

Министерство науки и высшего образования Российской Федерации
НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
ТОМСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ (НИ ТГУ)

Институт прикладной математики и компьютерных наук

УТВЕРЖДАЮ:
Директор



А. В. Замятин

20 22 г.

Рабочая программа дисциплины

Рефакторинг и обратное проектирование

по направлению подготовки

02.04.02 Фундаментальная информатика и информационные технологии

Направленность (профиль) подготовки :
Моделирование систем искусственного интеллекта

Форма обучения

Очная

Квалификация

Магистр

Год приема

2022

Код дисциплины в учебном плане: Б1.В.02.02

СОГЛАСОВАНО:

Руководитель ОП

 А.Н. Моисеев

Председатель УМК

 С.П. Сущенко

Томск – 2022

1. Цель и планируемые результаты освоения дисциплины

Целью освоения дисциплины является формирование следующих компетенций:

ПК-1 – способность проектировать программное обеспечение.

ПК-2 – способность создавать архитектурный проект программного средства.

Результатами освоения дисциплины являются следующие индикаторы достижения компетенций:

ИПК-1.1 Использует существующие типовые решения и шаблоны проектирования программного обеспечения.

ИПК-1.2 Применяет методы и средства проектирования программного обеспечения, структур данных, баз данных, программных интерфейсов.

ИПК-1.3 Знает методы и средства проектирования программного обеспечения, методы и средства проектирования баз данных.

ИПК-2.1 Оценивает возможность создания архитектурного проекта программного средства.

ИПК-2.2 Определяет цели архитектуры программного средства.

ИПК-2.3 Определяет ключевые сценарии для архитектуры программного средства.

2. Задачи освоения дисциплины

– Освоить аппарат, применяемый при оценке качества исходного кода программного обеспечения.

– Освоить методы устранения недостатков исходного кода.

– Научиться создавать автоматические тесты для проверки работоспособности исходного кода.

– Научиться применять понятийный аппарат и инструменты для поиска и устранения недостатков исходного кода для решения практических задач профессиональной деятельности.

3. Место дисциплины в структуре образовательной программы

Дисциплина относится к части образовательной программы, формируемой участниками образовательных отношений, предлагается обучающимся на выбор. Дисциплина входит в модуль Б1.В.02 Специализация.

4. Семестр(ы) освоения и форма(ы) промежуточной аттестации по дисциплине

Второй семестр, экзамен.

5. Входные требования для освоения дисциплины

Для успешного освоения дисциплины требуются результаты обучения по следующим дисциплинам: Введение в программную инженерию.

6. Язык реализации

Русский

7. Объем дисциплины

Общая трудоемкость дисциплины составляет 3 з.е., 108 часов, из которых:

-лекции: 16 ч.

-практические занятия: 16 ч.

в том числе практическая подготовка: 16 ч.

Объем самостоятельной работы студента определен учебным планом.

8. Содержание дисциплины, структурированное по темам

Тема 1. Качество программного обеспечения.

Введение. Признаки качественного программного обеспечения. Антипаттерны проектирования. Недостатки исходного кода.

Тема 2. Тестирование программного обеспечения.

Виды тестов. Основные принципы создания модульных тестов. Роль тестирования в процессе рефакторинга.

Тема 3. Рефакторинг программного обеспечения.

Роль рефакторинга в жизненном цикле программного обеспечения. Методы рефакторинга исходного кода.

9. Текущий контроль по дисциплине

Текущий контроль по дисциплине проводится путем контроля посещаемости, выполнения заданий индивидуального проекта и фиксируется в форме контрольной точки не менее одного раза в семестр. В результате оценки результатов выполнения заданий выставляются баллы, сумма баллов за семестр преобразуется в оценку за дисциплину.

10. Порядок проведения и критерии оценивания промежуточной аттестации

Экзамен во втором семестре проводится в форме проверки результатов выполнения индивидуального проекта. Проект состоит из трех частей. Продолжительность экзамена 1,5 часа.

Первая часть индивидуального проекта заключается в проведении анализа исходного кода проекта с целью выявления недостатков проектирования и реализации. Данная работа проверяет ИПК-2.1, ИПК-2.2.

Вторая часть заключается в создании модульных тестов для проанализированного исходного кода. Данная работа проверяет ИПК-1.2, ИПК-2.3.

Третья часть заключается в проведении процедуры рефакторинга для устранения недостатков, найденных в результате выполнения первой части индивидуального проекта. Данная работа проверяет ИПК-1.1, ИПК-1.3.

Примеры задач индивидуального проекта:

1. Задача 1.

Требуется: провести обратное проектирование проекта:

1. Составить диаграмму пакетов.

2. Составить диаграммы классов частей проекта с наибольшим количеством недостатков.

3. Необходимо провести анализ исходного кода приложения с целью обнаружения недостатков кода.

2. Задача 2.

Требуется: покрыть Unit-тестами недостатки кода, обнаруженные при выполнении работы №1.

3. Задача 3.

Требуется: провести рефакторинг недостатков, которые были обнаружены при выполнении работы №1 и/или покрыты тестами при выполнении работы №2.

Результаты экзамена определяются оценками «отлично», «хорошо», «удовлетворительно», «неудовлетворительно».

Итоговая оценка за дисциплину формируется переводом набранных за семестр баллов в пятибалльную систему. Максимальное число баллов 30 (соответствует оценке «отлично»). 10 баллов соответствуют оценке «удовлетворительно», 20 баллов соответствуют оценке «хорошо».

11. Учебно-методическое обеспечение

а) Электронный учебный курс по дисциплине в электронном университете «Moodle».

б) Оценочные материалы текущего контроля и промежуточной аттестации по дисциплине.

12. Перечень учебной литературы и ресурсов сети Интернет

а) основная литература:

1. Макконнелл С. Совершенный код. Мастер-класс. / Пер. с англ. – М. «Русская редакция», 2014. – 896 с.

2. Мартин Р. Чистый код: создание, анализ и рефакторинг. Библиотека программиста. – СПб.: Питер, 2016. – 464 с. Паттерны проектирования. – СПб.: Питер, 2016.

3. Кириевски, Д. Рефакторинг с использованием шаблонов. : Пер. с англ. – М.: ООО «И.Д.Вильямс», 2016. – 400 с.

4. Фаулер М., Бек К., Брант Д., Опдак У., Робертс Д. Рефакторинг: улучшение проекта существующего кода: Пер. с англ. – СПб.: ООО «Альфа-книга», 2017. – 448 с.

б) дополнительная литература:

– Мод Л. Масштабируемый рефакторинг. Возвращаем контроль над кодом. – СПб.: Питер, 2022. – 256 с.

Гамма Э., Хелм Р., Джонсон Р., Влссидес Дж. Приемы объектно-ориентированного проектирования. Паттерны проектирования. – СПб.: Питер, 2012. – 368 с.

в) ресурсы сети Интернет:

– открытые онлайн-курсы

13. Перечень информационных технологий

а) лицензионное и свободно распространяемое программное обеспечение:

– публично доступные облачные технологии (Google Docs, Яндекс диск и т.п.).

14. Материально-техническое обеспечение

Аудитории для проведения занятий лекционного типа.

Аудитории для проведения занятий семинарского типа, индивидуальных и групповых консультаций, текущего контроля и промежуточной аттестации.

Помещения для самостоятельной работы, оснащенные компьютерной техникой и доступом к сети Интернет, в электронную информационно-образовательную среду и к информационным справочным системам.

15. Информация о разработчиках

Иванова Лидия Сергеевна, ассистент кафедры программной инженерии НИ ТГУ.