

**Федеральное государственное автономное  
образовательное учреждение высшего образования  
«Национальный исследовательский Нижегородский государственный университет  
им. Н.И. Лобачевского»**

Институт информационных технологий, математики и механики  

---

(факультет / институт / филиал)

УТВЕРЖДЕНО  
решением Ученого совета ННГУ  
протокол  
№13 от 30.11.2022 г.

**Рабочая программа дисциплины**

Компоновка программ, шаблоны  

---

(наименование дисциплины (модуля))

Уровень высшего образования  
**бакалавриат**  

---

(бакалавриат / магистратура / специалитет)

Направление подготовки / специальность  
**09.03.03 «Прикладная информатика»**  

---

(указывается код и наименование направления подготовки / специальности)

Направленность образовательной программы  
**Прикладная информатика в области обработки данных**  

---

(указывается профиль / магистерская программа / специализация)

Форма обучения  
**Очно-заочная**  

---

(очная / очно-заочная / заочная)

Нижегород

2023

## 1. Место дисциплины в структуре ОПОП

Дисциплина относится части, формируемой участниками образовательных отношений

№ варианта	Место дисциплины в учебном плане образовательной программы	Стандартный текст для автоматического заполнения в конструкторе РПД
2	Блок 1. Дисциплины (модули) Часть, формируемая участниками образовательных отношений	Дисциплина Б1.В.ДВ.06.02 Компоновка программ, шаблоны относится к части ООП направления подготовки 09.03.03 Прикладная информатика, формируемой участниками образовательных отношений.

## 2. Планируемые результаты обучения по дисциплине, соотнесенные с планируемыми результатами освоения образовательной программы (компетенциями и индикаторами достижения компетенций)

Формируемые компетенции (код, содержание компетенции)	Планируемые результаты обучения по дисциплине (модулю), в соответствии с индикатором достижения компетенции		Наименование оценочного средства
	Индикатор достижения компетенции* (код, содержание индикатора)	Результаты обучения по дисциплине**	
<i>ПК-8. Способен разрабатывать лингвистическое, информационное и программное обеспечение ИС (ИИС) и сопровождающую его документацию</i>	<i>ПК-8.1. Демонстрирует знание современных языков и систем программирования, формализмов описания знаний на концептуальном и инфологическом уровнях, требований к технической документации на все виды обеспечения ИС (ИИС).</i>	<i>Знать основные определения, методы и технологии проектирования сложных объектно-ориентированных систем: принципы функциональной декомпозиции, осознавать проблему изменяющихся требований, знать основные термины, понятия и принципы объектно-ориентированной парадигмы. Знать фундаментальные механизмы объектно-ориентированной технологии (композиция, делегирование); понятие сложности систем, объектной модели. Знать принципы классификации систем и идентификация классов и объектов; ключевые абстракции и механизмы. Знать механизмы повторного использования; систему каталогизации шаблонов проектирования; Основные понятия, принципы и методы программирования ООП. Знать типовые решения (паттерны проектирования): фасад, адаптер, мост, компоновщик, итератор, декоратор, стратегия, одиночка, прототип, абстрактная фабрика, фабричный метод, шаблонный метод, строитель, посредник, цепочка обязанностей, команда, хранитель, посетитель, интерпретатор, заместитель, приспособленец, состояние;</i>	<i>Собеседование Задачи</i>
	<i>ПК-8.2. Применяет современные языки и системы программирования, формализмы</i>	<i>Уметь создавать и воплощать проектные решения сложных объектно-ориентированных систем: редуцировать сложные системы методом функциональной декомпозиции; осуществлять поиск требуемых типовых решений в рамках</i>	

	<i>описания знаний на концептуальном и инфологическом уровнях при разработке лингвистического, информационного и программного обеспечения ИИС и сопровождающей ее документации.</i>	системы каталогизации шаблонов проектирования; реализовывать программы средствами объектно-ориентированного языка; <i>программировать приложения и создавать программные прототипы решения прикладных задач</i> (паттерны проектирования): фасад, адаптер, мост, компоновщик, итератор, декоратор, стратегия, одиночка, прототип, абстрактная фабрика, фабричный метод, шаблонный метод, строитель, посредник, цепочка обязанностей, команда, хранитель, посетитель, интерпретатор, заместитель, приспособленец, состояние.	
	<b>ПК-8.3. Имеет практический опыт разработки лингвистического, информационного и программного обеспечения конкретной ИС (ИИС) и сопровождающей ее документации.</b>	<i>Владеть методами и технологиями проектирования сложных программных систем:</i> техникой функциональной декомпозиции; техникой выбора, реализации и адаптации шаблонов проектирования. опыт реализации программных систем: фасад, адаптер, мост, компоновщик, итератор, декоратор, стратегия, одиночка, прототип, абстрактная фабрика, фабричный метод, шаблонный метод, строитель, посредник, цепочка обязанностей, команда, хранитель, посетитель, интерпретатор, заместитель, приспособленец, состояние.	

### 3. Структура и содержание дисциплины

#### 3.1. Трудоемкость дисциплины

	Очно-заочная форма обучения
<b>Общая трудоемкость</b>	<b>3 ЗЕТ</b>
<b>Часов по учебному плану</b>	<b>108</b>
<b>в том числе</b>	
<b>аудиторные занятия (контактная работа):</b>	<b>34</b>
- занятия лекционного типа	<b>0</b>
- занятия семинарского типа	<b>32</b>
- занятия лабораторного типа	<b>0</b>
- текущий контроль (КСР)	<b>2</b>
<b>самостоятельная работа</b>	<b>38</b>
<b>Промежуточная аттестация – экзамен</b>	<b>36</b>

#### 3.2. Содержание дисциплины

Наименование и краткое содержание разделов и тем дисциплины	Всего (часы)	В том числе	
		Контактная работа (работа во взаимодействии с преподавателем), часы. Из них	аудиторная работа обучаю

		Занятия лекционного типа	Занятия семинарского типа	Занятия лабораторного типа	Всего	
Тема 1. Объектно-ориентированная парадигма	5	0	2	0	2	3
Тема 2. Введение в объектно-ориентированный анализ и базис языка визуального моделирования	5	0	2	0	2	3
Тема 3. Шаблонный метод проектирования	5	0	2	0	2	3
Тема 4. Шаблоны Фасад и Адаптер	7	0	4	0	4	3
Тема 5. Шаблон Мост	5	0	2	0	2	3
6. Шаблоны Компоновщик и Итератор	5	0	2	0	2	3
Тема 7. Шаблоны Декоратор и Стратегия	5	0	2	0	2	3
Тема 8. Инстанцирование объектно-организованных систем	8	0	4	0	4	4
Тема 9. Информационный обмен между объектами	5	0	2	0	2	3
Тема 10. Управление системой объектов	7	0	4	0	4	3
Тема 11. Функциональное расширение системы с минимальными изменениями	8	0	4	0	4	4
Тема 12. Проектирование с элементами оптимизации	5	0	2	0	2	3
Текущий контроль (КСР)	2					
Промежуточная аттестация – экзамен	36					
Итого	108	0	32	0	34	74

Текущий контроль успеваемости реализуется в формах опросов на занятиях семинарского типа

Промежуточная аттестация проходит в традиционных формах (зачет, экзамен)

#### 4. Учебно-методическое обеспечение самостоятельной работы обучающихся

Самостоятельная работа проходит в форме Выполнение домашних практических заданий.

Контрольные вопросы и задания для проведения текущего контроля и промежуточной аттестации по итогам освоения дисциплины приведены в п. 5.2.

**5. Фонд оценочных средств для промежуточной аттестации по дисциплине (модулю), включающий:**

#### 5.1. Описание шкал оценивания результатов обучения по дисциплине

Уровень сформированности компетенций	Шкала оценивания сформированности компетенций						
	плохо	неудовлетворительно	удовлетворительно	хорошо	очень хорошо	отлично	превосходно

(индикатора достижения компетенций)	Не зачтено		Зачтено				
	<u>Знания</u>	Отсутствие знаний теоретического материала.  Невозможность оценить полноту знаний вследствие отказа обучающегося от ответа	Уровень знаний ниже минимальных требований. Имели место грубые ошибки.	Минимально допустимый уровень знаний. Допущено много негрубых ошибок.	Уровень знаний в объеме, соответствующем программе подготовки. Допущено несколько негрубых ошибок	Уровень знаний в объеме, соответствующем программе подготовки. Допущено несколько несущественных ошибок	Уровень знаний в объеме, соответствующем программе подготовки, без ошибок.
<u>Умения</u>	Отсутствие минимальных умений. Невозможность оценить наличие умений вследствие отказа обучающегося от ответа	При решении стандартных задач не продемонстрированы основные умения.  Имели место грубые ошибки.	Продemonстрированы основные умения. Решены типовые задачи с негрубыми ошибками. Выполнены все задания, но не в полном объеме.	Продemonстрированы все основные умения. Решены все основные задачи с негрубыми ошибками. Выполнены все задания, в полном объеме, но некоторые с недочетами.	Продemonстрированы все основные умения. Решены все основные задачи. Выполнены все задания, в полном объеме, но некоторые с недочетами.	Продemonстрированы все основные умения, решены все основные задачи с отдельными несущественными недочетами, выполнены все задания в полном объеме.	Продemonстрированы все основные умения, решены все основные задачи. Выполнены все задания, в полном объеме без недочетов
<u>Навыки</u>	Отсутствие владения материалом. Невозможность оценить наличие навыков вследствие отказа обучающегося от ответа	При решении стандартных задач не продемонстрированы базовые навыки.  Имели место грубые ошибки.	Имеется минимальный набор навыков для решения стандартных задач с некоторыми недочетами.	Продemonстрированы базовые навыки при решении стандартных задач с некоторыми недочетами	Продemonстрированы базовые навыки при решении стандартных задач без ошибок и недочетов.	Продemonстрированы навыки при решении нестандартных задач без ошибок и недочетов.	Продemonстрирован творческий подход к решению нестандартных задач.

### Шкала оценки при промежуточной аттестации

Оценка		Уровень подготовки
зачтено	Превосходно	Все компетенции (части компетенций), на формирование которых направлена дисциплина, сформированы на уровне не ниже «превосходно»
	Отлично	Все компетенции (части компетенций), на формирование которых направлена дисциплина, сформированы на уровне не ниже «отлично», при этом хотя бы одна компетенция сформирована на уровне «отлично»

	Очень хорошо	Все компетенции (части компетенций), на формирование которых направлена дисциплина, сформированы на уровне не ниже «очень хорошо», при этом хотя бы одна компетенция сформирована на уровне «очень хорошо»
	Хорошо	Все компетенции (части компетенций), на формирование которых направлена дисциплина, сформированы на уровне не ниже «хорошо», при этом хотя бы одна компетенция сформирована на уровне «хорошо»
	Удовлетворительно	Все компетенции (части компетенций), на формирование которых направлена дисциплина, сформированы на уровне не ниже «удовлетворительно», при этом хотя бы одна компетенция сформирована на уровне «удовлетворительно»
не зачтено	Неудовлетворительно	Хотя бы одна компетенция сформирована на уровне «неудовлетворительно», ни одна из компетенций не сформирована на уровне «плохо»
	Плохо	Хотя бы одна компетенция сформирована на уровне «плохо»

## 5.2. Типовые контрольные задания или иные материалы, необходимые для оценки результатов обучения

### 5.2.1 Контрольные вопросы

<i>вопросы</i>	<i>Код формируемых компетенций</i>
<ol style="list-style-type: none"> <li>1. Функциональная декомпозиция. Проблема обработки изменяющихся требований.</li> <li>2. Объектно-ориентированная парадигма. Модели наследования и композиции.</li> <li>3. Объектно-ориентированная парадигма. Инкапсуляция в механизме полиморфизма.</li> <li>4. Объектно-ориентированная парадигма. Модели делегирования.</li> <li>5. Абстрактный класс и интерфейс. Реализации класса и интерфейса. Проектирование на уровне интерфейсов.</li> <li>6. Сложность систем. Объектная модель. Классификация. Идентификация классов и объектов. Ключевые абстракции и механизмы.</li> <li>7. Основные этапы разработки объектно-ориентированной системы.</li> <li>8. UML. Определение. Назначение. Концептуальная модель.</li> <li>9. Концептуальная модель языка UML. Сущности.</li> <li>10. Концептуальная модель языка UML. Отношения.</li> <li>11. Концептуальная модель языка UML. Диаграмма прецедентов и диаграммы взаимодействий.</li> <li>12. Концептуальная модель языка UML. Статические и динамические модели программных систем на примере диаграмм классов и диаграмм объектов.</li> <li>13. Шаблоны проектирования. Основные термины и понятия. Механизмы повторного использования. Система каталогизации шаблонов проектирования.</li> <li>14. Методы проектирования, основанные на стратегиях и шаблонах проектирования.</li> <li>15. Модели инкапсуляции. Стратегия «Инкапсуляции изменчивости» в анализе общности и изменчивости.</li> <li>16. Стратегии «Композиция предпочтительней наследования» и «Одно правило, одно место».</li> <li>17. Специфика работа со сложной системой с множеством интерфейсов. Шаблон Фасад (Facade).</li> <li>18. Проблема совместимости интерфейсов. Шаблон Адаптер (Adapter).</li> </ol>	ПК-8

19. Понятия абстракции и реализации. Механизм отделения абстракции от реализации. Шаблон Мост (Bridge).
20. Механизм компоновки объектов в плоские коллекции и иерархические структуры. Манипулирование составными объектами. Шаблон Компоновщик (Composite). Виды Компоновщика.
21. Составные объекты. Организация доступа к элементам составного объекта на базе шаблона Итератор (Iterator).
22. Робастность итераторов плоских коллекций и иерархических структур.
23. Динамическое расширение функциональности объектов. Шаблон Декоратор (Decorator) – как гибкая альтернатива порождению подклассов.
24. Инкапсуляция алгоритма в объект. Механизм «прозрачной» замены алгоритма. Шаблон Стратегия (Strategy).
25. Принципы инстанцирования объектов и систем. Идеология объекта-одиночки (Singleton) в системе объектов. Способы доступа к объекту-одиночке.
26. Конфигурирование и инстанцирование систем объектов. Абстрактная Фабрика (Abstract Factory) и Инструментарий (Kit).
27. Применение решений Фабричного Метода (Factory Method) и Шаблонного Метода (Template Method) в конструировании каркасов приложений с использованием.
28. Клонирование объектов и систем объектов. Поверхностное и глубокое клонирование на базе Прототипа (Prototype).
29. Организация процесса конструирования различных представлений сложного объекта на базе решения Строитель (Builder).
30. Информационный обмен между объектами. Основополагающие принципы. Классификация моделей. Простейшие модели и модель на базе Посредника (Mediator).
31. Модель доставки сообщения на базе решения Цепочка Обязанностей (Chain of Responsibility). Решение без менеджера и с менеджером. Проксирование сообщений.
32. Широковещательные трансляции на базе шаблона Наблюдатель (Observer). Особенности реализации систем типа Субъект-Наблюдатель без менеджера и с менеджером.
33. Объекты с функциональностью Субъекта и Наблюдателя. Проблема учета циклических связей (зависимостей). Решение без менеджера и с менеджером.
34. Идеология представление команды (операции) в виде объекта. Манипулирование командами как объектами. Протоколирование команд.
35. Идеология представление команды (операции) в виде объекта. Манипулирование командами как объектами. Организация макросов (составные команды) на базе шаблона Компоновщик (Composite).
36. Менеджер команд и универсальные механизмы отката (отмены операций) на базе решений Команда (Command) и Хранитель (Memento).
37. Нарращивание функциональности отдельных объектов (классов) без изменения существующего кода на базе решений Декоратор (Decorator) и Стратегия (Strategy).
38. Двойная диспетчеризация. Динамическое определение новых функций для систем объектов без изменения существующего кода на базе решения Посетитель (Visitor).
39. Представление грамматики языка и интерпретация предложений на базе шаблона Интерпретатор (Interpreter).
40. Планирование вычислительных ресурсов. Идеологии кэширования и отложенной реакции на событие. Объектно-ориентированная организация событийных систем на основе решения Заместитель (Proxy).

<p>41. Идеология разделения объекта и его состояния. Объектно-ориентированная организация систем с большим числом объектов на основе решения Приспособленец (Flyweight).</p> <p>42. Идеология совмещения в одном объекта разных состояний на основе решения Состояние (State).</p>	
--	--

### 5.2.2. Типовые задания/задачи для оценки сформированности компетенции ПК-1

Типовая задача 1. По описанию архитектуры построить программный код средствами объектно-ориентированного языка.

Типовая задача 2. Для заданной архитектуры построить ее развитие так, чтобы обеспечить возможность выполнения заданного сценария на уровне клиентского кода средствами объектно-ориентированного языка.

Типовая задача 3. Для заданной модели функционирования некоторой системы построить ее программную реализацию на основе методологии шаблонов проектирования.

Типовая задача 4. По коду построить UML-диаграммы классов для основных элементов архитектуры.

Типовая задача 5. По коду клиентской части построить UML-диаграммы последовательностей/объектов.

Типовая задача 6. Для заданного шаблона проектирования построить UML-диаграмму классов.

Типовая задача 7. Для заданного шаблона проектирования построить UML-диаграмму последовательностей, демонстрирующей принципы его работы.

Типовая задача 8. В рамках заданной архитектуры для заданного сценария реализовать клиентский код средствами объектно-ориентированного языка.

Типовая задача 9. Для заданного шаблона проектирования построить программный код, демонстрирующий принципы его работы.

## 6. Учебно-методическое и информационное обеспечение дисциплины

а) основная литература:

1. Хорев П. Б. - Технологии объектно-ориентированного программирования: учеб. пособие для студентов, обучающихся по направлению "Информатика и вычислительная техника". - М.: Академия, 2004. - 448 с. – 40 экз.

б) дополнительная литература:

2. Лафоре Р. - Объектно-ориентированное программирование в C++. - СПб.: Питер, 2007. - 928 с. – 6 экз

в) программное обеспечение и Интернет-ресурсы

1. Справочник по шаблонам проектирования. <http://www.oodesign.com/>
2. Архитектурное проектирование программного обеспечения. Бесплатный курс для самообразования на официальном сайте Интуит.ру. <http://www.intuit.ru/studies/courses/3509/751/info>
3. Объектно-ориентированный анализ и программирование. Бесплатный курс для самообразования на официальном сайте Интуит.ру. <http://www.intuit.ru/studies/courses/491/347/info>
4. Проектирование информационных систем. Бесплатный курс для самообразования на официальном сайте Интуит.ру. <http://www.intuit.ru/studies/courses/491/347/info>

## **7. Материально-техническое обеспечение дисциплины**

Помещения представляют собой учебные аудитории для проведения учебных занятий, предусмотренных программой, оснащенные оборудованием и техническими средствами обучения: Использование терминал-класса для выполнения практических работ

Помещения для самостоятельной работы обучающихся оснащены компьютерной техникой с возможностью подключения к сети "Интернет" и обеспечены доступом в электронную информационно-образовательную среду

Программа составлена в соответствии с требованиями ОС ННГУ с учетом рекомендаций и ОПОП ВО по направлению (профилю) **09.03.03 Прикладная информатика**

Автор \_\_\_\_\_ профессор Старостин Н.В.

Рецензент \_\_\_\_\_ профессор Федосенко Ю.С.

Заведующий кафедрой \_\_\_\_\_ профессор М.Х. Прилуцкий

Программа одобрена на заседании методической комиссии института информационных технологий, математики и механики

30.11.2022 года, протокол № 3