

Документ подписан простой электронной подписью

Информация о владельце:

ФИО: Фурсов Владимир Александрович

Должность: И.о. директора Пятигорского института (филиал) Северо-Кавказского федерального университета

Дата подписания: 2026.02.11 10:44

Уникальный программный ключ:

1c378726a41fd0143ae5bccc8ba81860b00daa62

**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ**
**Федеральное государственное автономное образовательное учреждение
высшего образования**
«СЕВЕРО-КАВКАЗСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»
Пятигорский институт (филиал) СКФУ
Колледж Пятигорского института (филиал) СКФУ

**МЕТОДИЧЕСКИЕ УКАЗАНИЯ ДЛЯ САМОСТОЯТЕЛЬНОЙ
РАБОТЫ**

МДК.02.01 Разработка программных модулей

Специальность СПО

09.02.11 Разработка и управление программным обеспечением

Пятигорск 2026

Методические указания для самостоятельной работы по дисциплине «Разработка программных модулей» составлены в соответствии с требованиями ФГОС СПО к подготовке выпуска для получения квалификации специалист по информационным системам. Предназначены для студентов, обучающихся по специальности 09.02.11 Разработка и управление программным обеспечением.

СОДЕРЖАНИЕ:

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА	4
ПЛАН-ГРАФИК ВЫПОЛНЕНИЯ СРС	6
МЕТОДИЧЕСКИЕ РЕКОМЕНДАЦИИ К СРС	7

Пояснительная записка

Методические указания предназначены для студентов групп СПО специальности 09.02.11 Разработка и управление программным обеспечением.

Самостоятельная работа студентов проводится с целью:

- систематизации и закрепления полученных теоретических знаний и практических умений обучающихся;
- углубления и расширения теоретических знаний;
- формирования умений использовать нормативную, правовую, справочную документацию и специальную литературу;
- развития познавательных способностей и активности обучающихся, творческой инициативы, ответственности и организованности;
- формирования самостоятельности мышления, способностей к саморазвитию, самосовершенствованию и самореализации;
- развития исследовательских умений.

В результате освоения учебной дисциплины обучающийся должен **уметь:**

- Распознавать задачу и/или проблему в профессиональном и/или социальном контексте; анализировать и выделять её составные части;
- Определять задачи для поиска информации, необходимые источники, планировать процесс поиска, структурировать и оформлять результаты;
- Разрабатывать программные модули в соответствии с техническим заданием;
- Применять базовые алгоритмические конструкции и структуры данных;
- Использовать системы контроля версий для управления кодом;
- Реализовывать объектно-ориентированные концепции в коде;
- Обрабатывать исключительные ситуации и создавать пользовательские исключения;
- Применять паттерны проектирования для решения типовых задач;
- Выполнять чтение и запись данных в файлы;
- Создавать многопоточные и сетевые приложения;
- Разрабатывать пользовательские интерфейсы и подключаться к базам данных;
- Оптимизировать и профилировать код, создавать документацию.

знать:

- Актуальный профессиональный и социальный контекст, основные источники информации и ресурсы для решения задач;
- Номенклатуру информационных источников, приемы структурирования информации, современные средства информатизации;
- Основные этапы жизненного цикла программного обеспечения и модели разработки;

- Принципы алгоритмизации, структуры данных и базовые алгоритмические конструкции;
- Синтаксис и семантику языка программирования высокого уровня;
- Принципы работы систем контроля версий (Git);
- Основные концепции объектно-ориентированного программирования;
- Механизмы обработки исключительных ситуаций;
- Паттерны проектирования и их классификацию;
- Основы работы с файловой системой и потоками ввода-вывода;
- Принципы многопоточного и сетевого программирования;
- Основы разработки пользовательских интерфейсов и работы с базами данных;
- Методы оптимизации, профилирования и документирования кода;
- Отраслевую нормативную документацию, современный отечественный и зарубежный опыт в области разработки программного обеспечения.

План-график выполнения СРС

№	Наименование разделов и тем дисциплины, их краткое содержание; вид самостоятельной работы	Форма контроля	Зачетные единицы (часы)
V семестр			
1.	Тема 1.7. Продвинутое концепции ООП Сравнительный анализ абстрактных классов и интерфейсов на примере различных языков программирования (Java, C#, Python)	Собеседование	2
2.	Тема 1.8. Паттерны проектирования Изучение порождающих (Singleton, Factory Method), структурных (Adapter) и поведенческих (Observer, Strategy) паттернов. Сравнительный анализ и примеры реализации	Собеседование	4
3.	Тема 1.9. Работа с файловой системой и потоками ввода-вывода Исследование различных режимов работы с файлами (чтение, запись, добавление). Сравнение производительности текстовых и бинарных файлов	Собеседование	2
4.	Тема 1.10. Многопоточное программирование Анализ проблем синхронизации (race condition, deadlock). Изучение механизмов синхронизации: mutex, semaphore, condition variable	Собеседование	2
5.	Тема 1.11. Основы сетевого программирования Изучение модели OSI и стека TCP/IP. Анализ принципов работы сокетов и протокола HTTP	Собеседование	2
Итого за V семестр			12
VI семестр			
6.	Тема 1.12. Разработка пользовательских интерфейсов Сравнительный анализ фреймворков для разработки GUI (Qt, JavaFX, Tkinter, Windows Forms). Принципы событийного программирования	Собеседование	4
7.	Тема 1.13. Работа с базами данных Изучение основ реляционных БД и языка SQL. Сравнение производительности чистого SQL и ORM (SQLAlchemy, Hibernate, Entity Framework) при выполнении CRUD-операций	Собеседование	8

8.	Тема 1.14. Разработка веб-сервисов и API Анализ архитектуры REST, принципов проектирования API. Изучение инструментов документирования API (Swagger/OpenAPI)	Собеседование	8
9.	Тема 1.15. Оптимизация и профилирование кода Исследование методов оптимизации кода и инструментов профилирования (cProfile, VisualVM, dotTrace). Анализ типичных "узких мест" производительности	Собеседование	4
10.	Тема 1.16. Документирование программных модулей Изучение стандартов оформления документации (Javadoc, Doxygen, Sphinx). Сравнение инструментов автоматической генерации документации	Собеседование	4
Итого за VI семестр			28
Всего за V-VI семестры			40

Методические рекомендации к СРС

1. Методические рекомендации по проведению собеседования.

Собеседование - наиболее распространенный метод контроля знаний учащихся, вариант текущей проверки, процессе которого преподаватель получает широкие возможности для изучения индивидуальных возможностей усвоения учащимися учебного материала.

При подготовке к собеседованию студент должен:

1. Предварительно повторить теоретический материал темы (тем) по которой проводится устный опрос.
2. Ознакомиться с заданием, уяснить его фабулу и поставленные вопросы.
3. Продумать логику и последовательность изложения материала. Ответы на поставленные вопросы должны быть аргументированными.

1. Критерии оценивания компетенций

1. Оценка «ОТЛИЧНО» выставляется студенту, глубоко и прочно усвоившему программный, в том числе лекционный материал, последовательно, четко и самостоятельно (без наводящих вопросов) отвечающему на вопрос.

2. Оценка «ХОРОШО» выставляется студенту, твердо знающему программный, в том числе лекционный материал, грамотно и по существу отвечающему на вопрос и не допускающему при этом существенных неточностей (неточностей, которые не могут быть исправлены наводящими вопросами или не имеют важного практического значения). То же относится к освещению практически важных вопросов

3. Оценка «УДОВЛЕТВОРИТЕЛЬНО» выставляется студенту, который обнаруживает знание основного материала, но не знает его деталей, допускает неточности, недостаточно правильные формулировки, излагает материал с нарушением последовательности, отвечает на практически важные вопросы с помощью или поправками преподавателя.

4. Оценка «НЕУДОВЛЕТВОРИТЕЛЬНО» выставляется студенту, который не знает значительной части программного, в том числе лекционного материала.

Список рекомендуемой литературы

3.2.1. Основные печатные издания

1. Седжвик, Р. Алгоритмы на Java / Р. Седжвик, К. Уэйн. — Москва : Вильямс, 2022. — 848 с. — ISBN 978-5-907515-34-1.
2. Шилдт, Г. Java. Полное руководство / Г. Шилдт. — Москва : Диалектика, 2023. — 1488 с. — ISBN 978-5-907515-36-5.
3. Лутц, М. Изучаем Python / М. Лутц. — Санкт-Петербург : Питер, 2022. — 1360 с. — ISBN 978-5-4461-1919-8.
4. Макконнелл, С. Совершенный код / С. Макконнелл. — Санкт-Петербург : Питер, 2021. — 896 с. — ISBN 978-5-4461-1486-5.
5. Гамма, Э. Приемы объектно-ориентированного проектирования. Паттерны проектирования / Э. Гамма, Р. Хелм, Р. Джонсон, Дж. Влиссидес. — Санкт-Петербург : Питер, 2022. — 544 с. — ISBN 978-5-4461-1875-7.
6. Чакон, С. Pro Git / С. Чакон, Б. Штрауб. — Москва : ДМК Пресс, 2022. — 560 с. — ISBN 978-5-93700-215-4.
7. Стивенс, Р. UNIX. Профессиональное программирование / Р. Стивенс, С. Раго. — Санкт-Петербург : Питер, 2021. — 928 с. — ISBN 978-5-4461-1618-0.
8. Мартин, Р. Чистый код. Создание, анализ и рефакторинг / Р. Мартин. — Санкт-Петербург : Питер, 2022. — 464 с. — ISBN 978-5-4461-1567-1.
9. Фаулер, М. Рефакторинг. Улучшение существующего кода / М. Фаулер. — Санкт-Петербург : Питер, 2021. — 448 с. — ISBN 978-5-4461-1659-3.
10. Кузнецов, С. Д. Базы данных : учебник для среднего профессионального образования / С. Д. Кузнецов. — Москва : Юрайт, 2023. — 420 с. — ISBN 978-5-534-16878-3.