нему оценивается по шкале от 0 баллов (нет ответа) до 10 баллов (экзаменуемым даны полные исчерпывающие ответы по всем вопросам билета, обучающийся свободно ориентируется в материале), таким образом, за вопросы билета обучающийся может получить до 30 баллов. Оставшийся бюджет из 30 баллов делится поровну по дополнительным вопросам. Преподаватель вправе снижать баллы за неточности и ошибки в зависимости от их грубости.

По желанию преподавателя на экзамен допустимо приглашать 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 экзамен ставится на основании голосования простого большинства. В спорных ситуациях преподаватель, ведущий дисциплину, имеет право принятия окончательного решения.

Перевод из баллов, полученных за экзамен в итоговый процент освоения курса делается по формуле

где I – итоговый процент освоения курса, n – заработанное обучающимся число баллов. Перевод в оценки делается по стандартной методике согласно приказу №7293/1 от 20.07.2018:

|  |  |  |
| --- | --- | --- |
| Итоговый процент освоения курса, % | Оценка СПбГУ при проведении экзамена | Оценка ECTS |
| 0-49 | Неудовлетворительно | F |
| 50-60 | Удовлетворительно | E |
| 61-69 | Удовлетворительно | D |
| 70-79 | Хорошо | C |
| 80-89 | Хорошо | B |
| 90-100 | Отлично | A |

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

В качестве примера вопросов на экзамен по дисциплине допустимо использовать основные вопросы, которые обсуждались на лекции в разбивке в п. 2.2:

1. Роль и задачи операционных систем

2. История развития операционных систем

3. Функциональность типичной операционной системы

4. Связь поколений компьютеров и поколений операционных систем

5. Вопросы, влияющие на устройство ОС

6. Методы структуризации

7. Абстракции, процессы и ресурсы

8. Создание программных интерфейсов приложений

9. Организация устройств

10. Прерывания

11 Переключения между режимами работы пользователя / супервизора

12. Состояния и диаграммы состояний

13. Структуры (готовности, блоки управления процессом, …)

14. Диспетчеризация и переключение контекстов

15. Роль прерываний

16. Параллельное исполнение: преимущества и недостатки

17. Проблемы " исключения " и некоторые решения

18. Тупики: причины, условия, способы предотвращения

19. Модели и механизмы (семафоры, мониторы, условные переменные, рандеву)

20. Проблема «читатель/писатель» и синхронизация

21. Вопросы, связанные с многопроцессорностью (спин-блокировка, реентерабельность)

22. Тупики: причины, условия, способы предотвращения

23. Проблема «читатель/писатель» и синхронизация

24. Вопросы, связанные с многопроцессорностью (спин-блокировка, реентерабельность)

25. Вытесняющее и невытесняющее планирование

26. Политики планирования

27. Процессы и нити

28. Особенности систем реального времени

29. Обзор физической памяти и управляющей аппаратуры

30. Оверлеи, подкачка и разделы

31. Алгоритмы динамического распределения памяти

32. Страничная организация памяти и сегментация

33. Стратегии подкачки и выгрузки страниц

34. Рабочие множества и пробуксовка

35. Кэширование

36. Характеристики последовательных и параллельных устройств

37. Абстрактные понятия различий устройств

38. Стратегии буферизации

39. Прямой доступ к памяти

40. Восстановление после сбоев

41. Основные понятия управления файлами (данные, метаданные, операции, организация, буферизация, последовательные файлы, файлы с непоследовательным размещением)

42. Содержание и структура каталогов

43. Методы работы файловой системы (сегментирование дисковой памяти, монтирование и демонтирование, виртуальные файловые системы)

44. Файлы, отображаемые в память

45. Файловые системы специального назначения

46. Именование, поиск и доступ

47. Стратегии резервного копирования

48. Сжатие файлов

49. Обзор системы безопасности ОС

50. Разделение политики безопасно и механизма ее реализации

51. Методы и устройства обеспечения безопасности

52. Защита, доступ и аутентификация

53. Модели защиты

54. Защита памяти

55. Шифрование

56. Управление восстановлением

57. Основные задачи, стоящие перед системой распределенных вычислений

58. Сетевое аппаратное обеспечение

59. Концепции программных решений

60. Архитектура клиент/сервер (определение, классы приложений, трехуровневая архитектура)

61. Протоколы

62. Распределенная передача сообщений

63. Синхронизация часов

64. Языки программирования систем и языки управления сценариями

65. Типы языков сценариев

66. Современные языки сценариев

67. Влияние языков сценариев на программирование

68. История операционных систем Microsoft

69. Структура операционной системы Windows 2000

70. Выполнение вызовов API в Windows 2000

71. Процессы и потоки в Windows 2000

72. Межпроцессное взаимодействие (каналы и именованные каналы, почтовые ящики, сокеты, вызов удаленной процедуры, совместно используемые файлы) в Windows 2000

73. Планирование в Windows 2000

74. Управление памятью в Windows 2000

75. Файловые системы в Windows 2000

76. Структура операционной системы Windows XP

77. Управление виртуальной памятью в Windows XP

78. Процессы, потоки и волокна в Windows XP

79. Организация сетевой поддержки в Windows XP

80. Управление файлами в Windows XP

81. Основные подходы к оценке производительности операционной системы

82. Оценка объема работ

83. Языки программирования; синтаксис, семантика, прагматика.

84. Понятие смешанного вычислителя. Интерпретаторы, проекции Футамуры-Ершова-Турчина.

85. Общий план устройства компилятора. Просмотры, промежуточные представления.

86. Архитектура ЭВМ с точки зрения компилятора.

87. Представление программ на уровне ассемблера. Формат объектного кода, редактор связей.

88. Принципы устройства отладчиков и профилировщиков.

89. Реализация сложно устроенных объектов (массивов, структур).

90. Процедуры и функции. Протокол организации вызова, дисциплина передачи параметров и возврата результатов.

91. Реализация вложенных функций.

92. Нелокальные переходы, сравнение различных подходов и свойств их реализаций.

93. Объектно-ориентированные конструкции; устройство объектов, реализация позднего связывания.

94. Параметрический полиморфизм и принципы его реализации.

95. Классы типов Haskell. Сравнение с объектно-ориентированными конструкциями.

|  |  |  |
| --- | --- | --- |
| № | 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 три вопроса на экзамене и ответы на дополнительные теоретические вопросы независимо оцениваются по шкале от 0 (нет ответа) до 10 (очень хорошо), далее оценка усредняется. Результат переводится в диапазон от 0 до 100 |
| 2 | ОПК-2.2 Уметь проводить формализацию и алгоритмизацию поставленных задач | обучающегося просят объяснить, как вопрос из билета реализуется в том или ином программном продукте, языке программирования и/или операционной системе. Ответ оценивается по шкале от 0 (нет ответа) до 100 (очень хорошо). |
| 3 | ОПК-5.1 Установка и настройка системного и прикладного ПО, необходимого для функционирования ИС в соответствии с трудовым заданием | обучающегося просят установить программный продукт, среду разработки на языке программирования и/или операционную систему из предыдущего задания. Ответ оценивается по шкале от 0 (не сделано) до 100 (очень хорошо) |
| 4 | ОПК-6.1 Уметь проверять и отлаживать программный код | обучающегося просят объяснить, как вопрос из билета реализуется в том или ином программном продукте, языке программирования и/или операционной системе. Ответ оценивается по шкале от 0 (нет ответа) до 100 (очень хорошо). |
| 5 | ОПК-7.2 Оценка осуществимости функционирования и сопровождения программного средства | обучающегося просят объяснить, как вопрос из билета реализуется в том или ином программном продукте, языке программирования и/или операционной системе. Ответ оценивается по шкале от 0 (нет ответа) до 100 (очень хорошо). |
| 6 | ОПК-8.1 Оценка и выбор технологии доступа к данным | обучающегося просят объяснить, как вопрос из билета реализуется в том или ином программном продукте, языке программирования и/или операционной системе. Ответ оценивается по шкале от 0 (нет ответа) до 100 (очень хорошо). |
| 7 | ПКП-1.1 Уметь разрабатывать технические спецификации на программные компоненты и их взаимодействие | обучающегося просят объяснить, как бы он реализовывал функциональность из вопроса из билета, если бы ему пришлось разрабатывать программный продукт, среду разработки или операционную систему с нуля. Ответ оценивается по шкале от 0 (нет ответа) до 100 (очень хорошо) |
| 8 | ПКП-2.1 Уметь описывать алгоритмы компонентов, включая методы и схемы | обучающемуся предлагается запрограммировать алгоритм из экзаменационного билета в установленной среде разработки.Результат оценивается по шкале от 0 (не сделано) до 100 (очень хорошо) |
| 9 | ПКП-3.1 Разработка модели бизнес-процессов заказчика | обучающегося просят объяснить, как бы он реализовывал функциональность из вопроса из билета, если бы ему пришлось разрабатывать программный продукт, среду разработки или операционную систему с нуля. Ответ оценивается по шкале от 0 (нет ответа) до 100 (очень хорошо) |
| 10 | ПКП-4.1 Оценка и выбор варианта архитектуры программного средства | обучающегося просят объяснить, как бы он реализовывал функциональность из вопроса из билета, если бы ему пришлось разрабатывать программный продукт, среду разработки или операционную систему с нуля. Ответ оценивается по шкале от 0 (нет ответа) до 100 (очень хорошо) |
| 11 | ПКП-5.1 Уметь выбирать технологии и средства разработки программного обеспечения | обучающемуся предлагается установить среду разработки для реализации одного из алгоритмов в экзаменационном билете.Результат оценивается по шкале от 0 (не сделано) до 100 (очень хорошо) |
| 12 | ПКП-6.1 Уметь разрабатывать процедуры проверки работоспособности и измерения характеристик программного обеспечения | обучающегося просят объяснить, как бы он реализовывал функциональность из вопроса из билета, если бы ему пришлось разрабатывать программный продукт, среду разработки или операционную систему с нуля. Ответ оценивается по шкале от 0 (нет ответа) до 100 (очень хорошо) |
| 13 | ПКП-7.1 Умеет проводить технические и управленческие ревизии создаваемого программного средства | ответы на вопросы экзаменационного билета и ответы на дополнительные вопросы оцениваются по шкале от 0 (нет ответа) до 10 (очень хорошо), далее оценка усредняется. Результат переводится в диапазон от 0 до 100 |
| 14 | УК 1.4. Оценивает достоинства, недостатки и последствия вариантов решения поставленных задач; | обучающегося просят объяснить, как бы он реализовывал функциональность из вопроса из билета, если бы ему пришлось разрабатывать программный продукт, среду разработки или операционную систему с нуля. Ответ оценивается по шкале от 0 (нет ответа) до 100 (очень хорошо) |
| 15 | УК-2.2. Предлагает способы решения поставленных задач; | ответы на три вопроса на экзамене и ответы на дополнительные теоретические вопросы независимо оцениваются по шкале от 0 (нет ответа) до 10 (очень хорошо), далее оценка усредняется. Результат переводится в диапазон от 0 до 100 |
| 16 | УКБ-1.5. Выполняет задачи в зоне своей ответственности в соответствии с запланированными результатами и точками контроля; | обучающемуся предлагается запрограммировать алгоритм из экзаменационного билета в установленной среде разработки.Результат оценивается по шкале от 0 (не сделано) до 100 (очень хорошо) |
| 17 | УКБ-3.2. Точно определяет тип и форму необходимой информации. | обучающегося просят объяснить, как бы он реализовывал функциональность из вопроса из билета, если бы ему пришлось разрабатывать программный продукт, среду разработки или операционную систему с нуля. Ответ оценивается по шкале от 0 (нет ответа) до 100 (очень хорошо) |

**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 могут быть допущены преподаватели, имеющие диплом о высшем техническом образовании по направлениям «Программная инженерия», «Математическое обеспечение и администрирование информационных систем» и другим смежным направлениям, связанным с информационными технологиями.

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

Специальных требований нет.

**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.

**3.4. Информационное обеспечение**

**3.4.1. Список литературы**

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

2. [www.intuit.ru](http://www.intuit.ru) – Интернет-университет информационных технологий.

3. [www.edx.org](http://www.edx.org) – платформа онлайн-образования.

4. [www.coursera.org](http://www.coursera.org) – платформа онлайн-образования.

**3.4.2. Иные информационные источники**

• Сайт Научной библиотеки им. М. Горького СПбГУ: <http://www.library.spbu.ru/>

• 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>

• Перечень электронных ресурсов, находящихся в доступе СПбГУ: <http://cufts.library.spbu.ru/CRDB/SPBGU/>

• 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**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ФИО | Ученая  степень | Ученое звание | Должность | Контакты |
| Булычев Дмитрий Юрьевич | к.-ф.-м.-н. | доцент | Доцент кафедры системного программирования | dboulytchev@gmail.com |
| Сартасов  Станислав Юрьевич |  |  | ст.преподаватель кафедры системного программирования | s.sartasov@spbu.ru |