

Федеральное государственное бюджетное образовательное учреждение
высшего образования
Московский государственный университет имени М.В. Ломоносова
Факультет вычислительной математики и кибернетики

УТВЕРЖДАЮ
Декан факультета ВМК МГУ
/И.А.Соколов/
_____ 2023 г.



РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ (МОДУЛЯ)

Наименование дисциплины (модуля):

«Программная инженерия»

Уровень высшего образования:

бакалавриат

Направление подготовки (специальность):

02.03.02 Фундаментальная информатика и информационные технологии

Направленность (профиль) ОПОП:

дисциплина относится к базовой части программы

Форма обучения:

очная

Москва 2023

Рабочая программа дисциплины (модуля) разработана в соответствии с самостоятельно установленным МГУ образовательным стандартом (ОС МГУ) для реализуемых основных профессиональных образовательных программ бакалавриата, реализуемых последовательно по схеме интегрированной подготовки по направлениям 02.03.02, 02.04.02 «Фундаментальная информатика и информационные технологии» в редакции приказа МГУ от 30 декабря 2016 г.

1. Дисциплина относится к базовой части ОПОП ВО.

2. Входные требования для освоения дисциплины (модуля): учащиеся должны владеть знаниями по алгоритмам, алгоритмическим языкам и программированию в объеме, соответствующем программе второго года обучения основных образовательных программ бакалавриата по укрупненному группам направлений и специальностей 01.00.00 «Математика и механика», 02.00.00 «Компьютерные и информационные науки».

3. Результаты обучения по дисциплине (модулю), соотнесенные с требуемыми компетенциями выпускников.

Компетенции выпускников, частично формируемые при реализации дисциплины (модуля):

- **ОПК-3** Способность к разработке алгоритмических и программных решений в области системного и прикладного программирования, математических, информационных и имитационных моделей, созданию информационных ресурсов глобальных сетей, образовательного контента, прикладных баз данных, тестов и средств тестирования систем и средств на соответствие стандартам и исходным требованиям
- **ПК-1** Способность собирать, обрабатывать и интерпретировать данные современных научных исследований, необходимые для формирования выводов по соответствующим научным исследованиям
- **ПК-7** Способность разрабатывать и реализовывать процессы жизненного цикла информационных систем, программного обеспечения, сервисов систем информационных технологий, а также методы и механизмы оценки и анализа функционирования средств и систем информационных технологий

Планируемые результаты обучения по дисциплине (модулю):

Знать:

1. основные понятия и принципы программной инженерии;
2. основные модели жизненного цикла программного обеспечения (ПО);
3. основные виды деятельности при разработке, развитии и сопровождении ПО;
4. задачи и техники управления проектами по разработке и сопровождении ПО;
5. основные методы выделения, анализа и описания требований к ПО;
6. основные методы проектирования, анализа и описания архитектуры ПО;
7. основные методы контроля качества ПО;
8. основы компонентных технологий разработки ПО.

Уметь:

1. определять, описывать и анализировать требования к ПО;
2. описывать и анализировать архитектуру ПО;
3. выделять и применять образцы проектирования ПО;
4. создавать и анализировать тестовые наборы для ПО;
5. анализировать удобство использования ПО.

Владеть:

1. навыками применения современных техник разработки и анализа ПО.

4. Формат обучения: лекции и семинарские занятия проводятся с использованием меловой доски и презентаций.

5. Объем дисциплины (модуля) составляет 2 з.е., в том числе 36 академических часов, отведенных на контактную работу обучающихся с преподавателем, 36 академических часа на самостоятельную работу обучающихся.

6. Содержание дисциплины (модуля), структурированное по темам (разделам) с указанием отведенного на них количества академических часов и виды учебных занятий.

Наименование и краткое содержание разделов и тем дисциплины (модуля), Форма промежуточной аттестации по дисциплине (модулю)	Всего (часы)	В том числе			Самостоятельная работа обучающихся, часы
		Контактная работа (работа во взаимодействии с преподавателем) Виды контактной работы, часы			
		Занятия лекционного типа*	Занятия семинарского типа*	Всего	
1. Предмет и основные принципы программной инженерии	4	2	0	2	2
2. Жизненный цикл программного обеспечения и процессы его разработки	8	2	2	4	4

3. Управление разработкой программного обеспечения	10	4	2	6	4
4. Анализ требований к программному обеспечению	10	4	2	6	4
5. Качество программного обеспечения и методы его контроля	5	2	1	3	2
6. Текущий контроль успеваемости: контрольная работа № 1	4	0	2	2	2
7. Архитектура программного обеспечения и ее проектирование	9	4	1	5	4
8. Удобство использования программного обеспечения	7	2	1	3	4
9. Компонентные технологии разработки распределенных программных систем	9	4	1	5	4
Промежуточная аттестация: зачет	6	0	0	0	6
Итого	72	24	12	36	36

7. Фонд оценочных средств (ФОС) для оценивания результатов обучения по дисциплине (модулю)

7.1. Типовые контрольные задания или иные материалы для проведения текущего контроля успеваемости.

Контрольная работа № 1		
1. Заполните следующую таблицу, указав в соответствующих столбцах номера перечисленных ниже характеристик более свойственных сложным программным системам, простым программам, а также тех, которые не связаны со сложностью ПО.		
Более свойственно сложным программам	Более свойственно простым программам	Не связано с тем, сложная программа или нет
1. Исходный код, написанный на языке С		
2. Необходимость разработки архитектуры до перехода к кодированию		
3. Много людей вовлечено в создание системы		
4. Отсутствие поддержки пользователей		
5. Высокая стоимость разработки		
2. Обозначим различные виды деятельности при разработке ПО следующим образом.		
А. Анализ требований		
В. Проектирование		
С. Кодирование		

- D. Тестирование
- E. Развертывание
- F. Эксплуатация

Поставьте галочки в первом столбце приведенной ниже таблицы рядом с последовательностями выполнения этих видов деятельности, которые могут возникнуть при использовании итеративной модели жизненного цикла ПО.

	ACBVCBABCDEF
	ABABABCDEF
	ABCDEFCDDEF
	BACDBCDEF
	ABABCABCDEF
	ABCBADEF
	ABCABABCDEF
	ABCDEABDF

3. Заполните следующую таблицу, поместив

в **первый столбец** номера перечисленных ниже техник, используемых при разработке ПО в соответствии с Rational Unified Process (RUP), в **второй столбец** — номера техник, используемых в Extreme Programming (XP), в **третий столбец** — номера техник, используемых в обоих видах процессов, в **четвертый столбец** — номера техник, не используемых ни в одном из этих двух процессов

Техники, используемые в RUP	Техники, используемые в XP	Техники, используемые и в RUP, и в XP	Техники, не используемые ни в RUP, ни в XP

1. Использование наиболее простых решений
2. Использование компонентной архитектуры
3. Фиксация требований в виде вариантов использования
4. Совместные инспекции кода
5. Коллективное владение кодом
6. Нацеленность на создание продукта, работоспособного в реальном окружении
7. Использование моделей как основного средства коммуникации

4 Отметьте те из указанных элементов, которые входят в технику «программирование парами».

Разработка кода выполняется парой программистов, один из которых пишет код, другой в это время отдыхает.
Разработка кода выполняется парой программистов, один из которых пишет код, другой в это время дает ему советы и думает над улучшением кода.
Пара программистов в течение всего проекта работает вместе.
Объединение программистов в пары меняется от задачи к задаче.
Разработка кода выполняется парой программистов, один из которых отвечает за все, сделанное ими, и руководит работой другого, а тот пишет код.
Разработка кода выполняется парой программистов попеременно, то одним, то другим, они меняются местами каждые 5-10 минут.

5. Занесите номера приведенных ниже описаний проблем предметной области в первый столбец таблицы, функций ПО — во второй столбец, требований к ПО — в третий.

Проблемы предметной области	Функции ПО	Требования к ПО

1. ПО анализа геологических данных должно повысить точность определения положения рудных тел и снизить ошибки определения их состава.
2. Web-сайт компании должен выдерживать нагрузки до 50000 запросов в час и должен быть недоступен не более 10-ти минут в год.
3. Система контроля городского движения должна существенно снизить количество аварий.
4. Пользователь-аналитик должен уже в первый день работы с системой понимать, как с ее помощью можно получить все виды годовых, квартальных и месячных отчетов о финансовой деятельности компании.
5. ПО контроля процесса производства должно автоматически отсеивать бракованные детали.
6. Пользователь должен иметь возможность установить размер шрифта всех выделенных символов равным любому целому числу пунктов, от 5 до 72.

6 К каким из характеристик качества ПО по ISO 9126 относятся описанные ошибки? Перечислите в первом столбце каждой строки с описанием ошибки номера соответствующих характеристик качества из предыдущего задания.

	Внесение первого же изменения, которое было затребовано пользователями, привело к кардинальной перестройке архитектуры системы.
	Код системы не содержит комментариев, плохо отформатирован и труден для восприятия.
	Для выполнения наиболее часто выполняемой пользователями операции системы — получения аналитического отчета за

некоторый временной отрезок — требуется нажать не менее 6-ти кнопок на 4-х диалогах.
Для установки системы необходимо разархивировать архив, содержащий установочный пакет, в директорию, находящуюся в корне диска C:, создать в реестре системы 5 новых записей и занести туда конфигурационные данные системы, создать .dat файл в установочной директории и записать туда в определенном формате конфигурацию драйвера видеокарты.
Система размером около 50000 строк состоит из одного класса, который имеет только один public метод, все остальные 863 метода в этом классе — private.

Вопросы к зачету.

1. Понятие программной системы.
2. Предмет и основные принципы программной инженерии. Свойства правильно спроектированных интерфейсов: адекватность, полнота, простота, минимальность.
3. Жизненный цикл ПО. Виды деятельности по разработке и сопровождению ПО. Основные модели жизненного цикла ПО: водопадная, итеративная, спиральная.
4. Стандарты на технологические процессы разработки и сопровождения ПО. Стандарты ISO 12207 и CMMI.
5. Примеры процессов разработки ПО. Унифицированный процесс разработки (RUP). Экстремальное программирование (XP).
6. Требования к ПО. Уровни абстракции требований: потребности, функции и детальные требования. Характеристики требований по стандарту IEEE 830.
7. Характеристики и атрибуты качества ПО по стандарту ISO 9126. Ошибки в ПО.
8. Методы контроля качества ПО. Тестирование ПО. Виды тестирования и характеристики тестов.
9. Архитектура ПО. Основные архитектурные стили: конвейер, интерактивные системы, вызов-возврат, хранилища данных, интерпретация. Анализ архитектуры с помощью метода SAAM.
10. Графические нотации, используемые при разработке ПО. Диаграммы потоков данных, диаграммы сущностей и связей, основы языка UML.
11. Образцы проектирования ПО. Образцы анализа. Идиомы. Образцы организации работ.
12. Удобство использования ПО. Принципы организации удобного интерфейса и типичные проблемы удобства ПО.
13. Понятие распределенной программной системы. Основные характеристики распределенных систем.
14. Технологии разработки Web-приложений на основе J2EE и .NET.
15. Планирование проектов по разработке ПО. Диаграммы задач и зависимостей. Выделение критического пути на графе задач.

ШКАЛА И КРИТЕРИИ ОЦЕНИВАНИЯ результатов обучения (РО) по дисциплине (модулю)

Оценка РО и соответствующие виды оценочных средств	2	3	4	5
Знания <i>Контрольная работа, зачет</i>	Отсутствие знаний	Фрагментарные знания	Общие, но не структурированные знания	Сформированные систематические знания
Умения <i>Контрольная работа, зачет</i>	Отсутствие умений	В целом успешное, но не систематическое умение	В целом успешное, но содержащее отдельные пробелы умение (небольшие неточности)	Успешное и систематическое умение
Навыки (владения, опыт деятельности) <i>Контрольная работа, зачет</i>	Отсутствие навыков (владений, опыта)	Наличие отдельных навыков (наличие фрагментарного опыта)	В целом, сформированные навыки (владения), но используемые не в активной форме	Сформированные навыки (владения), применяемые при решении задач

Соответствие результатов обучения и компетенций, в развитии которых участвует дисциплина (модуль)

Результаты обучения	Компетенция, с частичным формированием которой связано достижение результата обучения
<p>Знать:</p> <ol style="list-style-type: none"> 1. основные виды деятельности при разработке, развитии и сопровождении ПО; 2. задачи и техники управления проектами по разработке и сопровождению ПО; 3. основные методы выделения, анализа и описания требований к ПО; 4. основные методы проектирования, анализа и описания архитектуры ПО; 5. основные методы контроля качества ПО; 6. основы компонентных технологий разработки ПО. <p>Уметь:</p> <ol style="list-style-type: none"> 1. определять, описывать и анализировать требования к ПО; 2. описывать и анализировать архитектуру ПО; 3. выделять и применять образцы проектирования ПО; 	ОПК-3

	<p>4. создавать и анализировать тестовые наборы для ПО; 5. анализировать удобство использования ПО.</p> <p>Владеть:</p> <ol style="list-style-type: none"> 1. навыками применения современных техник разработки и анализа ПО. 	
<p style="text-align: center;">ПК-1</p>	<p>Знать:</p> <ol style="list-style-type: none"> 1. основные понятия и принципы программной инженерии; 2. основные модели жизненного цикла программного обеспечения (ПО); 3. основные виды деятельности при разработке, развитии и сопровождении ПО; 4. задачи и техники управления проектами по разработке и сопровождении ПО; 5. основные методы выделения, анализа и описания требований к ПО; 6. основные методы проектирования, анализа и описания архитектуры ПО; 7. основные методы контроля качества ПО; 8. основы компонентных технологий разработки ПО. <p>Уметь:</p> <ol style="list-style-type: none"> 1. определять, описывать и анализировать требования к ПО; 2. описывать и анализировать архитектуру ПО; 3. анализировать удобство использования ПО. 	
<p style="text-align: center;">ПК-7</p>	<p>Знать:</p> <ol style="list-style-type: none"> 1. основные виды деятельности при разработке, развитии и сопровождении ПО; 2. задачи и техники управления проектами по разработке и сопровождении ПО; 3. основные методы выделения, анализа и описания требований к ПО; 4. основные методы проектирования, анализа и описания архитектуры ПО; 5. основные методы контроля качества ПО; 6. основы компонентных технологий разработки ПО. <p>Уметь:</p> <ol style="list-style-type: none"> 1. определять, описывать и анализировать требования к ПО; 2. описывать и анализировать архитектуру ПО; 3. выделять и применять образцы проектирования ПО; 4. создавать и анализировать тестовые наборы для ПО; 5. анализировать удобство использования ПО. <p>Владеть:</p> <ol style="list-style-type: none"> 1. навыками применения современных техник разработки и анализа ПО. 	

8. Ресурсное обеспечение:

Основная литература:

1. И. Соммервилл. Инженерия программного обеспечения. М.: Вильямс, 2002.
2. Л. Басс, П. Клементс, Р. Кацман. Архитектура программного обеспечения на практике. СПб.: Питер, 2006.
3. Л. Константайн, Л. Локвуд. Разработка программного обеспечения. СПб.: Питер, 2004.
4. Е. А. Жоголев. Лекции по технологии программирования: Учебное пособие. М.: Издательский отдел факультета ВМК МГУ, 2001.
5. А. Коберн. Современные методы описания требований к системам. М.: Лори, 2002.
6. Г. Буч, Дж. Рамбо, А. Джекобсон. Язык UML. Руководство пользователя. М.: ДМК, 2000.
7. А. Якобсон, Г. Буч, Дж. Рамбо. Унифицированный процесс разработки программного обеспечения. СПб.: Питер, 2002.
8. К. Бек. Экстремальное программирование. СПб.: Питер, 2002.
9. Э. Хант, Д. Томас. Программист-прагматик. М.: Лори, 2004.
10. Э. Гамма, Р. Хелм, Р. Джонсон, Дж. Влиссидес. Приемы объектно-ориентированного проектирования. Паттерны проектирования. СПб.: Питер-ДМК, 2001.
11. Р. Монсон-Хейфел. Enterprise JavaBeans. СПб.: Символ-Плюс, 2002.
12. Р. Андерсон, Б. Френсис, А. Хомер, Р. Хоурд, Д. Сасмэн, К. Уотсон. ASP.NET 1.0 для профессионалов. М.: Лори, 2004.
13. Ф. Брукс. Мифический человеко-месяц или Как создаются программные системы. СПб.: Символ-Плюс, 2001.
14. Т. Демарко, Т. Листер. Человеческий фактор: успешные проекты и команды. СПб.: Символ-Плюс, 2005.
15. В. В. Кулямин. Технологии программирования. Компонентный подход. М.: Интернет-университет информационных технологий, Бином, 2007.

Дополнительная литература:

1. Software Engineering Body of Knowledge, 2005.
2. Э. Дж. Брауде. Технологии разработки программного обеспечения. СПб.: Питер, 2004.
3. У. Ройс. Управление проектами по созданию программного обеспечения. М.: Лори, 2002.
4. Д. Леффингуэлл, Д. Уидриг. Принципы работы с требованиями к программному обеспечению. Унифицированный подход. М.: Вильямс, 2002.
5. В. В. Липаев. Методы обеспечения качества крупномасштабных программных средств. М., Синтег, 2003.
6. М. Fowler. Analysis Patterns: Reusable Object Models. Addison-Wesley, 1997.
7. М. Фаулер и др. Архитектура корпоративных программных приложений. М.: Вильямс, 2004.
8. Б. Майер. Объектно-ориентированное программирование. Концепции разработки. М.: Русская редакция, 2004.
9. Дж. Рамбо, А. Якобсон, Г. Буч. UML: Специальный справочник. СПб.: Питер, 2002.

10. Г. Буч. Объектно-ориентированный анализ и проектирование с примерами приложений на С++. Второе издание. М.: Бином, СПб.: Невский диалект, 2000.
11. Б. У. Перри. Java сервлеты и JSP: сборник рецептов. М.: Кудиц-Образ, 2005.
12. Х. Дейтел, П. Дейтел, С. Сантри. Технологии программирования на Java 2. Книга 3: Корпоративные системы, сервлеты, JSP, Web-сервисы. М.: Бином, 2003.
13. А. Купер. Психбольница в руках пациентов. СПб.: Символ-Плюс, 2004

Информационные справочные системы:

1. Software Engineering Body of Knowledge, 2005. http://www.swebok.org/ironman/pdf/SWEBOK_Guide_2004.pdf.
2. Документация по платформе J2EE, версии 5. <http://java.sun.com/j2ee/5.0/index.jsp>.
3. Документация по платформе .NET. <http://msdn.microsoft.com/library/default.asp>.
4. Веб-страница курса <http://sites.google.com/view/msusecourse>

Материально-техническое обеспечение: аудитория с партами и меловой доской.

9. Язык преподавания - русский.
10. Преподаватели: Доцент кафедры системного программирования факультета ВМК МГУ В. В. Кулямин.
11. Авторы программы: Доцент кафедры системного программирования факультета ВМК МГУ В. В. Кулямин.