

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

**федеральное государственное автономное
образовательное учреждение высшего образования
«Национальный исследовательский Нижегородский государственный университет
им. Н.И. Лобачевского»**

Институт информационных технологий, математики и механики

УТВЕРЖДЕНО
решением Ученого совета ННГУ
протокол № 10 от 27.08.2025 г.

Рабочая программа дисциплины

Инструменты программирования

Уровень высшего образования
Бакалавриат

Направление подготовки / специальность
01.03.02 - Прикладная математика и информатика

Направленность образовательной программы
Математическое моделирование и искусственный интеллект

Форма обучения
очная

г. Нижний Новгород

2025 год начала подготовки

1. Место дисциплины в структуре ОПОП

Дисциплина Б1.В.17 Инструменты программирования относится к части, формируемой участниками образовательных отношений образовательной программы.

2. Планируемые результаты обучения по дисциплине, соотнесенные с планируемыми результатами освоения образовательной программы (компетенциями и индикаторами достижения компетенций)

Формируемые компетенции (код, содержание компетенции)	Планируемые результаты обучения по дисциплине (модулю), в соответствии с индикатором достижения компетенции		Наименование оценочного средства	
	Индикатор достижения компетенции (код, содержание индикатора)	Результаты обучения по дисциплине	Для текущего контроля успеваемости	Для промежуточной аттестации
ППК-Р6: Способен участвовать в промышленной разработке программного обеспечения (топ)	<p>ППК-Р6.1: Работает в соответствии с промышленными методологиями разработки</p> <p>ППК-Р6.2: Использует инструменты промышленной разработки</p> <p>ППК-Р6.3: Разрабатывает масштабируемый и поддерживаемый код</p>	<p>ППК-Р6.1:</p> <p>ППК-Р6.1. 3-1. Знает принципы Agile и их применение в промышленных проектах.</p> <p>ППК-Р6.1. 3-2. Знает процессы code review, принципы коллективного владения кодом (collective code ownership).</p> <p>ППК-Р6.1. У-1. Умеет оценивать объем задачи и срок ее выполнения, участвовать в планировании спринтов.</p> <p>ППК-Р6.1. У-2. Умеет работать в команде с использованием инструментов управления проектами.</p> <p>ППК-Р6.2:</p> <p>ППК-Р6.2. 3-1. Знает принципы Continuous Integration and Continuous Delivery (CI/CD).</p> <p>ППК-Р6.2. 3-2. Знает системы мониторинга и логирования в продуктивной среде.</p> <p>ППК-Р6.2. У-1. Умеет настраивать потоки работ CI/CD.</p> <p>ППК-Р6.2. У-2. Умеет работать с контейнеризацией и оркестрацией.</p> <p>ППК-Р6.2. У-2. Умеет</p>	Собеседование Задания	Зачёт: Задания Контрольная работа

		<p>настраивать мониторинг в продуктивной среде.</p> <p>ППК-Р6.3: ППК-Р6.3. 3-1. Знает принципы чистого кода, SOLID, DRY, KISS и др. ППК-Р6.3. 3-2. Знает принципы предметно-ориентированного проектирования (ПОП) программного обеспечения. ППК-Р6.3. 3-3. Знает паттерны проектирования и антипаттерны. ППК-Р6.3. У-1. Умеет разрабатывать модульный и тестируемый программный код. ППК-Р6.3. У-2. Умеет выполнять модульное, интеграционное и нагрузочное тестирование. ППК-Р6.3. У-3. Умеет проводить рефакторинг для повышения качества кода.</p>		
--	--	---	--	--

3. Структура и содержание дисциплины

3.1 Трудоемкость дисциплины

	очная
Общая трудоемкость, з.е.	2
Часов по учебному плану	72
в том числе	
аудиторные занятия (контактная работа):	
- занятия лекционного типа	16
- занятия семинарского типа (практические занятия / лабораторные работы)	16
- КСР	1
самостоятельная работа	39
Промежуточная аттестация	0
	Зачёт

3.2. Содержание дисциплины

(структурированное по темам (разделам) с указанием отведенного на них количества академических часов и виды учебных занятий)

Наименование разделов и тем дисциплины	Всего	в том числе
--	-------	-------------

	(часы)	Контактная работа (работа во взаимодействии с преподавателем), часы из них			Самостоятельная работа обучающегося, часы
		Занятия лекционного типа	Занятия семинарского типа (практические занятия/лабораторные работы), часы	Всего	
	0 Ф 0	0 Ф 0	0 Ф 0	0 Ф 0	0 Ф 0
Рабочее место программиста	3	1		1	2
Текстовые форматы	4	1	1	2	2
Обработка текста и регулярные выражения	4	1	1	2	2
Текстовые редакторы	4	1	1	2	2
Автоматизация: командная строка	4	1	1	2	2
Автоматизация: скриптовые языки	5	1	1	2	3
Системы контроля версий	5	1	1	2	3
Интегрированные среды разработки	4	1	1	2	2
Описание и построение проектов	4	1	1	2	2
Анализ бинарных модулей	4	1	1	2	2
Контроль качества кода	4	1	1	2	2
Отладка	4	1	1	2	2
Тестирование	4	1	1	2	2
Непрерывная интеграция	3	0	1	1	2
Командная разработка. Формирование сообщества	3	1		1	2
Анализ производительности программ с использованием инструментов профилировки	3	1	1	2	1
Введение в оптимизацию программ под ограниченные вычислительные ресурсы	4	1	1	2	2
Введение в использование инструментов ИИ при разработке программ	5		1	1	4
Аттестация	0				
КСР	1			1	
Итого	72	16	16	33	39

Содержание разделов и тем дисциплины

Цели и задачи изучения дисциплины.

Целью освоения дисциплины является овладение инструментальными средствами разработки программного обеспечения.

Данная цель предполагает решение следующих задач:

1. Изучение возможностей текстовых редакторов. Регулярные выражения.
2. Освоение базовых команд командной строки.
3. Изучение базовых возможностей скриптовых языков для автоматизации рутинной работы разработчика.
4. Изучение состава и возможностей интегрированных сред разработки и систем контроля версий.
5. Освоение основ командной разработки (непрерывная интеграция).

1. Рабочее место программиста
2. Текстовые форматы
3. Обработка текста и регулярные выражения
4. Текстовые редакторы
5. Автоматизация: командная строка
6. Автоматизация: скриптовые языки
7. Системы контроля версий
8. Интегрированные среды разработки
9. Описание и построение проектов
10. Анализ бинарных модулей
11. Контроль качества кода
12. Отладка
13. Тестирование
14. Непрерывная интеграция
15. Командная разработка. Формирование сообщества
16. Анализ производительности программ с использованием инструментов профилировки
17. Введение в оптимизацию программ под ограниченные вычислительные ресурсы
18. Введение в использование инструментов ИИ при разработке программ

4. Учебно-методическое обеспечение самостоятельной работы обучающихся

Самостоятельная работа обучающихся включает в себя подготовку к контрольным вопросам и заданиям для текущего контроля и промежуточной аттестации по итогам освоения дисциплины приведенным в п. 5.

– Страуструп Б. Курс «Язык программирования C++ для профессионалов».

<http://www.intuit.ru/studies/courses/98/98/info>

– Сафонов А. Возможности Visual Studio 2013 и их использование для облачных вычислений. Лекция 1: Концепция современной интегрированной среды разработки приложений.

<http://www.intuit.ru/studies/courses/13996/1223/lecture/23386>

– Котляров В. Основы тестирования программного обеспечения.

<http://www.intuit.ru/studies/courses/48/48/info>

- Инструменты ИИ <https://giga.chat/>

- материалы дисциплины по анализу производительности и оптимизации программ <https://hpc-education.unn.ru/ru/обучение/курсы/оптимизация>

5. Фонд оценочных средств для текущего контроля успеваемости и промежуточной аттестации по дисциплине (модулю)

5.1 Типовые задания, необходимые для оценки результатов обучения при проведении текущего контроля успеваемости с указанием критериев их оценивания:

5.1.1 Типовые задания (оценочное средство - Собеседование) для оценки сформированности компетенции ППК-Р6:

1. Общее назначение инструментов, примеры.
2. Признаки "хороших" инструментов, с пояснениями.
3. Примеры практик Программной инженерии, их суть.
4. Приведите примеры инструментов, которые помогают применять практики.
5. Диаграмма каскадной модели жизненного цикла.
6. Диаграмма работы программиста над задачей.
7. Определение системы контроля версий (СКВ).
8. Основные функции/возможности современных СКВ.
9. Преимущества распределенной системы контроля версий (Distributed version control systems, DVCS).
10. Модель работы с репозиторием. Centralized Workflow (диаграмма, достоинства и недостатки)

Критерии оценивания (оценочное средство - Собеседование)

Оценка	Критерии оценивания
зачтено	Студент дал развернутый ответ на все вопросы без существенных ошибок.
не зачтено	При ответе студент допускает грубые ошибки в основном материале.

5.1.2 Типовые задания (оценочное средство - Задания) для оценки сформированности компетенции ППК-Р6:

1. Настройка окружения разработки.
2. Проектирование программного интерфейса (API), создание пользовательской документации.
3. Разработка функциональности.
4. Интеграция кода в общий проект.
5. Создание модульных тестов.

Критерии оценивания (оценочное средство - Задания)

Оценка	Критерии оценивания
зачтено	Выполнена основная часть задания, возможно с незначительными недочетами
не зачтено	Выполнено менее половины задания, есть существенные недочеты

5.2. Описание шкал оценивания результатов обучения по дисциплине при промежуточной аттестации

Шкала оценивания сформированности компетенций

Уровень сформированности компетенций (индикатора достижения компетенций)	плохо	неудовлетворительно	удовлетворительно	хорошо	очень хорошо	отлично	превосходно
	не зачтено			зачтено			
<u>Знания</u>	Отсутствие знаний теоретического материала. Невозможность оценить полноту знаний вследствие отказа обучающегося от ответа	Уровень знаний ниже минимальных требований. Имели место грубые ошибки	Минимально допустимый уровень знаний. Допущено много негрубых ошибок	Уровень знаний в объеме, соответствующем программе подготовки. Допущено несколько негрубых ошибок	Уровень знаний в объеме, соответствующем программе подготовки. Допущено несколько несущественных ошибок	Уровень знаний в объеме, соответствующем программе подготовки. Ошибок нет.	Уровень знаний в объеме, превышающем программу подготовки.
<u>Умения</u>	Отсутствие минимальных умений. Невозможность оценить наличие умений вследствие отказа обучающегося от ответа	При решении стандартных задач не продемонстрированы основные умения. Имели место грубые ошибки	Продемонстрированы основные умения. Решены типовые задачи с негрубыми ошибками. Выполнены все задания, но не в полном объеме	Продемонстрированы все основные умения. Решены все основные задачи с негрубыми ошибками. Выполнены все задания в полном объеме, но некоторые с недочетами	Продемонстрированы все основные умения. Решены все основные задачи. Выполнены все задания в полном объеме, но некоторые с недочетами	Продемонстрированы все основные умения. Решены все основные задачи с отдельными несущественными недочетами, выполнены все задания в полном объеме	Продемонстрированы все основные умения. Решены все основные задачи. Выполнены все задания, в полном объеме без недочетов
<u>Навыки</u>	Отсутствие базовых навыков. Невозможность оценить наличие навыков вследствие отказа обучающегося от ответа	При решении стандартных задач не продемонстрированы базовые навыки. Имели место грубые ошибки	Имеется минимальный набор навыков для решения стандартных задач с некоторыми недочетами	Продемонстрированы базовые навыки при решении стандартных задач с некоторыми недочетами	Продемонстрированы базовые навыки при решении стандартных задач без ошибок и недочетов	Продемонстрированы навыки при решении нестандартных задач без ошибок и недочетов	Продемонстрирован творческий подход к решению нестандартных задач

Шкала оценивания при промежуточной аттестации

Оценка		Уровень подготовки
зачтено	превосходно	Все компетенции (части компетенций), на формирование которых направлена дисциплина, сформированы на уровне не ниже «превосходно», продемонстрированы знания, умения, владения по соответствующим компетенциям на уровне выше предусмотренного программой

	отлично	Все компетенции (части компетенций), на формирование которых направлена дисциплина, сформированы на уровне не ниже «отлично».
	очень хорошо	Все компетенции (части компетенций), на формирование которых направлена дисциплина, сформированы на уровне не ниже «очень хорошо»
	хорошо	Все компетенции (части компетенций), на формирование которых направлена дисциплина, сформированы на уровне не ниже «хорошо».
	удовлетворительно	Все компетенции (части компетенций), на формирование которых направлена дисциплина, сформированы на уровне не ниже «удовлетворительно», при этом хотя бы одна компетенция сформирована на уровне «удовлетворительно»
не зачтено	неудовлетворительно	Хотя бы одна компетенция сформирована на уровне «неудовлетворительно».
	плохо	Хотя бы одна компетенция сформирована на уровне «плохо»

5.3 Типовые контрольные задания или иные материалы, необходимые для оценки результатов обучения на промежуточной аттестации с указанием критериев их оценивания:

5.3.1 Типовые задания (оценочное средство - Задания) для оценки сформированности компетенции ППК-Р6

1. Преимущества и недостатки простого текста.
2. Преимущества и недостатки бинарного формата.
3. Примеры ситуаций, когда удобно использовать TXT, XML, YAML, JSON.

Критерии оценивания (оценочное средство - Задания)

Оценка	Критерии оценивания
зачтено	Выполнена основная часть задания, возможно с незначительными недочетами.
не зачтено	Выполнено менее половины задания, есть существенные недочеты.

5.3.2 Типовые задания (оценочное средство - Контрольная работа) для оценки сформированности компетенции ППК-Р6

1. Общее назначение инструментов, примеры.
2. Признаки "хороших" инструментов, с пояснениями.
3. Примеры практик Программной инженерии, их суть.
4. Приведите примеры инструментов, помогающих применять практики.
5. Диаграмма каскадной модели жизненного цикла.
6. Диаграмма работы программиста над задачей.
7. Определение системы контроля версий (СКВ).
8. Основные функции/возможности современных СКВ.

9. Преимущества распределенной системы контроля версий (Distributed version control systems, DVCS).

10. Модель работы с репозиторием. Centralized Workflow (диаграмма, достоинства и недостатки)

Критерии оценивания (оценочное средство - Контрольная работа)

Оценка	Критерии оценивания
зачтено	Студент ответил на большую часть вопросов возможно с незначительными недочетами.
не зачтено	При ответе студент допускает грубые ошибки в основном материале и решении стандартных задач.

6. Учебно-методическое и информационное обеспечение дисциплины (модуля)

Основная литература:

1. Язык программирования C++ для профессионалов / Страуструп Б. - Москва : ИНТУИТ, 2016., <https://e-lib.unn.ru/MegaPro/UserEntry?Action=FindDocs&ids=663690&idb=0>.
2. Щербак Алексей Викторович. Тестирование программного обеспечения : учебник для вузов / А. В. Щербак. - Москва : Юрайт, 2025. - 145 с. - (Высшее образование). - ISBN 978-5-534-19291-9 : 589.00. - Текст : электронный // ЭБС "Юрайт"., <https://e-lib.unn.ru/MegaPro/UserEntry?Action=FindDocs&ids=963212&idb=0>.
3. Тестирование и верификация ПО: Практикум / Чернов Е. А., Петренко А. А., Петрова А. А., Овчинникова М. А. - Москва : РТУ МИРЭА, 2025. - 106 с. - Книга из коллекции РТУ МИРЭА - Информатика. - ISBN 978-5-7339-2491-5., <https://e-lib.unn.ru/MegaPro/UserEntry?Action=FindDocs&ids=976389&idb=0>.
4. Турнецкая Е. Л. Программная инженерия. Интеграционный подход к разработке : учебник для вузов / Турнецкая Е. Л., Аграновский А. В. - 2-е изд., стер. - Санкт-Петербург : Лань, 2025. - 216 с. - Книга из коллекции Лань - Информатика. - ISBN 978-5-507-50848-8., <https://e-lib.unn.ru/MegaPro/UserEntry?Action=FindDocs&ids=973811&idb=0>.
5. Попова Ю. Б. Тестирование и отладка программного обеспечения : пособие / Попова Ю. Б. - Минск : БНТУ, 2020. - 66 с. - Рекомендовано УМО по образованию в области информатики и радиоэлектроники для направления специальности 1-40 05 01-04 «Информационные системы и технологии (в обработке и представлении информации)». - Книга из коллекции БНТУ - Информатика. - ISBN 978-985-583-056-7., <https://e-lib.unn.ru/MegaPro/UserEntry?Action=FindDocs&ids=805658&idb=0>.
6. Рощин П. Г. Командная разработка программного обеспечения с помощью системы контроля версий Git: Конспект лекций : учебное пособие / Рощин П. Г. - Москва : НИЯУ МИФИ, 2022. - 108 с. - Книга из коллекции НИЯУ МИФИ - Информатика. - ISBN 978-5-7262-2846-4., <https://e-lib.unn.ru/MegaPro/UserEntry?Action=FindDocs&ids=884202&idb=0>.
7. Introduction to performance optimization using Intel SW tools / Ануфриенко А.В., Идрисов Р.И. - Москва : ИНТУИТ, 2016., <https://e-lib.unn.ru/MegaPro/UserEntry?Action=FindDocs&ids=662824&idb=0>.

Дополнительная литература:

1. Возможности Visual Studio 2013 и их использование для облачных вычислений / Сафонов В.О. - Москва : ИНТУИТ, 2016., <https://e-lib.unn.ru/MegaPro/UserEntry?Action=FindDocs&ids=662959&idb=0>.
2. Чернышев Станислав Андреевич. Принципы, паттерны и методологии разработки программного обеспечения : учебник для вузов / С. А. Чернышев. - Москва : Юрайт, 2025. - 176 с. - (Высшее образование). - ISBN 978-5-534-14383-6 : 789.00. - Текст : электронный // ЭБС "Юрайт"., <https://e-lib.unn.ru/MegaPro/UserEntry?Action=FindDocs&ids=961318&idb=0>.
3. Альтман Е. А. Система контроля версий GIT : учебно-методическое пособие к выполнению лабораторных работ / Альтман Е. А., Александров А. В., Васеева Т. В. - Омск : ОмГУПС, 2021. - 26 с. - Утверждено методическим советом университета. - Библиогр.: доступна в карточке книги, на сайте ЭБС Лань. - Книга из коллекции ОмГУПС - Информатика., <https://e-lib.unn.ru/MegaPro/UserEntry?Action=FindDocs&ids=782870&idb=0>.

Программное обеспечение и Интернет-ресурсы (в соответствии с содержанием дисциплины):

1. Интегрированная среда разработки: Borland Builder, Microsoft Visual Studio, Eclipse, QtCreator, Xcode. Одна из перечисленных на выбор студента.
2. Клиент системы контроля версий Git: git-scm[<https://git-scm.com>], GitHub Desktop [<https://desktop.github.com>]. Один из перечисленных на выбор студента.
3. GitHub [<http://github.com>].
4. Утилита для сборки проектов CMake [<http://cmake.org>].
5. GoogleTest [<https://github.com/google/googletest>].
6. TravisCI [<https://travis-ci.org>].
7. Redmine [<http://www.redmine.org>].

7. Материально-техническое обеспечение дисциплины (модуля)

Учебные аудитории для проведения учебных занятий, предусмотренных образовательной программой, оснащены мультимедийным оборудованием (проектор, экран), техническими средствами обучения, компьютерами.

Помещения для самостоятельной работы обучающихся оснащены компьютерной техникой с возможностью подключения к сети "Интернет" и обеспечены доступом в электронную информационно-образовательную среду.

Программа составлена в соответствии с требованиями ОС ННГУ по направлению подготовки/специальности 01.03.02 - Прикладная математика и информатика.

Автор(ы): Мееров Иосиф Борисович, кандидат технических наук, доцент.

Рецензент(ы): Ломакина Любовь Сергеевна, доктор технических наук.

Заведующий кафедрой: Мееров Иосиф Борисович, кандидат технических наук.

Программа одобрена на заседании методической комиссии от 25.06.2025, протокол № Протокол №11.