

**УТВЕРЖДЕНА**

Решением Ученого совета  
АНО ВО «Центральный университет»  
«07» марта 2024 г.  
Протокол №1

**Рабочая программа дисциплины (модуля)  
«Инструменты разработчика»**

**Направление подготовки:** 02.04.01 Математика и компьютерные науки

**Направленность (профиль) подготовки:** Продуктовый менеджмент

**Квалификация (степень) выпускника:** магистр

**Форма обучения:** очная

**Срок освоения программы:** 2 года

**Год набора:** 2024

**Москва  
2024**

## Содержание

|   |    |
|---|----|
| 1. Краткая характеристика дисциплины (модуля) ..... | 3  |
| 2. Перечень планируемых результатов обучения.....   | 4  |
| 3. Тематический план.....                           | 6  |
| 4. Содержание дисциплины (модуля).....              | 7  |
| 5. Учебно-методическое обеспечение .....            | 8  |
| 6. Материально-техническое обеспечение .....        | 8  |
| 7. Методические и оценочные материалы .....         | 10 |

## 1. Краткая характеристика дисциплины (модуля)

Рабочая программа дисциплины (модуля) «Инструменты разработчика» составлена в соответствии с федеральным государственным образовательным стандартом высшего образования – магистратура по специальности 02.04.01 Математика и компьютерные науки, профиль Продуктовый менеджмент, утвержденный приказом Министерства науки и высшего образования Российской Федерации № 810 от 23.08.2017 года.

Изучение дисциплины (модуля) «Инструменты разработчика» помогает студентам овладеть современными инструментами разработчика, позволяет повысить качество кода, ускорить процесс разработки и обеспечить удобное сотрудничество в команде. Это способствует созданию надежных и масштабируемых программных продуктов, соответствующих требованиям рынка.

### **Место дисциплины (модуля) в структуре образовательной программы**

Настоящая дисциплина (модуль) включена в учебный план по программе подготовки магистратуры по направлению 02.04.01 Математика и компьютерные науки, профиль Продуктовый менеджмент и входит в вариативную часть Блока 1, формируемую участниками образовательных отношений.

Дисциплина (модуль) изучается на 2 курсе в 3 семестре.

**Цель изучения дисциплины (модуля):** формирование у студентов понимание основных программных средств и технологий, используемых для эффективной разработки, тестирования и сопровождения программного обеспечения.

### **Задачи изучения дисциплины (модуля):**

- формирование знаний принципов работы протокола SSH;
- формирование навыка подключаться к удаленному серверу и копировать файлы на удаленный сервер;
- формирование знаний о различиях между системами сборки, основанными на целях и основанных на конвейерах сборки;
- формирование умения работать с удаленными репозиториями через интерфейсы GitHub и GitLab;
- формирование умения использовать инструменты разработчика для браузеров для отладки и мониторинга веб-приложений;
- формирование умения организовать рабочий процесс в команде при помощи систем контроля версий;
- формирование умения настроить запуск долгоживущего сервиса на Java на удаленном сервере.

## 2. Перечень планируемых результатов обучения

Компетенции, формируемые в результате освоения дисциплины (модуля) при проведении учебных занятий в форме контактной работы обучающихся с педагогическими работниками Университета и в форме самостоятельной работы обучающихся:

| Компетенция | Содержание компетенции  | Индикатор компетенции | Перечень планируемых результатов обучения по дисциплине (модулю)   |
|-------------|---|-----------------------|--|
| УК-6.       | Способен определять и реализовывать приоритеты собственной деятельности и способы ее совершенствования на основе самооценки                     | УК-6.1.               | Знает основные методы самооценки и анализа своей деятельности, а также принципы управления временем и целеполагания  |
|             |   | УК-6.2                | Умеет ставить реалистичные и достижимые цели, определять приоритеты в своей деятельности, а также разрабатывать и внедрять планы по совершенствованию своих навыков и компетенций на основе полученной самооценки                              |
|             |   | УК-6.3                | Имеет практический опыт применения методов самооценки в своей профессиональной деятельности, включая участие в тренингах, семинарах и проектах, направленных на развитие личной эффективности и профессионального роста                        |
| ОПК-2.      | Способен создавать и исследовать новые математические модели в естественных науках, совершенствовать и разрабатывать концепции, теории и методы | ОПК-2.1.              | Знает основные математические модели и методы, используемые в естественных науках, включая статистическое моделирование, дифференциальные уравнения и численные методы, а также современные подходы к исследованию и анализу данных            |
|             |   | ОПК-2.2               | Умеет разрабатывать и адаптировать математические модели для решения конкретных проблем в естественных науках, проводить их анализ и верификацию, а также интерпретировать полученные результаты в контексте научных исследований              |
|             |   | ОПК-2.3               | Имеет практический опыт создания и исследования математических моделей в рамках научных проектов или исследований, включая участие в публикациях, конференциях или коллаборациях, где были разработаны и апробированы новые концепции и методы |
| ПК-3.       | Способен решать задачи профессиональной   | ПК-3.1.               | Знает методы и инструменты продуктового менеджмента  |

|       |  |         |  |
|-------|--|---------|--|
|       | деятельности в области продуктового менеджмента, формулировать результаты анализа и выявлять последствия полученных данных для принятия обоснованных решений и оптимизации продуктов | ПК-3.2. | Умеет применять аналитические инструменты и программное обеспечение для обработки и визуализации данных, а также формулировать выводы на основе проведенного анализа   |
|       |  | ПК-3.3. | Имеет опыт работы над реальными проектами в области продуктового менеджмента, включая анализ пользовательского поведения и оптимизацию продуктов на основе полученных данных                                 |
| ПК-4. | Способен публично представлять собственные и известные научные результаты  | ПК-4.1. | Знает основные принципы эффективного публичного выступления, методы визуализации данных и основные требования к научным презентациям, включая структуру и содержание   |
|       |  | ПК-4.2. | Умеет четко и логично формулировать свои научные результаты, адаптируя их для различных аудиторий, а также использовать визуальные средства для улучшения восприятия информации                              |
|       |  | ПК-4.3. | Имеет практический опыт участия в научных конференциях, семинарах или других мероприятиях, где успешно представлял свои и известные научные результаты, получая обратную связь и взаимодействуя с аудиторией |

### 3. Тематический план

| №<br>п/п | Наименование<br>раздела дисциплины<br>(модуля) | Трудоемкость, академические часы |           |              |                                | ТКУ<br>(текущий контроль<br>успеваемости)                       |
|----------|--|----------------------------------|-----------|--------------|--------------------------------|---|
|          |  | <i>Очная форма</i>               |           |              |                                |   |
|          |  | Аудиторная работа                |           | Контр<br>оль | Самосто<br>ятельна<br>я работа |   |
| Лекции   | Семинары<br>(Практическ<br>ие занятия)         |                                  |           |              |                                |   |
| 1        | Введение в терминал и командную строку         | 1                                | 1         |              | 7                              | Домашнее задание<br>Подготовка к семинару                       |
| 2        | Основы Linux и файловой системы                | 1                                | 1         |              | 7                              | Домашнее задание<br>Подготовка к семинару                       |
| 3        | Продвинутые операции в Linux                   | 1                                | 1         |              | 7                              | Домашнее задание<br>Подготовка к семинару                       |
| 4        | Системы контроля версий. Основы                | 1                                | 1         |              | 7                              | Домашнее задание<br>Подготовка к семинару                       |
| 5        | Системы контроля версий. Продвинутые темы      | 1                                | 1         |              | 7                              | Домашнее задание<br>Подготовка к семинару                       |
| 6        | Ускорение разработки в текстовых редакторах    | 2                                | 2         |              | 7                              | Домашнее задание<br>Подготовка к семинару<br>Контрольная работа |
| 7        | Основы удаленного взаимодействия               | 1                                | 1         |              | 7                              | Домашнее задание<br>Подготовка к семинару                       |
| 8        | Bash Скриптинг. Основы                         | 1                                | 1         |              | 7                              | Домашнее задание<br>Подготовка к семинару                       |
| 9        | Bash Скриптинг. Продвинутые темы               | 1                                | 1         |              | 7                              | Домашнее задание<br>Подготовка к семинару                       |
| 10       | Ручная сборка проектов                         | 2                                | 2         |              | 7                              | Домашнее задание<br>Подготовка к семинару                       |
| 11       | Продвинутая работа в терминале                 | 1                                | 1         |              | 6                              | Домашнее задание<br>Подготовка к семинару                       |
| 12       | Основы сетевого взаимодействия                 | 1                                | 1         |              | 6                              | Домашнее задание<br>Подготовка к семинару                       |
|          | <i>Зачет</i>                                   |                                  |           | <b>4</b>     |                                |   |
|          | <b>Итого:</b>                                  | <b>14</b>                        | <b>14</b> | <b>4</b>     | <b>82</b>                      |   |
|          | <b>Объем дисциплины (модуля) (в ак. ч.)</b>    | <b>114</b>                       |           |              |                                |   |
|          | <b>Объем дисциплины (модуля) (в зач. ед.)</b>  | <b>3</b>                         |           |              |                                |   |

#### 4. Содержание дисциплины (модуля)

| №п/п | Наименование раздела дисциплины (модуля)    | Содержание дисциплины (модуля) по темам   |
|------|---|---|
| 1    | Введение в терминал и командную строку      | Основные команды в терминале. Знакомство с командной строкой в Linux  |
| 2    | Основы Linux и файловой системы             | Устройство файловой системы в Linux, отличие от MacOS и Windows. Понятие inode в Unix-системах  |
| 3    | Продвинутое операции в Linux                | Продвинутое операции для работы с файлами. Управляющие процессы (демоны). Работа с пакетными менеджерами  |
| 4    | Системы контроля версий. Основы             | Виды систем контроля версий. Базовые команды в Git: commit, push, checkout, switch, pull, fetch. Работа с удаленным репозиторием                    |
| 5    | Системы контроля версий. Продвинутое темы   | Работа с ветвлениями. Merge или Rebase. Контроль за процессами слияния. Git - продвинутое темы. Git Workflows, Git Hooks                            |
| 6    | Ускорение разработки в текстовых редакторах | Основные виды команд и Shortcut-ов в окружениях разработчика  |
| 7    | Основы удаленного взаимодействия            | Основы удаленного взаимодействия: SSH, SCP, подключение к удаленной командной строке  |
| 8    | Bash Скриптинг. Основы                      | Переменные окружения, аргументы командной строки. Работа с вводом и выводом   |
| 9    | Bash Скриптинг. Продвинутое темы            | Логические операции. Отличия разных видов фигурных скобок. Конвейерные команды (pipe, xargs)  |
| 10   | Ручная сборка проектов                      | Процесс запуска классов в Java. Файлы манифестов в Java-проектах. Команда make как процесс ускорения сборки проектов                                |
| 11   | Продвинутое работа в терминале              | Продвинутое работа в терминале для отслеживания и мониторинга процессов. Команды find, grep, cut, awk, parallel                                     |
| 12   | Основы сетевого взаимодействия              | Основы протокола HTTP, передача параметров через протокол HTTP. Понятие socket-а в Linux-системах. Знакомство с инструментами командной строки curl |

## 5. Учебно-методическое обеспечение

Университет располагает полным набором лицензионного и свободно распространяемого программного обеспечения, включая продукты отечественного производства.

Каждый студент в течение всего периода обучения получает индивидуальный неограниченный доступ к электронно-библиотечной системе и электронной информационно-образовательной среде университета. Эти системы предоставляют возможность доступа к ресурсам из любой точки, где есть подключение к сети Интернет, как на территории университета, так и за его пределами.

Студентам обеспечен удаленный доступ к современным профессиональным базам данных и информационным справочным системам.

### *Основная литература:*

1. Зараменских, Е. П. Разработка информационных систем : учебник и практикум для вузов / Е. П. Зараменских. — 2-е изд. — Москва : Издательство Юрайт, 2025. — 78 с. — (Высшее образование). — ISBN 978-5-534-21420-8. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/571333>.

### *Дополнительная литература:*

1. Linux. Командная строка. Лучшие практики. — СПб.: Питер, 2023. — 256 с.: ил. — (Серия «Бестселлеры О’Reilly»). ISBN 978-5-4461-2300-1.

2. Скуликар А. Изучаем Git: Пер. с англ. - Астана: АЛИСТ, 2024. - 288 с.: ил. ISBN 978-601-08-4120-8.

## 6. Материально-техническое обеспечение

Университет располагает материально-технической базой, соответствующей действующим противопожарным правилам и нормам и обеспечивающей проведение всех видов дисциплинарной и междисциплинарной подготовки, практической и научно-исследовательской работ обучающихся, предусмотренных учебным планом.

Помещения, которые представляют собой учебные аудитории для проведения занятий лекционного типа, занятий семинарского (практического) типа, групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации, а также помещения для самостоятельной работы и помещения для хранения и профилактического обслуживания учебного оборудования. Помещения укомплектованы специализированной мебелью и техническими средствами обучения, служащими для представления учебной информации большой аудитории.

Изучение дисциплины (модуля) обеспечивается в учебных аудиториях, оснащенных:

- столами и стульями;
- компьютерной техникой;
- специализированным оборудованием, включая демонстрационное оборудование.

Помещения для самостоятельной работы обучающихся, в том числе приспособленные для использования инвалидами и лицами с ограниченными возможностями здоровья, оснащены компьютерной техникой с возможностью подключения к сети «Интернет» и обеспечением доступа к в электронную информационно-образовательную среду Университета.

Обучающимся предоставляется доступ (в том числе удаленный) к ресурсам информационно-телекоммуникационной сети «Интернет», электронным ресурсам (в том числе электронным библиотечным системам, современным профессиональным базам данных и информационным справочным системам):

| №  | Наименование портала (издания, курса, документа)   | Ссылка  |
|----|--|---|
| 1  | Катастрофы, стихийные бедствия, аварии, эпидемии. Солнечная и геомагнитная активность. /ежедневный обзор | <a href="http://www.disasters.chat.ru">http://www.disasters.chat.ru</a>                       |
| 2  | Каталог по безопасности жизнедеятельности  | <a href="http://www.eun.chat.ru">http://www.eun.chat.ru</a>                                   |
| 3  | Научная электронная библиотека eLibrary.ru библиотека  | <a href="https://elibrary.ru/defaultx.asp">https://elibrary.ru/defaultx.asp</a>               |
| 4  | База данных для IT-специалистов  | <a href="https://habr.com">https://habr.com</a>   |
| 5  | База данных ScienceDirect  | <a href="https://www.sciencedirect.com">https://www.sciencedirect.com</a>                     |
| 6  | Официальный сайт Министерства науки и высшего образования Российской Федерации                           | <a href="https://minobrnauki.gov.ru/">https://minobrnauki.gov.ru/</a>                         |
| 7  | Федеральный портал «Российское образование»  | <a href="https://www.edu.ru/">https://www.edu.ru/</a>   |
| 8  | Информационная система "Единое окно доступа к образовательным ресурсам"                                  | <a href="http://window.edu.ru/">http://window.edu.ru/</a>                                     |
| 9  | Единая коллекция цифровых образовательных ресурсов   | <a href="http://school-collection.edu.ru/">http://school-collection.edu.ru/</a>               |
| 10 | Федеральный центр информационно - образовательных ресурсов   | <a href="http://fcior.edu.ru/">http://fcior.edu.ru/</a>                                       |
| 11 | Сайт различных плагинов  | <a href="https://maven.apache.org/plugin/s/">https://maven.apache.org/plugin/s/</a>           |
| 12 | Maven central repository - хранилище библиотек и фреймворков   | <a href="https://mvnrepository.com/repos/central">https://mvnrepository.com/repos/central</a> |

Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине (модулю), в том числе комплект лицензионного программного обеспечения, современные профессиональные базы данных и информационные справочные системы:

| Наименование ПО   | Производство  | Лицензионное / свободно распространяемое |
|---|---------------|--|
| <b>Операционные системы:</b>  |               |  |
| Microsoft Imagine (Windows Client, Server)                          | зарубежное    | лицензионное                             |
| <b>Браузеры:</b>  |               |  |
| Яндекс.Браузер  | отечественное | свободно распространяемое                |
| Google Chrome   | зарубежное    | свободно распространяемое                |
| <b>Офисные приложения:</b>  |               |  |
| Microsoft Imagine (Visio, OneNote)                                  | зарубежное    | лицензионное                             |
| TeXstudio   | зарубежное    | свободно распространяемое                |
| Adobe Acrobat Reader  | зарубежное    | свободно распространяемое                |
| <b>Программное обеспечение для планирования и учета времени:</b>    |               |  |
| Toggle app  | зарубежное    | свободно распространяемое                |
| <b>Системы управления проектами:</b>                                |               |  |
| Microsoft Imagine (Project)   | зарубежное    | лицензионное                             |
| <b>Системы управления базами данных:</b>                            |               |  |
| Microsoft Imagine (SQL Server)                                      | зарубежное    | лицензионное                             |
| <b>Системы резервного копирования (backup):</b>                     |               |  |
| Acronis Backup Advanced for HyperV                                  | зарубежное    | лицензионное                             |
| <b>Справочно-правовые системы:</b>                                  |               |  |
| КонсультантПлюс: справочно-правовая система                         | отечественное | лицензионное                             |
| <b>Средства антивирусной защиты:</b>                                |               |  |
| Kaspersky Endpoint Security для бизнеса Стандартный Russian Edition | отечественное | лицензионное                             |

|  |            |                           |
|--|------------|---------------------------|
| <b>Пакеты программных средств и библиотек:</b>           |            |                           |
| AutoPsy  | зарубежное | свободно распространяемое |
| Interactive Disassembler (IDA)                           | зарубежное | свободно распространяемое |
| <b>Системы управления библиографической информацией:</b> |            |                           |
| Zotero   | зарубежное | свободно распространяемое |
| <b>Сервисы и службы:</b>                                 |            |                           |
| Bind   | зарубежное | свободно распространяемое |
| Docker   | зарубежное | свободно распространяемое |

## 7. Методические и оценочные материалы

### Методические указания для обучающихся по освоению дисциплины (модуля)

В процессе изучения дисциплины (модуля) «Инструменты разработчика» в рамках текущего контроля успеваемости используются такие виды учебной работы, как лекции, семинары, подготовка к семинарам, домашние задания, контрольная работа, а также различные виды самостоятельной работы обучающихся по заданию преподавателя, направленные на развитие навыков профессиональной лексики, закрепление практических профессиональных компетенций, поощрение инициатив.

*Лекция* – систематическое, последовательное, монологическое изложение преподавателем учебного материала, как правило, теоретического характера.

В процессе лекций рекомендуется вести конспект лекций: кратко и схематично фиксировать основные идеи, выводы и обобщения лекции; выделять важные мысли, ключевые слова и термины. Необходимо отметить вопросы или материалы, которые вызывают затруднения, и попытаться найти ответы в рекомендованной литературе. Если разобраться в материале не удастся, следует сформулировать вопрос и задать его преподавателю на консультации или во время семинарского (практического) занятия.

*Участие в семинаре (аудиторная работа)* — это форма учебной деятельности, проводимая в учебном заведении под руководством преподавателя, где студенты активно участвуют в обсуждениях, практических заданиях и других формах взаимодействия.

Для успешной подготовки к аудиторной работе рекомендуется заранее ознакомиться с темой занятия и основными материалами, чтобы иметь возможность активно участвовать в обсуждении. Также полезно подготовить вопросы и идеи для обсуждения, что поможет глубже понять материал и продемонстрировать заинтересованность.

*Домашнее задание* – набор заданий по темам недели.

При работе над домашними заданиями важно внимательно ознакомиться с требованиями и сроками выполнения. Рекомендуется разбивать задания на этапы, чтобы избежать перегрузки и лучше усвоить материал. Использовать различные источники информации, включая учебники и онлайн-ресурсы, для более глубокого понимания темы.

*Контрольная работа* – письменная работа с набором задач, которые нужно решить за ограниченное время.

Цель контрольной работы – получить специальные знания по одной или нескольким темам дисциплины (модуля) и продемонстрировать навыки их практического применения.

*Самостоятельная работа* – работа студентов, направленная на углубленное изучение отдельных тем и вопросов учебной дисциплины (модуля).

В процессе самостоятельной работы студенты взаимодействуют с рекомендованными материалами при минимальном участии преподавателя. Задачи студента включают работу с конспектами лекций (обработка текста), повторное изучение учебных материалов планов и тезисов ответов, изучение дополнительных тем, выполнение учебно-исследовательских заданий и другое.

## Система оценивания результатов обучения по дисциплине (модулю)

**Критерии получения уровня и оценивания сформированности компетенций по дисциплине (модулю) «Инструменты разработчика».**

Оценивание уровня учебных достижений обучающихся по дисциплине (модулю) осуществляется в виде текущего контроля успеваемости.

**Промежуточная аттестация** по дисциплине (модулю) осуществляется в форме *зачета*, при этом проводится оценка компетенций, сформированных по дисциплине.

Для оценивания текущего контроля успеваемости и промежуточной аттестации используется десятибалльная шкала оценивания, которая соотносится с традиционной пятибалльной шкалой следующим образом:

| Десятибалльная оценка | Пятибалльная оценка | Оценка за зачет | Общая характеристика результата обучения по дисциплине (модулю)  |
|-----------------------|---------------------|-----------------|--|
| 10                    | Отлично             | Зачтено         | Студент полностью владеет знаниями, изложенными в рабочей программе, и глубоко осмысляет дисциплину (модуль). Он самостоятельно и логически последовательно отвечает на все вопросы, акцентируя внимание на наиболее важном. Умеет анализировать, сравнивать, классифицировать, обобщать, конкретизировать и систематизировать изученный материал, выделяя ключевые моменты и устанавливая причинно-следственные связи. Четко формулирует ответы, уверенно интерпретирует результаты анализов и других исследований, а также решает сложные задачи. Студент хорошо знаком с методами исследования, необходимыми для практической деятельности, и умеет связывать теоретические аспекты дисциплины (модуля) с практическими задачами. |
| 9                     | Отлично             | Зачтено         |  |
| 8                     | Отлично             | Зачтено         |  |
| 7                     | Хорошо              | Зачтено         | Студент обладает знаниями предмета почти в полном объеме рабочей программы и самостоятельно, логически последовательно и всесторонне отвечает на все вопросы, акцентируя внимание на наиболее значимых моментах. Он умеет анализировать, сравнивать, классифицировать, обобщать, конкретизировать и систематизировать изученный материал, выделяя его ключевые аспекты и устанавливая причинно-следственные связи. Формулирует свои ответы, уверенно интерпретирует результаты анализов и других исследований, а также решает  |
| 6                     | Хорошо              | Зачтено         |  |

| Десятибалльная оценка | Пятибалльная оценка | Оценка за зачет | Общая характеристика результата обучения по дисциплине (модулю)   |
|-----------------------|---------------------|-----------------|---|
|                       |                     |                 | сложные ситуационные задачи. Студент хорошо знаком с методами исследования, необходимыми для практической деятельности, и умеет связывать теоретические аспекты предмета с практическими задачами.  |
| 5                     | Удовлетворительно   | Зачтено         | Студент обладает базовыми знаниями по дисциплине (модулю), но испытывает трудности при самостоятельных ответах и использует неточные формулировки. В ходе ответов он допускает ошибки, касающиеся сути вопросов. Студент способен решать только самые простые задачи и владеет лишь минимальным набором методов исследования. |
| 4                     | Удовлетворительно   | Зачтено         |   |
| 3                     | Не сдан             | Не зачтено      | Студент не овладел обязательным минимумом знаний по предмету и не может ответить на вопросы, даже если преподаватель задает дополнительные наводящие вопросы.   |
| 2                     | Не сдан             | Не зачтено      |   |
| 1                     | Не сдан             | Не зачтено      |   |

Дисциплина (модуль) «Инструменты разработчика» оценивается следующим образом:

| Активность         | Вес | Описание  |
|--------------------|-----|---|
| Домашние задания   | 70% | 7 домашних заданий, которые оцениваются по критериям. Максимально можно набрать 10 баллов за каждое из заданий.   |
| Аудиторная работа  | 15% | На каждом семинаре студент может заработать баллы за интересные вопросы, работу на семинаре и выполнение заданий.   |
| Контрольная работа | 15% | Одна контрольная работа, на которой оценивается процент правильных ответов и конвертируется в количество набранных баллов (так, за 100% правильных ответов студент получает 10 баллов). |

**Формула расчёта итоговой оценки по дисциплине (модулю) «Инструменты разработчика»:**  $\langle 0,15 \times \text{среднее за аудиторную работу} + 0,7 \times \text{среднее за домашние задания} + 0,15 \times \text{контрольная работа} \rangle$ .

**Для получения зачет с оценкой нужно получить 4 или более баллов.**

**Текущий контроль успеваемости обучающихся по дисциплине (модулю)**

**Примерные вопросы для подготовки к семинару**

**Введение в терминал и командную строку**

1. Что такое терминал и каковы его основные функции?
2. Как открыть терминал в различных дистрибутивах Linux?
3. Назовите основные команды для навигации по файловой системе в терминале.
4. Как вывести содержимое текущей директории?
5. Как создать новую директорию с помощью командной строки?
6. Как удалить файл или директорию через терминал?

7. Что такое командная строка и как она отличается от графического интерфейса?

### **Основы Linux и файловой системы**

8. Как устроена файловая система в Linux?

9. В чем отличие файловой системы Linux от файловых систем MacOS и Windows?

10. Что такое inode и какую роль он играет в Unix-системах?

11. Как получить информацию об inode файла?

12. Какие типы файлов существуют в Linux?

### **Продвинутые операции в Linux**

13. Как использовать команды для копирования и перемещения файлов?

14. Что такое демоны и как они работают в Linux?

15. Как установить и удалить пакеты с помощью пакетного менеджера?

16. Как обновить систему с помощью пакетного менеджера?

### **Системы контроля версий. Основы**

17. Что такое система контроля версий и зачем она нужна?

18. Назовите основные команды Git и их функции.

19. Как создать новый репозиторий в Git?

20. Как выполнить команду push и что она делает?

### **Системы контроля версий. Продвинутые темы**

21. В чем разница между merge и rebase в Git?

22. Как создать новую ветку в Git?

23. Как контролировать процессы слияния в Git?

24. Что такое Git Hooks и как они используются?

### **Ускорение разработки в текстовых редакторах**

25. Какие основные команды и сочетания клавиш (shortcuts) существуют в текстовых редакторах?

26. Как настроить текстовый редактор для повышения продуктивности?

### **Основы удаленного взаимодействия**

27. Что такое SSH и как он используется для удаленного доступа?

28. Как скопировать файлы с помощью SCP?

29. Как подключиться к удаленной командной строке с помощью SSH?

### **Bash Скриптинг. Основы**

30. Что такое переменные окружения в Bash?

31. Как передавать аргументы командной строки в Bash-скрипт?

32. Как работать с вводом и выводом в Bash?

### **Bash Скриптинг. Продвинутые темы**

33. Как использовать логические операции в Bash?

34. В чем разница между различными видами фигурных скобок в Bash?

35. Как работают конвейерные команды (pipe) в Bash?

### **Ручная сборка проектов**

36. Как запустить классы в Java из командной строки?

37. Что такое файлы манифестов в Java-проектах?

38. Как использовать команду make для ускорения сборки проектов?

### **Продвинутая работа в терминале**

39. Как отслеживать и мониторить процессы в терминале?

40. Как использовать команды find, grep, cut, awk и parallel для обработки данных?

## **Примерные домашние задания**

### **Домашнее задание**

Необходимо написать аналог утилиты `du` с опцией `-bs`

Предполагается, что утилита всегда работает с файлами и директориями, в поддиректориях нет ссылок на другие файлы, все файлы при этом могут быть прочитаны пользователем.

Можно пользоваться только командами `stat` (или `ls -l`) и `awk`

Usage:

`./du.sh <path to file or directory>`

## Процедура сдачи задания

1. Создайте приватный репозиторий
2. Добавьте akhtyamov Pavel в коллабораторы репозитория (желательно это делать на `gitlab.akhcheck.ru`)
3. Добавьте пользователей в репозиторий:
  - `techprogchecker` - для GitHub
  - `checker` - для `gitlab.akhcheck.ru`
4. Создайте ветку `tasks-bash`
5. В ветке `tasks-bash` создайте папку `tasks-bash`
6. В папке `tasks-bash` создайте папку `du`
7. В папке `du` создайте скрипт `run.sh`, который необходим для сдачи задания

## Домашнее задание

Напишите скрипт, который позволяет разбить датасет в формате `csv` на `train` и `val` выборку.

Можно протестировать работу на датасете.

## Принимаемые параметры

- `--input ...` (путь к датасету)
- `--train_ratio ...` (доля объектов в обучающей выборке - от 0 до 100)
- `--shuffle` (Флаг есть, если необходимо перемешать датасет)
- `--train_file ...` (путь к `train` данным)
- `--val_file ...` (путь к `val` данным)

## Процедура сдачи задания

1. Создайте приватный репозиторий
2. Добавьте akhtyamov Pavel в коллабораторы репозитория (желательно это делать на `gitlab.akhcheck.ru`)
3. Добавьте пользователей в репозиторий:
  - `techprogchecker` - для GitHub
  - `checker` - для `gitlab.akhcheck.ru`
4. Создайте ветку `tasks-bash`
5. В ветке `tasks-bash` создайте папку `tasks-bash`
6. В папке `tasks-bash` создайте папку `train-val-split`

## Домашнее задание

### Описание задания

Вам предложен готовый проект на `java`, необходимо научиться его собирать с помощью `maven`.

### Общие требования к выполнению задания:

- Вносить правки в код и скрипты запрещено
- Необходимо указать, что для сборки необходимо использовать 11 версию языка `Java` для компилятора

- Укажите формат кодировки: **UTF-8**

### Алгоритм действий

1. Склонировать себе проект
  - Его можно найти [здесь](#)
2. Написать `pom.xml` для подпроекта `easterPreparation`
  - Необходимо осуществить компиляцию файлов и запуск тестовПроверка будет осуществляться командой `mvn test`
2. Написать сборку для всего проекта `easter`
  - Требования к подпроекту `easterPreparation` изложены в пункте 1
  - На каждый подпроект должен быть свой `pom.xml`
  - Для успешной сборки подпроекта `easterParty` требуется подпроект `easterPreparation`, а также файл `attenders.json`, который генерируется скриптом `create_people.sh`
    - Скрипт должен быть вызван с параметрами 3 и 4
    - Должен существовать `pom.xml` на весь проект, лежащий в директории `easter`
    - В результате сборки подпроекта `easterPreparation` должен получиться `jar`-архив. Его следует положить в папку `task-maven/jar/`, необходимые для его работы зависимости следует положить рядом с ним в отдельную папку, назвав ее `libs`
      - Подпроект `easterParty` должен иметь `artifactId: easter-party, a version: 1.0.0`Сборка будет осуществляться с помощью команды `mvn package`, запуск `jar java -jar [jar-name]`
3. Доработать сборку
  - Осуществить автоматическую генерацию `javadoc`-а: он за вас уже написан, поэтому настройте автоматическую генерацию сайта по существующим описаниям. Не забудьте собрать единый джава док из всех модулей! Переносить в другую директорию не нужно: все должно остаться в `task-maven/target/site/`
    - Настроить очистку проекта: необходимо почистить все папки для сборки, папку `jar` и файл `attenders.json`Данный этап будет проверяться следующей последовательностью команд: `mvn install -> mvn javadoc:aggregate -> mvn clean`

### Критерии оценивания

За первый пункт дается 3 балла, за второй - 4, за последний - оставшиеся 3 балла.

Обратите внимание, что для выполнения какого-либо пункта необходимо перед этим закрыть предыдущие.

### Процедура сдачи задания

1. Создайте в репозитории ветку `task-maven`
2. Скопируйте проект в ветку `task-maven`. В ветке `task-maven` должна быть папка `task-maven`. В папке `task-maven` должно быть содержимое проекта.
3. Создайте `merge request` из ветки `task-maven` в ветку `master/main`, добавьте ревьюера в `merge request` и не сливайте этот `merge request`!

## Примерные задания по контрольной работе

### Контрольная работа по системам контроля версий и Git

#### Задание 1: Виды систем контроля версий

1. Опишите основные виды систем контроля версий. В чем их отличия и преимущества?

Электронный документ

*Задание 2: Базовые команды в Git*

2. Объясните, что делает команда `git commit`. Какова её основная цель и какие параметры можно использовать?

*Задание 3: Работа с удаленным репозиторием*

3. Как вы можете подключиться к удаленному репозиторию в Git? Приведите пример команды.

*Задание 4: Команды для работы с ветками*

4. Каковы основные отличия между командами `git checkout` и `git switch`? Когда следует использовать каждую из них?

*Задание 5: Слияние и ребейз*

5. Объясните разницу между `merge` и `rebase`. В каких случаях предпочтительнее использовать каждую из этих операций?

*Задание 6: Контроль за процессами слияния*

6. Как вы можете разрешить конфликты, возникающие при слиянии веток в Git? Опишите шаги, которые необходимо предпринять.

*Задание 7: Git Workflows*

7. Опишите один из популярных Git Workflows (например, Git Flow или GitHub Flow). Каковы его основные этапы?

*Задание 8: Git Hooks*

8. Что такое Git Hooks? Приведите примеры ситуаций, в которых их можно использовать.

*Задание 9: Основные команды и Shortcuts*

9. Назовите и опишите три основных команды Git, которые вы используете чаще всего. Как они помогают в вашей работе?

*Задание 10: Сценарий использования*

10. Представьте, что вы работаете над проектом с командой. Опишите процесс, начиная с создания новой ветки для новой функции и заканчивая слиянием этой ветки обратно в основную. Укажите, какие команды Git вы будете использовать на каждом этапе.

**Задания для промежуточной аттестации по дисциплине (модулю)**

| № п/п | Задание  | Ответ                         | Компетенция |
|-------|--|-------------------------------|-------------|
| 1.    | Из файла <code>foo.md</code> создали два объекта: <ul style="list-style-type: none"><li>• символическую ссылку: <code>foo-soft-link.md</code></li><li>• жесткую ссылку: <code>foo-hard-link.md</code></li></ul> Далее файл <code>foo.md</code> переместили в файл <code>foo2.md</code> .<br><b>Укажи название файла, применение команды <code>cat</code> к которому завершиться с ошибкой.</b> | <code>foo-soft-link.md</code> | УК-6        |
| 2.    | Какие атрибуты прав имеет файл с правами <code>0735</code> ?<br>А. <code>rwX-wXr-X</code><br>Б. <code>r-xrw-rwx</code><br>В. <code>rwXr-xrw-</code><br>Г. <code>r-xrwxrw-</code>   | А                             | УК-6        |
| 3.    | В репозитории Git воссоздали следующую структуру первой фиксацией ( <code>commit-ом</code> ).<br><pre>graph TD; Root[.] --- docs; Root --- TEST[TEST.md]; Root --- README[README.md];</pre>  | 2                             | ПК-3        |

|    |   |                |       |
|----|---|----------------|-------|
|    | <pre>└─ tmp.txt</pre> <p>Выпишите количество объектов вида tree с их атрибутами, которые были созданы в локальном репозитории Git.</p>  |                |       |
| 4. | <p>На рисунке изображена система 파티ций файловой системы. Напишите название партиции с твердотельным накопителем, в котором расположены данные операционной системы, доступные для задач общего назначения</p> <pre>Filesystem      Size  Used Avail Use% Mounted on udev            32G   0    32G   0% /dev tmpfs           6,3G  3,0M  6,3G   1% /run /dev/nvme0n1p7 230G 187G  32G  86% / tmpfs           32G 161M  32G   1% /dev/shm tmpfs           5,0M  4,0K  5,0M   1% /run/lock tmpfs           32G   0    32G   0% /sys/fs/cgroup /dev/sdb        7,3T  6,9T  27G 100% /8tbsata /dev/nvme0n1p1 646M  73M  574M  12% /boot/efi /dev/sda5       1,5T  1,4T  48G  97% /2tbsata tmpfs           6,3G  36K  6,3G   1% /run/user/1000</pre> | /dev/nvme0n1p7 | УК-6  |
| 5. | <p>Напишите основной метод протокола HTTP, в рамках которых тело запроса может быть проигнорировано принимающим сервером.</p>   | GET            | ОПК-2 |
| 6. | <p>Команда ведет разработку ЦУ. Такси в общем Git-репозитории. Посмотрите на версии тегов и отсортируйте их в порядке возрастания версий (ответ запиши в последовательности букв, например: ДВГАБ – без знаков препинания и пробелов):</p> <p>Version<br/> А) 1.10.2-beta.15<br/> Б) 2.7.0<br/> В) 1.10.2<br/> Г) 1.15.1-alpha<br/> Д) 1.10.1-rc.2</p>  | ДАВГБ / давгб  | ПК-3  |
| 7. | <p>Какой код выведет скрипт ./01-script.sh в случае запуска при помощи следующих аргументов: перед запуском скрипта была запущена команда <b>export a=10</b></p> <pre>#!/bin/bash echo \$a sleep 2 a=1 ./01-script.sh</pre> <p>А. Последовательность из 10.<br/> Б. Последовательность из 1<br/> В. Пустой вывод.<br/> Г. 10 и последовательность из 1.</p>   | Г              | ПК-4  |
| 8. | <p>Укажите название протокола, при помощи которого происходит отправка в удаленный репозиторий без запроса пароля.</p>  | SSH/ssh        | УК-6  |

|     |   |                                      |       |
|-----|---|--------------------------------------|-------|
| 9.  | <p>Разработчик написал однострочный скрипт, который позволит найти все изображения расширения <code>.png</code>, находящиеся в папке <code>/home/cu</code></p> <p>Укажите опции команды <code>find</code>, которые использовал разработчик. Перечислите команды через запятую</p> | <code>-type,-name/-name,-type</code> | ПК-4  |
| 10. | <p>Укажите размер директории со следующей структурой:</p> <pre> . ├── docs │   └── TEST.md ├── README.md └── tmp.txt </pre> <p>Каждый inode вида директории составляет 4 КБ, каждый inode вида файл составляет 2 КБ.</p> <p>Ответ укажите в КБ.</p>                               | 16                                   | ОПК-2 |
| 11. | <p>В вашем проекте был настроен Git Flow. Однако в удаленном репозитории <code>origin</code> на каждый Merge в Main/Master ветку не был найден тег <code>v1.0</code></p>  | <code>git push origin v1.0</code>    | ПК-3  |
| 12. | <p>Назовите стадию сборки проекта, в котором один или несколько объектных файлов собираются в один общий бинарных файлов.</p>   | Линковка / линковка                  | УК-6  |