

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

УТВЕРЖДАЮ
декан факультета вычислительной
математики и кибернетики


М.А. Соколов /
«27» сентября 2023г.

ФОНД ОЦЕНОЧНЫХ СРЕДСТВ

по дисциплине

Основы программной инженерии

Уровень высшего образования:

бакалавриат

Направление подготовки / специальность:

02.03.02 "Фундаментальная информатика и информационные технологии" (3++)

Направленность (профиль) ОПОП:

Искусственный интеллект и анализ данных

Форма обучения:

очная

Рассмотрен и утвержден

на заседании Ученого совета факультета ВМК

(протокол №7, от 27 сентября 2023 года)

Москва 2023

1. ФОРМЫ И ОЦЕНОЧНЫЕ МАТЕРИАЛЫ ТЕКУЩЕГО КОНТРОЛЯ УСПЕВАЕМОСТИ И ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ

В процессе и по завершении изучения дисциплины оценивается формирование у студентов следующих компетенций:

Планируемые результаты обучения по дисциплине (модулю)		
Содержание и код компетенции.	Индикатор (показатель) достижения компетенции	Планируемые результаты обучения по дисциплине, сопряженные с индикаторами достижения компетенций
ОПК-6. Способен понимать принципы работы современных информационных технологий и использовать их для решения задач профессиональной деятельности	ОПК-6.1. Знает принципы работы современных информационных технологий ОПК-6.2. Использует современные информационные технологии для решения задач профессиональной деятельности	<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. выделять и применять образцы проектирования ПО; 4. создавать и анализировать тестовые наборы для ПО; 5. анализировать удобство использования ПО. <p>Владеть:</p>

		1. навыками применения современных техник разработки и анализа ПО.
--	--	--

1.1. Текущий контроль успеваемости

Текущий контроль успеваемости осуществляется путем оценки результатов выполнения заданий практических (семинарских) занятий, самостоятельной работы, предусмотренных учебным планом и посещения занятий/активность на занятиях.

В качестве оценочных средств текущего контроля успеваемости предусмотрены:

контрольная работа

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.
- A. Анализ требований
 - B. Проектирование
 - C. Кодирование
 - D. Тестирование
 - E. Развертывание
 - F. Эксплуатация

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

	ACBCBABCDEF
	ABABABCDEF
	ABCDEF CDEF
	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),
- во 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».

<input type="checkbox"/>	Разработка кода выполняется парой программистов, один из которых пишет код, другой в это время отдыхает.
<input type="checkbox"/>	Разработка кода выполняется парой программистов, один из которых пишет код, другой в это время дает ему советы и думает над улучшением кода.
<input type="checkbox"/>	Пара программистов в течение всего проекта работает вместе.
<input type="checkbox"/>	Объединение программистов в пары меняется от задачи к задаче.
<input type="checkbox"/>	Разработка кода выполняется парой программистов, один из которых отвечает за все, сделанное ими, и руководит работой другого, а тот пишет код.
<input type="checkbox"/>	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.

<input type="checkbox"/>	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.

1.2. Промежуточная аттестация

Промежуточная аттестация осуществляется в форме зачет

В качестве средств, используемых на промежуточной аттестации предусматривается:

Билеты

1.3. Типовые задания для проведения промежуточной аттестации

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 СММІ.
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.

2. КРИТЕРИИ ОЦЕНКИ ПО ДИСЦИПЛИНЕ

ШКАЛА И КРИТЕРИИ ОЦЕНИВАНИЯ результатов обучения (РО) по дисциплине				
Оценка	2 (не зачтено)	3 (зачтено)	4 (зачтено)	5 (зачтено)
виды оценочных средств				
Знания (виды оценочных средств: приведены в п. 1.2.)	Отсутствие знаний	Фрагментарные знания	Общие, но не структурированные знания	Сформированные систематические знания
Умения (виды оценочных средств: приведены в п. 1.2.)	Отсутствие умений	В целом успешное, но не систематическое умение	В целом успешное, но содержащее отдельные пробелы умение (допускает неточности не принципиального характера)	Успешное и систематическое умение
Навыки (владения, опыт деятельности) (виды оценочных средств: приведены в п. 1.2..)	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