

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ  
Федеральное государственное автономное образовательное учреждение  
высшего образования  
«Уральский федеральный университет имени первого Президента России Б.Н. Ельцина»

Институт фундаментального образования

УТВЕРЖДАЮ  
Проректор по науке

В.В. Кружаев

«\_\_» \_\_\_\_\_ 20\_\_ г.

**РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ**

**Методология разработки программного обеспечения**

<b>Перечень сведений о рабочей программе дисциплины</b>	<b>Учетные данные</b>
<b>Образовательная программа</b> Математическое моделирование, численные методы и комплексы программ	<b>Код ОП</b> 09.06.01
<b>Направление подготовки</b> Информатика и вычислительная техника	<b>Код направления и уровня подготовки</b> 09.06.01
<b>Уровень подготовки</b> Подготовка кадров высшей квалификации	
<b>ФГОС ВО</b>	<b>Реквизиты приказа Минобрнауки РФ об утверждении ФГОС ВО:</b> от 30 июля 2014 г. №875 с изменениями и дополнениями от 30.04.2015

**СОГЛАСОВАНО**  
УПРАВЛЕНИЕ ПОДГОТОВКИ  
КАДРОВ ВЫСШЕЙ  
КВАЛИФИКАЦИИ

Екатеринбург, 2016 г.

Рабочая программа дисциплины составлена авторами:

<b>№ п/п</b>	<b>ФИО</b>	<b>Ученая степень, ученое звание</b>	<b>Должность</b>	<b>Кафедра</b>	<b>Подпись</b>
1	Берестова Светлана Александровна	Д. ф.-м. н., до- цент	Заведую- щий ка- федрой	Кафедра теорети- ческой механики	

**Рекомендовано Методическим советом УрФУ**

Председатель Методического совета УрФУ

Е.В.Вострецова

**Согласовано:**

Начальник ОПНПК

О.А. Неволлина

# **1. ОБЩАЯ ХАРАКТЕРИСТИКА ДИСЦИПЛИНЫ МЕТОДОЛОГИЯ РАЗРАБОТКИ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ**

## **1.1. Аннотация содержания дисциплины**

Методология разработки программного обеспечения является дисциплиной по выбору аспиранта. В рамках дисциплины изучаются основы разработки программного обеспечения и методологии проектирования. Рассматриваются разработка, тестирование, отладка, внедрение и сопровождение программного обеспечения вычислительной техники с использованием современных CALS-технологий и CASE-средств, а также способы внедрения информационных систем.

Для усвоения дисциплины «Методология разработки программного обеспечения» обучаемый должен обладать базовыми знаниями по основам создания информационных систем и использованию новых информационных технологий обработки информации, о жизненном цикле программного обеспечения, об объектно-ориентированном программировании, знать элементы теории сложности.

## **1.2. Язык реализации программы – русский**

## **1.3. Планируемые результаты обучения по дисциплине.**

Результатом обучения в рамках дисциплины является формирование у студента следующих компетенций:

- готовностью участвовать в работе российских и международных исследовательских коллективов по решению научных и научно-образовательных задач (УК-3);
- способностью следовать этическим нормам в профессиональной деятельности (УК-5);
- владением культурой научного исследования, в том числе с использованием новейших информационно-коммуникационных технологий (ОПК-2);
- способностью к разработке новых методов исследования и их применению в самостоятельной профессиональной научно-исследовательской деятельности в области профессиональной деятельности (ОПК-3);
- способность к интенсивной научно-исследовательской и научно-исследовательской деятельности (ПК-1);
- умение ориентироваться в современных алгоритмах компьютерной математики, совершенствовать, углублять и развивать математическую теорию, лежащую в их основе собственным видением прикладного аспекта в строгих математических формулировках (ПК-2);
- умение применять базовые модели и алгоритмы вычислительной математики к решению задач прикладного характера (ПК-3);
- способность разрабатывать, анализировать и обосновывать адекватность математических моделей (ПК-4);
- способность проводить сравнительный анализ и осуществлять обоснованный выбор алгоритмических и программно-аппаратных средств (ПК-5);
- способность понимать сущность и значение информации в развитии современного общества, применять достижения информатики и вычислительной техники, перерабатывать большие объемы информации, проводить целенаправленный поиск в различных источниках информации по профилю деятельности, в том числе в глобальных компьютерных системах (ПК-6).

В результате освоения дисциплины студент должен:

знать:

- жизненный цикл программ, методы оценки качества программных продуктов, технологии разработки программных комплексов, CASE-средства;

- методы и алгоритмы объектно-ориентированного программирования;
- методики, языки и стандарты информационной поддержки изделий (CALS-технологий) на различных этапах их жизненного цикла;

уметь:

- использовать типовые программные продукты, ориентированные на решение научных, проектных и технологических задач;
- эффективно работать в качестве члена команды по разработке программного обеспечения;

владеть:

- методиками сбора, переработки и представления научно-технических материалов по результатам исследований к опубликованию в печати, а также в виде обзоров, рефератов, отчетов, докладов и лекций;
- способностью брать на себя ответственность за результаты работы по разработке программных средств.

#### 1.4. Объем дисциплины

№ п/п	Виды учебной работы	Объем дисциплины		Распределение объема дисциплины по семестрам (час.)
		Всего часов	В т.ч. контактная работа (час.)	5
1.	<b>Аудиторные занятия</b>	<b>4</b>	<b>4,85</b>	<b>4</b>
2.	Лекции	4	4	4
3.	Практические занятия		0	0
4.	Лабораторные работы	0	0	0
5.	<b>Самостоятельная работа студентов, включая все виды текущей аттестации</b>	104	0,6	104
6.	<b>Промежуточная аттестация</b>	Зачет	0,25	Зачет
7.	<b>Общий объем по учебному плану, час.</b>	108	<b>4,85</b>	108
8.	<b>Общий объем по учебному плану, з.е.</b>	3		3

#### 2. СОДЕРЖАНИЕ ДИСЦИПЛИНЫ

Код раздела, темы	Раздел, тема Дисциплины	Содержание
P1	Сложность программного обеспечения. Жизненный цикл программного обеспечения. Выявление требований к программной системе.	Почему программному обеспечению присуща сложность. Сложность реальной предметной области, сложность описания поведения больших дискретных систем, сложность управления коллективом разработчиков. Проблемы, возникающие при общении с заказчиками программных систем. Сложность оценки качества программного обеспечения. Стандарты разработки ПО и CASE-технологии. Жизненный цикл программного обеспечения. Распределение финансовых и временных затрат на реализацию каждого из этапов разработки программного обеспечения. Обследование системы, общение с заказчиком, планирование разработки, составление технического задания.

		Детальный анализ предметной области, принятие окончательного решения о необходимости создания информационной системы, проектирование общей архитектуры системы, выбор метода проектирования.
<b>P2</b>	Обзор методологий проектирования программных продуктов. Технологии быстрой разработки программного обеспечения	Каскадные и итеративные технологии. Критичность и масштабность программных проектов. Технология экстремального программирования. SCRUM технология. Преимущества и недостатки технологий быстрой разработки программного обеспечения. Организация коллективной работы над проектом при использовании технологий быстрой разработки.
<b>P3</b>	Объектно-ориентированное проектирование программной системы. Средства информационной поддержки программных проектов и изделий (CALS) технологий	Построение объектно-ориентированной архитектуры системы. Методы объектно-ориентированного анализа для выявления классов и объектов. CASE - средства объектно-ориентированного проектирования. Средства управления проектами. Применение данных средств при разработке и сопровождении программных продуктов. Использование средств коллективного владения кодом при создании корпоративных информационных систем.
<b>P4</b>	Тестирование и отладка программных систем. Оценка качества программного обеспечения. Внедрение и сопровождение программных продуктов.	Стратегии и методы тестирования. Прямое и обратное тестирование. Программные средства автоматизации тестирования. Методики оценки качества ПО. Процессный подход к оценке качества ПО. Планирование процесса внедрения программного продукта. Основные задачи решаемые на этапе внедрения. Процесс устранения ошибок на этапе внедрения. Техническая поддержка пользователей на этапе сопровождения.

### 3. РАСПРЕДЕЛЕНИЕ УЧЕБНОГО ВРЕМЕНИ

#### 3.1. Распределение аудиторной нагрузки и мероприятий самостоятельной работы по разделам дисциплины



#### 4. ОРГАНИЗАЦИЯ ПРАКТИЧЕСКИХ ЗАНЯТИЙ, САМОСТОЯТЕЛЬНОЙ РАБОТЫ ПО ДИСЦИПЛИНЕ

##### 4.1. Лабораторные работы

*«не предусмотрено»*

##### 4.2. Практические занятия

*«не предусмотрено»*

##### 4.3. Примерная тематика самостоятельной работы

###### 4.3.1. Примерный перечень тем домашних работ

*«не предусмотрено»*

###### 4.3.2. Примерный перечень тем графических работ

*«не предусмотрено»*

###### 4.3.3. Примерный перечень тем рефератов (эссе, творческих работ)

*«не предусмотрено»*

###### 4.3.4. Примерная тематика индивидуальных или групповых проектов

*«не предусмотрено»*

###### 4.3.5. Примерный перечень тем расчетных работ (программных продуктов)

*«не предусмотрено»*

###### 4.3.6. Примерный перечень тем расчетно-графических работ

*«не предусмотрено»*

###### 4.3.7. Примерный перечень тем курсовых проектов (курсовых работ)

*«не предусмотрено»*

###### 4.3.8. Примерная тематика контрольных работ

*«не предусмотрено»*

###### 4.3.9. Примерная тематика коллоквиумов

*«не предусмотрено»*

#### 5. СООТНОШЕНИЕ РАЗДЕЛОВ, ТЕМ ДИСЦИПЛИНЫ И ПРИМЕНЯЕМЫХ ТЕХНОЛОГИЙ ОБУЧЕНИЯ

Код раздела, темы дисциплины	Активные методы обучения						Дистанционные образовательные технологии и электронное обучение					
	Проектная работа	Кейс-анализ	Деловые игры	Проблемное обучение	Командная работа	Другие (указать, какие)	Сетевые учебные курсы	Виртуальные практикумы и тренажеры	Вебинары и видеоконференции	Асинхронные web-конференции и семинары	Совместная работа и разработка контента	Другие (указать, какие)
P1			*	*								
P2		*										
P3	*		*									
P4		*	*									

## **6. ФОНД ОЦЕНОЧНЫХ СРЕДСТВ ДЛЯ ПРОВЕДЕНИЯ ТЕКУЩЕЙ И ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ ПО ДИСЦИПЛИНЕ (Приложение 1)**

## **7. УЧЕБНО-МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ**

### **7.1.Рекомендуемая литература**

#### **7.1.1. Основная литература**

1. Буч Г., Рамбо Д., Декобсон А. Язык UML. Руководство пользователя: Пер. с англ. — М.: ДМК, 2000. — 432 с.: ил.
2. С. Орлов. Технологии разработки программного обеспечения. Учебное пособие. — СПб.: Изд-во «Питер», 2003. — 480 с.
3. Мирошниченко Е.А. Технология программирования: Учебное пособие. — Томск: Изд. ТПУ, 2019. — 42 с.
4. Буч Г. Объектно-ориентированный анализ и проектирование с примерами приложений на С++, 2-е изд. / Пер. с англ. — М.: «Издательство Бином», СПб: «Невский диалект», 1998. — 560 с.: ил.
5. Фокс Дж.. Программное обеспечение и его разработка. — М.: Мир, 1989. — 360 с.
6. ИСО 9000-3: ИСО 9001 Общее руководство качеством и стандарты по обеспечению качества, часть 3: Руководящие указания по применению ИСО 9001 при разработке, поставке и обслуживанию программного. Международная организация стандартов, Женева, 1991.
7. ИСО/МЭК 9126 Информационные технологии. Оценка продукции программного обеспечения. Характеристики качества и инструкции по их применению. Международная организация стандартов, Женева, 1991.

#### **7.1.2. Дополнительная литература**

1. Коннолли Т. Базы данных: проектирование, реализация и сопровождение. Теория и практика: [пер. с англ.] / Т. Коннолли, К. Бегг,

### **7.2. Методические разработки**

*«не используются»*

### **7.3.Программное обеспечение**

1. Операционные системы семейства Windows или свободно-распространяемые ОС Linux, ОС мобильных устройств.
2. Microsoft Office 2007 или более новая версия
3. Интегрированные среды разработки программного обеспечения (MS Visual Studio, IntelliJ IDEA и др.)

### **7.4. Базы данных, информационно-справочные и поисковые системы**

- Электронные журналы издательства Taylor&Francis (компания Metapress) на английском языке (<http://www.tandfonline.com>).
- Academic Search Complete (<http://search.ebscohost.com>).
- Oxford University Press (<http://www.oxfordjournals.org/en/>).
- Wiley Online Library (<http://pubs.acs.org/>).
- Web of Science (<http://apps.webofknowledge.com/>).
- IEEE Xplore, Institute of Electric and Electronic Engineers (IEEE) (<http://www.ieee.org/ieeexplore>).

- ООО Научная электронная библиотека(<http://elibrary.ru>).
- Oxford University Press (<http://www.oxfordjournals.org/en/>).
- ScienceDirect Freedom Collection (<http://www.sciencedirect.com/>).
- Scopus (<http://www.scopus.com/>).
- Springer Materials (<http://materials.springer.com/>).

#### **7.5. Электронные образовательные ресурсы**

- Зональная научная библиотека <http://lib.urfu.ru/>
- Каталоги библиотеки <http://lib.urfu.ru/about/department/catalog/rescatalog/>
- Электронный каталог <http://lib.urfu.ru/resources/ec/>
- Ресурсы <http://lib.urfu.ru/resources>
- Поиск <http://lib.urfu.ru/search>.

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

*Для освоения дисциплины "Методология разработки программного обеспечения" необходимо использовать компьютерную аудиторию с установленным офисным пакетом (например Microsoft Office 2007 или более новая версия или альтернативные пакеты) и интегрированные среды разработки программного обеспечения (MS Visual Studio, Intel-ij IDEA и др.).*

## 6. ФОНД ОЦЕНОЧНЫХ СРЕДСТВ ДЛЯ ПРОВЕДЕНИЯ ТЕКУЩЕЙ И ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ ПО ДИСЦИПЛИНЕ

### 6.1. ОЦЕНОЧНЫЕ СРЕДСТВА ДЛЯ ПРОВЕДЕНИЯ ТЕКУЩЕЙ И ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ

**6.1.1. Примерные задания для проведения мини-контрольных в рамках учебных занятий**  
*«не предусмотрено»*

**6.1.2. Примерные контрольные задачи в рамках учебных занятий**  
*«не предусмотрено»*

**6.1.3. Примерные контрольные кейсы**  
*«не предусмотрено»*

**6.1.4. Перечень примерных вопросов для зачета**

1. Что такое промышленный программный продукт. Дать определения пакета прикладных программ, программной системы.
2. Жизненный цикл программного обеспечения. Дать краткую характеристику каждого этапа.
3. Почему программные системы сложны. Привести пять признаков сложной системы.
4. Техническое задание. Перечислить и охарактеризовать разделы, входящие в техническое задание.
5. Унифицированный процесс разработки программного обеспечения. Жизненный цикл унифицированного процесса.
6. Работа с кадрами. Перечислить роли разработчиков и дать характеристику каждой из них.
7. Дать определения проекта, процесса, продукта с точки зрения унифицированного процесса разработки программного обеспечения.
8. Что такое артефакт. В чем преимущества организованного процесса разработки программного обеспечения.
9. Использование языка UML при проектировании сложных программных систем. Какие диаграммы используются в UML для создания моделей программной системы.
10. Диаграмма вариантов использования, ее назначение. Рассказать о варианте использования и действующем лице. Правила построения диаграммы вариантов использования.
11. Понятие класса и объекта. Что может быть объектом. Что такое атрибут и операция.
12. Пять критериев проверки правильности построения класса.
13. Что такое классификация с точки зрения объектно-ориентированного проектирования программных систем. Теории классификации.
14. Методы классификации.
15. Микропроцесс проектирования. Перечислить этапы и основные виды деятельности, выполняемые на каждом из них.
16. Микропроцесс проектирования – первый этап.
17. Микропроцесс проектирования – второй этап.
18. Микропроцесс проектирования – третий этап.
19. Микропроцесс проектирования – четвертый этап.
20. Диаграммы взаимодействия. Основное назначение.
21. Диаграмма классов. Ее назначение. Что она включает. Рассказать об основных видах связей между классами.
22. Дать определение тестированию и отладке. Особенности и объекты тестирования. Автономное и комплексное тестирование.
23. Дать определение тестированию и отладке. Направления тестирования. Стратегия тестирования. Контрольный лист тестирования модуля.

24. Дать определение тестированию и отладке. Локализация ошибок. Классификация ошибок. Безопасное программирование.
25. Оценки ошибок.
26. Документирование. Состав и содержание документов, прилагаемых к программной системе.
27. Внедрение программного комплекса. Планирование испытаний.
28. Внедрение программного комплекса. Подготовка тестовых данных. Анализ результатов испытаний.
29. Что такое качество с точки зрения квалиметрии. Дать определение свойству и показателю качества ПО. Основные задачи решаемые при оценке качества.
30. Оценка качества программного обеспечения. Методы оценки свойств программного обеспечения.

#### **6.1.5. Перечень примерных вопросов для экзамена**

*«не предусмотрено»*