

**Министерство науки и высшего образования РФ**  
**Федеральное государственное бюджетное образовательное учреждение**  
**высшего образования**  
**«Национальный исследовательский университет «МЭИ»**

---

Направление подготовки/специальность: 01.03.02 Прикладная математика и информатика

Наименование образовательной программы: Математическое и программное обеспечение вычислительных машин и компьютерных сетей

Уровень образования: высшее образование - бакалавриат

Форма обучения: Очная

**Рабочая программа дисциплины**  
**ВЕДЕНИЕ КРУПНЫХ ПРОГРАММНЫХ ПРОЕКТОВ**

<b>Блок:</b>	Блок 1 «Дисциплины (модули)»
<b>Часть образовательной программы:</b>	Часть, формируемая участниками образовательных отношений
<b>№ дисциплины по учебному плану:</b>	Б1.Ч.13
<b>Трудоемкость в зачетных единицах:</b>	8 семестр - 4;
<b>Часов (всего) по учебному плану:</b>	144 часа
<b>Лекции</b>	8 семестр - 28 часа;
<b>Практические занятия</b>	8 семестр - 14 часов;
<b>Лабораторные работы</b>	8 семестр - 14 часов;
<b>Консультации</b>	8 семестр - 2 часа;
<b>Самостоятельная работа</b>	8 семестр - 85,5 часа;
<b>в том числе на КП/КР</b>	не предусмотрено учебным планом
<b>Иная контактная работа</b>	проводится в рамках часов аудиторных занятий
<b>включая:</b> Контрольная работа Лабораторная работа Реферат	
<b>Промежуточная аттестация:</b>	
<b>Экзамен</b>	8 семестр - 0,5 часа;

**Москва 2025**

## ПРОГРАММУ СОСТАВИЛ:

Преподаватель

---

	Подписано электронной подписью ФГБОУ ВО «НИУ «МЭИ»	
	Сведения о владельце ЦЭП МЭИ	
	Владелец	Алехин Р. В.
	Идентификатор	Rdb72ad74-AlekhinRV-aacea985

Р.В. Алехин

---

## СОГЛАСОВАНО:

Руководитель  
образовательной  
программы

---

	Подписано электронной подписью ФГБОУ ВО «НИУ «МЭИ»	
	Сведения о владельце ЦЭП МЭИ	
	Владелец	Ионова Т. В.
	Идентификатор	R5ac51726-IonovaTV-b9dd3591

Т.В. Ионова

---

Заведующий выпускающей  
кафедрой

---

	Подписано электронной подписью ФГБОУ ВО «НИУ «МЭИ»	
	Сведения о владельце ЦЭП МЭИ	
	Владелец	Варшавский П. Р.
	Идентификатор	R9a563c96-VarshavskyPR-efb4bbd

П.Р.  
Варшавский

---

## 1. ЦЕЛИ И ЗАДАЧИ ОСВОЕНИЯ ДИСЦИПЛИНЫ

**Цель освоения дисциплины:** изучение технологии реализации крупных программных проектов.

### Задачи дисциплины

- ознакомление с проблемами, возникающими при реализации крупных программных проектов;
- изучение современных технологий и методологий разработки программного обеспечения;
- ознакомление с концепцией фабрики программного обеспечения и приобретение навыка ее внедрения;
- получение опыта использования современных средств автоматизации процесса разработки программ.

Формируемые у обучающегося **компетенции** и запланированные **результаты обучения** по дисциплине, соотнесенные с **индикаторами достижения компетенций**:

Код и наименование компетенции	Код и наименование индикатора достижения компетенции	Запланированные результаты обучения
ПК-1 Способен выполнять все этапы жизненного цикла программного обеспечения	ИД-1 <sub>ПК-1</sub> Демонстрирует знания современных CASE-средств для выполнения этапов жизненного цикла ПО	знать: - современные методологии разработки программного обеспечения.  уметь: - применять современные CASE-средства.
ПК-1 Способен выполнять все этапы жизненного цикла программного обеспечения	ИД-3 <sub>ПК-1</sub> Формирует архитектуру ПО	знать: - современные архитектурные концепции, применяемые при разработке программных систем.  уметь: - применять типовые архитектурные решения.
ПК-2 Способен проектировать и реализовывать базы данных	ИД-2 <sub>ПК-2</sub> Выбирает СУБД на основе их характеристики	знать: - современную технику планирования работ в проекте по разработке программного обеспечения.  уметь: - анализировать архитектуру программных систем, оценивать уместность применения той или иной разновидности решения и технологии доступа к данным.
ПК-2 Способен проектировать и реализовывать базы данных	ИД-3 <sub>ПК-2</sub> Формирует и реализует базы данных в СУБД и организует программный доступ к данным	знать: - технику снижения трудоемкости крупного проекта по разработке программного обеспечения.  уметь: - создавать и внедрять стандарты кодирования и проектирования.

## **2. МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ОБРАЗОВАТЕЛЬНОЙ ПРОГРАММЫ ВО**

Дисциплина относится к основной профессиональной образовательной программе Математическое и программное обеспечение вычислительных машин и компьютерных сетей (далее – ОПОП), направления подготовки 01.03.02 Прикладная математика и информатика, уровень образования: высшее образование - бакалавриат.

Требования к входным знаниям и умениям:

- знать программную инженерию
- знать языки и методы программирования
- знать методы контроля программ

Результаты обучения, полученные при освоении дисциплины, необходимы при выполнении выпускной квалификационной работы.

### 3. СТРУКТУРА И СОДЕРЖАНИЕ ДИСЦИПЛИНЫ

#### 3.1 Структура дисциплины

Общая трудоемкость дисциплины составляет 4 зачетных единицы, 144 часа.

№ п/п	Разделы/темы дисциплины/формы промежуточной аттестации	Всего часов на раздел	Семестр	Распределение трудоемкости раздела (в часах) по видам учебной работы										Содержание самостоятельной работы/ методические указания
				Контактная работа							СР			
				Лек	Лаб	Пр	Консультация		ИКР		ПА	Работа в семестре	Подготовка к аттестации /контроль	
КПР	ГК	ИККП	ТК											
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1	Управление крупным проектом по разработке программного обеспечения	22	8	6	3	3	-	-	-	-	-	10	-	<p><b><u>Самостоятельное изучение теоретического материала:</u></b> Изучение дополнительного материала по разделу "Управление крупным проектом по разработке программного обеспечения"</p> <p><b><u>Изучение материалов литературных источников:</u></b></p> <p>[1], ч. 1 [2], р.1,2 [4], п. 1.1 [5], р. 2 [6], с. 51-68</p>
1.1	Методы сетевого планирования	12		3	1	2	-	-	-	-	-	6	-	
1.2	Гибкое планирование	10		3	2	1	-	-	-	-	-	4	-	
2	Техника снижения трудоемкости крупного проекта по разработке программного обеспечения	86	8	22	11	11	-	-	-	-	-	42	-	<p><b><u>Самостоятельное изучение теоретического материала:</u></b> Изучение дополнительного материала по разделу "Техника снижения трудоемкости крупного проекта по разработке программного обеспечения"</p> <p><b><u>Изучение материалов литературных источников:</u></b></p> <p>[3], п. 1</p>
2.1	Системы контроля версий и конфигурационного управления.	14		3	2	1	-	-	-	-	-	8	-	
2.2	Автоматизация сборки.	14		3	2	1	-	-	-	-	-	8	-	
2.3	Генерация кода по шаблону.	12		3	1	2	-	-	-	-	-	6	-	
2.4	Контроль работоспособности	13		3	2	2	-	-	-	-	-	6	-	

	кода.												
2.5	Контроль качества.	11	4	2	1	-	-	-	-	-	4	-	
2.6	Автоматизация тестирования.	12	3	1	2	-	-	-	-	-	6	-	
2.7	Автоматическое документирование и средства корпоративной памяти.	10	3	1	2	-	-	-	-	-	4	-	
	Экзамен	36.0	-	-	-	-	2	-	-	0.5	-	33.5	
	<b>Всего за семестр</b>	<b>144.0</b>	<b>28</b>	<b>14</b>	<b>14</b>	<b>-</b>	<b>2</b>	<b>-</b>	<b>-</b>	<b>0.5</b>	<b>52</b>	<b>33.5</b>	
	<b>Итого за семестр</b>	<b>144.0</b>	<b>28</b>	<b>14</b>	<b>14</b>	<b>2</b>	<b>-</b>	<b>-</b>	<b>-</b>	<b>0.5</b>	<b>85.5</b>		

**Примечание:** Лек – лекции; Лаб – лабораторные работы; Пр – практические занятия; КПр – аудиторные консультации по курсовым проектам/работам; ИККП – индивидуальные консультации по курсовым проектам/работам; ГК- групповые консультации по разделам дисциплины; СР – самостоятельная работа студента; ИКР – иная контактная работа; ТК – текущий контроль; ПА – промежуточная аттестация

## 3.2 Краткое содержание разделов

### 1. Управление крупным проектом по разработке программного обеспечения

#### 1.1. Методы сетевого планирования

Структурная декомпозиция работ.. Разработка структурной декомпозиции работ.. Сетевая модель проекта и метод критического пути.. Диаграммы Гантта. Построение диаграммы Гантта..

#### 1.2. Гибкое планирование

Техника гибкого планирования.. Системы управления задачами и заявками.. Обзор практик для повышения эффективности процессов разработки (Development) и эксплуатации (Operation) программного обеспечения (ПО) за счет их непрерывной интеграции и активного взаимодействия профильных специалистов с помощью инструментов автоматизации..

### 2. Техника снижения трудоемкости крупного проекта по разработке программного обеспечения

#### 2.1. Системы контроля версий и конфигурационного управления.

Модель работы с изменяющейся информацией. Контроль версий.. Модели версионирования.. Система управления версиями Subversion.. Система контроля версий GIT. Процесс Git Flow.. Система управления проектом Microfocus StarTeam..

#### 2.2. Автоматизация сборки.

Задача автоматизации сборки. Требования к системам автоматизации сборки.. Типы систем автоматизации сборки.. Apache Ant.. MS Build.. Компонентная сборка.. Apache Maven.. JetBrains TeamCity.. Jenkins.. Component Builder..

#### 2.3. Генерация кода по шаблону.

Стандартизация в разработке больших программных комплексов. Генерация кода. Методы генерации кода по шаблону.. Генерация кода на основе диаграмм.. Прямое визуальное проектирование.. Использование макропроцессоров для генерации кода.. Технология XSLT.. Создание шаблонов приложений в Embarcadero RAD Studio.. Шаблоны проектов и генерация каркасов типовых приложений в Eclipse.. Генерация кода с применением Java Emitter Templates.. Шаблоны проектов и генерация типовых приложений в Microsoft Visual Studio.. Разработка шаблонов с применением Text template transformation toolkit..

#### 2.4. Контроль работоспособности кода.

Работоспособность кода.. Средства контроля работоспособности кода.. Модульное тестирование.. Разработка посредством тестирования.. Разработка модульных тестов в среде Visual Studio..

#### 2.5. Контроль качества.

Качество кода. Оценка качества кода.. Метод метрик. Основные метрики кода.. Обнаружение и исправление ошибок в коде C/C++, C#, Java.. Контроль качества кода на базе правил.. Средства статического анализа кода. FxCop. Roslyn Analyzers.. Применение расширений Visual Assist X и JetBrains ReSharper..

#### 2.6. Автоматизация тестирования.

Средства автоматизации тестирования. Автоматизация тестирования приложений через графический интерфейс пользователя.. Применение планировщиков для автоматизации тестирования. Специализированные программы-планировщики. Планировщик xStarter. Планировщик Macro Scheduler.. Автоматизация тестирования приложений с помощью Microfocus Silk Test.. Автоматизация тестирования программ с помощью IBM Rational Functional Tester..

2.7. Автоматическое документирование и средства корпоративной памяти.

Автоматическая генерация документации.. Системы корпоративной памяти и накопления знаний.. Системы накопления знаний, организованные по принципу «единого источника»..

### **3.3. Темы практических занятий**

1. Разработка набора тестовых данных для автоматических тестов;
2. Разработка структурной декомпозиции работ (СДР);
3. Расчет сетевого графика;
4. Разработка проектных правил для организации автоматической сборки;
5. Расчет показателей качества кода;
6. Разработка шаблонов для генерации кода;
7. Доказательство корректности работы программы.

### **3.4. Темы лабораторных работ**

1. Применение средств динамического анализа – оценка производительности и поиск «узких мест», оценка потребления ресурсов и поиск утечек памяти;
2. Применение средства контроля качества кода на базе правил;
3. Практическое применение систем контроля качества кода на основе метрик;
4. Построение СДР и диаграммы Гантта;
5. Использование системы корпоративной памяти;
6. Практическое применение средств автоматической генерации документации по исходному коду;
7. Практическое применение систем контроля версий;
8. Использование систем управления задачами и заявками;
9. Разработка модульных тестов;
10. Автоматизация сборки.

### **3.5 Консультации**

#### *Групповые консультации по разделам дисциплины (ГК)*

1. Обсуждение материалов по кейсам раздела "Управление крупным проектом по разработке программного обеспечения"
2. Обсуждение материалов по кейсам раздела "Техника снижения трудоемкости крупного проекта по разработке программного обеспечения"

### **3.6 Тематика курсовых проектов/курсовых работ**

Курсовой проект/ работа не предусмотрены



### 3.7. Соответствие разделов дисциплины и формируемых в них компетенций

Запланированные результаты обучения по дисциплине (в соответствии с разделом 1)	Коды индикаторов	Номер раздела дисциплины (в соответствии с п.3.1)		Оценочное средство (тип и наименование)
		1	2	
<b>Знать:</b>				
современные методологии разработки программного обеспечения	ИД-1ПК-1	+		Контрольная работа/Расчет сетевого графика
современные архитектурные концепции, применяемые при разработке программных систем	ИД-3ПК-1		+	Лабораторная работа/Разработка шаблонов для генерации кода
современную технику планирования работ в проекте по разработке программного обеспечения	ИД-2ПК-2		+	Лабораторная работа/Применение средства контроля качества кода на базе правил
технику снижения трудоемкости крупного проекта по разработке программного обеспечения	ИД-3ПК-2		+	Лабораторная работа/Практическое применение систем контроля качества кода на основе метрик на примере Microsoft Visual Studio.
<b>Уметь:</b>				
применять современные CASE-средства	ИД-1ПК-1		+	Лабораторная работа/Автоматизация документирования программного продукта Лабораторная работа/Практическое применение систем контроля версий
применять типовые архитектурные решения	ИД-3ПК-1		+	/Автоматизация сборки программного проекта. Применение системы компонентной сборки. Лабораторная работа/Разработка шаблонов для генерации кода
анализировать архитектуру программных систем, оценивать уместность применения той или иной разновидности решения и технологии доступа к данным	ИД-2ПК-2		+	Лабораторная работа/Практическое применение систем контроля качества кода на основе метрик на примере Microsoft Visual Studio. Лабораторная работа/Применение средства контроля качества кода на базе правил
создавать и внедрять стандарты кодирования и	ИД-3ПК-2		+	Реферат/Расчет показателей качества кода

проектирования				
----------------	--	--	--	--

## **4. КОМПЕТЕНТНОСТНО-ОРИЕНТИРОВАННЫЕ ОЦЕНОЧНЫЕ СРЕДСТВА ДЛЯ КОНТРОЛЯ ОСВОЕНИЯ РЕЗУЛЬТАТОВ ОБУЧЕНИЯ ПО ДИСЦИПЛИНЕ (ТЕКУЩИЙ КОНТРОЛЬ УСПЕВАЕМОСТИ, ПРОМЕЖУТОЧНАЯ АТТЕСТАЦИЯ ПО ДИСЦИПЛИНЕ)**

### **4.1. Текущий контроль успеваемости**

#### **8 семестр**

Форма реализации: Компьютерное задание

1. Автоматизация документирования программного продукта (Лабораторная работа)
2. Автоматизация сборки программного проекта. Применение системы компонентной сборки. ()
3. Практическое применение систем контроля версий (Лабораторная работа)
4. Практическое применение систем контроля качества кода на основе метрик на примере Microsoft Visual Studio. (Лабораторная работа)
5. Применение средства контроля качества кода на базе правил (Лабораторная работа)
6. Разработка шаблонов для генерации кода (Лабораторная работа)

Форма реализации: Проверка задания

1. Расчет показателей качества кода (Реферат)
2. Расчет сетевого графика (Контрольная работа)

Балльно-рейтинговая структура дисциплины является приложением А.

### **4.2 Промежуточная аттестация по дисциплине**

*Экзамен (Семестр №8)*

Оценка определяется в соответствии с Положением о балльно-рейтинговой системе для студентов НИУ «МЭИ» на основании семестровой и аттестационной составляющих.

В диплом выставляется оценка за 8 семестр.

**Примечание:** Оценочные материалы по дисциплине приведены в фонде оценочных материалов ОПОП.

## **5. УЧЕБНО-МЕТОДИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ**

### **5.1 Печатные и электронные издания:**

1. Беркун, С. Искусство управления IT- проектами : пер. с англ. / С. Беркун. – 2-е изд. – СПб. : Питер, 2011. – 432 с. – (Бестселлеры O'Reilly). – ISBN 978-5-388-00543-4.;
2. Куриленко, И. Е. Современные методологии разработки программных средств : учебное пособие по курсам "CASE-технологии разработки программных средств", "Технологии разработки программных средств" по направлениям "Прикладная математика и информатика", "Информатика и вычислительная техника", "Информационные системы" / И. Е. Куриленко, Моск. энерг. ин-т (МЭИ ТУ). – М. : Издательский дом МЭИ, 2011. – 112 с. – ISBN 978-5-383-00618-4.  
<http://elibr.mpei.ru/elibr/view.php?id=2859>;
3. Куриленко, И. Е. Модульное тестирование : учебное пособие по курсам "CASE-технологии разработки программных средств", "Технологии разработки программных средств" по направлениям "Прикладная математика и информатика", "Информатика и вычислительная техника", "Информационные системы" / И. Е. Куриленко, П. Р. Варшавский, Моск. энерг. ин-т (МЭИ ТУ). – М. : Издательский дом МЭИ, 2011. – 48 с. – ISBN 978-5-383-

00632-0.

<http://elib.mpei.ru/elib/view.php?id=2858>;

4. А. А. Гайнутдинова, А. С. Брысаев- "Инновационное управление производственными программами и проектами в НГХК", Издательство: "Казанский научно-исследовательский технологический университет (КНИТУ)", Казань, 2013 - (112 с.)

<https://biblioclub.ru/index.php?page=book&id=258765>;

5. Маран М. М.- "Программная инженерия", (2-е изд., стер.), Издательство: "Лань", Санкт-Петербург, 2021 - (196 с.)

<https://e.lanbook.com/book/175503>;

6. "Руководство к своду знаний по управлению проектами (Руководства РМВОК®)", (5-е изд.), Издательство: "Олимп-Бизнес", Москва, 2018 - (613 с.)

<https://biblioclub.ru/index.php?page=book&id=494449>.

## **5.2 Лицензионное и свободно распространяемое программное обеспечение:**

1. СДО "Прометей";

2. Office / Российский пакет офисных программ;

3. Windows / Операционная система семейства Linux;

4. Видеоконференции (Майнд, Сберджаз, ВК и др);

5. Visual Studio;

6. GitHub;

7. Git;

8. Libre Office.

## **5.3 Интернет-ресурсы, включая профессиональные базы данных и информационно-справочные системы:**

1. ЭБС Лань - <https://e.lanbook.com/>

2. Электронная библиотека МЭИ (ЭБ МЭИ) - <http://elib.mpei.ru/login.php>

## **6. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ**

<b>Тип помещения</b>	<b>Номер аудитории, наименование</b>	<b>Оснащение</b>
Учебные аудитории для проведения лекционных занятий и текущего контроля	М-805, Учебная аудитория каф. "ПМИИ"	парта со скамьей, доска меловая, компьютерная сеть с выходом в Интернет, мультимедийный проектор, экран, компьютер персональный
	Ж-120, Машинный зал ИВЦ	сервер, кондиционер
Учебные аудитории для проведения практических занятий, КР и КП	М-811, Учебная аудитория	стол преподавателя, стол учебный, стул, вешалка для одежды, доска меловая, компьютерная сеть с выходом в Интернет, компьютер персональный
	Ж-120, Машинный зал ИВЦ	сервер, кондиционер
Учебные аудитории для проведения лабораторных занятий	М-706, Дисплейный класс каф. "ПМИИ"	стол преподавателя, стол компьютерный, стул, компьютерная сеть с выходом в Интернет, мультимедийный проектор, экран, компьютер персональный, кондиционер
Учебные аудитории для проведения промежуточной	М-706, Дисплейный класс каф. "ПМИИ"	стол преподавателя, стол компьютерный, стул, компьютерная сеть с выходом в Интернет,

аттестации		мультимедийный проектор, экран, компьютер персональный, кондиционер
Помещения для самостоятельной работы	НТБ-303, Лекционная аудитория	стол компьютерный, стул, стол письменный, вешалка для одежды, компьютерная сеть с выходом в Интернет, компьютер персональный, принтер, кондиционер
Помещения для консультирования	М-704, Преподавательская кафедры ПМИИ	стол, стул, шкаф, тумба, доска меловая, компьютерная сеть с выходом в Интернет, мультимедийный проектор, экран, компьютер персональный, холодильник, кондиционер
Помещения для хранения оборудования и учебного инвентаря	М-703а/1, Кладовая каф. "ПМИИ"	стеллаж для хранения книг, тумба, экран, ноутбук, книги, учебники, пособия

## БАЛЛЬНО-РЕЙТИНГОВАЯ СТРУКТУРА ДИСЦИПЛИНЫ

### Ведение крупных программных проектов

(название дисциплины)

#### 8 семестр

**Перечень контрольных мероприятий текущего контроля успеваемости по дисциплине:**

- КМ-1 Расчет сетевого графика (Контрольная работа)
- КМ-2 Практическое применение систем контроля версий (Лабораторная работа)
- КМ-3 Разработка шаблонов для генерации кода (Лабораторная работа)
- КМ-4 Автоматизация сборки программного проекта. Применение системы компонентной сборки.
- КМ-5 Расчет показателей качества кода (Реферат)
- КМ-6 Автоматизация документирования программного продукта (Лабораторная работа)
- КМ-7 Практическое применение систем контроля качества кода на основе метрик на примере Microsoft Visual Studio. (Лабораторная работа)
- КМ-8 Применение средства контроля качества кода на базе правил (Лабораторная работа)

**Вид промежуточной аттестации – Экзамен.**

Номер раздела	Раздел дисциплины	Индекс КМ:	КМ-1	КМ-2	КМ-3	КМ-4	КМ-5	КМ-6	КМ-7	КМ-8
		Неделя КМ:	2	5	6	7	8	10	12	15
1	Управление крупным проектом по разработке программного обеспечения									
1.1	Методы сетевого планирования		+							
1.2	Гибкое планирование		+							
2	Техника снижения трудоемкости крупного проекта по разработке программного обеспечения									
2.1	Системы контроля версий и конфигурационного управления.			+				+		
2.2	Автоматизация сборки.				+	+				
2.3	Генерация кода по шаблону.			+	+			+		
2.4	Контроль работоспособности кода.						+		+	+
2.5	Контроль качества.						+		+	+
2.6	Автоматизация тестирования.						+			

2.7	Автоматическое документирование и средства корпоративной памяти.							+	+
Вес КМ, %:		10	15	15	10	20	10	10	10