

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ
ФЕДЕРАЦИИ
федеральное государственное автономное образовательное учреждение высшего
образования
"САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
АЭРОКОСМИЧЕСКОГО ПРИБОРОСТРОЕНИЯ"

Кафедра № 43

УТВЕРЖДАЮ
Руководитель образовательной программы

(должность, уч. степень, звание)

Д.В. Курьяник

(инициалы, фамилия)

(подпись)

«16» февраля 2026 г

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

«Технология программирования»
(Наименование дисциплины)

| | |
|---|---|
| Код направления подготовки/ специальности | 09.03.01 |
| Наименование направления подготовки/ специальности | Информатика и вычислительная техника |
| Наименование направленности/ специализации | Компьютерные технологии, системы и сети |
| Форма обучения | очная |
| Год приема | 2026 |

Лист согласования рабочей программы дисциплины

Программу составил (а)

Ст. преподаватель
(должность, уч. степень, звание)



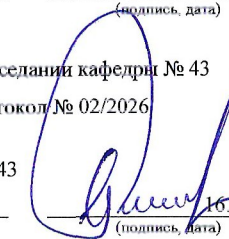
16.02.26
(подпись, дата)

Н.А. Соловьева
(инициалы, фамилия)

Программа одобрена на заседании кафедры № 43
«16» февраля 2026 г, протокол № 02/2026

Заведующий кафедрой № 43

д.т.н., проф.
(уч. степень, звание)



16.02.26
(подпись, дата)

М.Ю. Охтилев
(инициалы, фамилия)

Заместитель директора института №4 по методической работе

доц. к.т.н.
(должность, уч. степень, звание)



16.02.26
(подпись, дата)

А.А. Фоменкова
(инициалы, фамилия)

Аннотация

Дисциплина «Технология программирования» входит в образовательную программу высшего образования – программу бакалавриата по направлению подготовки/специальности 09.03.01 «Информатика и вычислительная техника» направленности/специализации «Компьютерные технологии, системы и сети». Дисциплина реализуется кафедрой «№43».

Дисциплина нацелена на формирование у выпускника следующих компетенций:

ОПК-2 «Способен понимать принципы работы современных информационных технологий и программных средств, в том числе отечественного производства, и использовать их при решении задач профессиональной деятельности»

ОПК-4 «Способен участвовать в разработке стандартов, норм и правил, а также технической документации, связанной с профессиональной деятельностью»

Содержание дисциплины охватывает круг вопросов, связанных с анализом и проектированием больших программных комплексов или информационных систем (ИС), с использованием современных методологий и средств проектирования. Также рассматриваются темы, связанные объектно-ориентированным программированием и проектирование.

Преподавание дисциплины предусматривает следующие формы организации учебного процесса: лекции, лабораторные работы, самостоятельная работа обучающегося.

Программой дисциплины предусмотрены следующие виды контроля: текущий контроль успеваемости, промежуточная аттестация в форме дифференцированного зачета (4 семестр), экзамена (5 семестр).

Общая трудоемкость освоения дисциплины составляет 5 зачетных единиц, 180 часов.

Язык обучения по дисциплине «русский»

1. Перечень планируемых результатов обучения по дисциплине

1.1. Цели преподавания дисциплины

Целью преподавания дисциплины «Технология программирования» является получение обучающимися необходимых знаний, умений и навыков в области проектирования и программирования информационных систем в различных прикладных областях с использованием современных методологий и технологических средств проектирования.

1.2. Дисциплина входит в состав обязательной части образовательной программы высшего образования (далее – ОП ВО).

1.3. Перечень планируемых результатов обучения по дисциплине, соотнесенных с планируемыми результатами освоения ОП ВО.

В результате изучения дисциплины обучающийся должен обладать следующими компетенциями или их частями. Компетенции и индикаторы их достижения приведены в таблице 1.

Таблица 1 – Перечень компетенций и индикаторов их достижения

| Категория (группа) компетенции | Код и наименование компетенции | Код и наименование индикатора достижения компетенции |
|----------------------------------|---|--|
| Общепрофессиональные компетенции | ОПК-2 Способен понимать принципы работы современных информационных технологий и программных средств, в том числе отечественного производства, и использовать их при решении задач профессиональной деятельности | ОПК-2.3.1 знать современные информационные технологии и программные средства, в том числе отечественного производства при решении задач профессиональной деятельности ОПК-2.У.1 уметь выбирать современные информационные технологии и программные средства, в том числе отечественного производства при решении задач профессиональной деятельности ОПК-2.В.1 владеть навыками применения современных информационных технологий и программных средств, в том числе отечественного производства, при решении задач профессиональной деятельности |
| Общепрофессиональные компетенции | ОПК-4 Способен участвовать в разработке стандартов, норм и правил, а также технической документации, связанной с профессиональной деятельностью | ОПК-4.3.1 знать основные стандарты оформления технической документации на различных стадиях жизненного цикла информационной системы ОПК-4.У.1 уметь применять стандарты оформления технической документации на различных стадиях жизненного цикла информационной системы ОПК-4.В.1 владеть навыком составления технической документации на различных этапах |

| | | |
|--|--|---|
| | | жизненного цикла информационной системы |
|--|--|---|

2. Место дисциплины в структуре ОП

Дисциплина может базироваться на знаниях, ранее приобретенных обучающимися при изучении следующих дисциплин:

- «Основы программирования»,
- «Алгоритмы и структуры данных».

Знания, полученные при изучении материала данной дисциплины, имеют как самостоятельное значение, так и используются при изучении других дисциплин:

- «Базы данных».

3. Объем и трудоемкость дисциплины

Данные об общем объеме дисциплины, трудоемкости отдельных видов учебной работы по дисциплине (и распределение этой трудоемкости по семестрам) представлены в таблице 2.

Таблица 2 – Объем и трудоемкость дисциплины

| Вид учебной работы | Всего | Трудоемкость по семестрам | |
|---|----------------------|---------------------------|--------|
| | | №4 | №5 |
| 1 | 2 | 3 | 4 |
| Общая трудоемкость дисциплины, ЗЕ/ (час) | 5/ 180 | 2/ 72 | 3/ 108 |
| Из них часов практической подготовки | | | |
| Аудиторные занятия, всего час. | 85 | 51 | 34 |
| в том числе: | | | |
| лекции (Л), (час) | 51 | 34 | 17 |
| практические/семинарские занятия (ПЗ), (час) | | | |
| лабораторные работы (ЛР), (час) | 34 | 17 | 17 |
| курсовой проект (работа) (КП, КР), (час) | | | |
| экзамен, (час) | 36 | | 36 |
| Самостоятельная работа, всего (час) | 59 | 21 | 38 |
| Вид промежуточной аттестации: зачет, дифф. зачет, экзамен (Зачет, Дифф. зач, Экз.) | Дифф. зач., Экз., | Дифф. зач., | Экз., |

4. Содержание дисциплины

4.1. Распределение трудоемкости дисциплины по разделам и видам занятий.

Разделы, темы дисциплины и их трудоемкость приведены в таблице 3.

Таблица 3 – Разделы, темы дисциплины, их трудоемкость

| Разделы, темы дисциплины | Лекции (час) | ПЗ (СЗ) | ЛР (час) | КП/КР (час) | СР (час) |
|---|--------------|---------|----------|-------------|----------|
| Семестр 4 | | | | | |
| Раздел 1. Информационные системы, подходы и методологии проектирования. | | | | | |
| Тема 1.1. Этапы развития технологии программирования и средств создания программного обеспечения. | 10 | | 3 | | 7 |

| | | | | | |
|--|----|--|----|--|----|
| <p>Тема 1.2. Жизненный цикл программного обеспечения, его основные этапы и модели.</p> <p>Тема 1.3. Тестирование программного обеспечения</p> <p>Тема 1.4. Стандарты в области программного обеспечения</p> <p>Тема 1.5. Документация программного обеспечения</p> | | | | | |
| <p>Раздел 2. Методология структурного анализа и проектирования программного обеспечения.</p> <p>Тема 2.1. Сущность и базовые принципы структурного подхода. Основные средства и диаграммы.</p> <p>Тема 2.2. Диаграммы потоков данных. Иерархия и декомпозиция диаграмм. Методы спецификации процессов. Построение словаря данных.</p> <p>Тема 2.3. Инфологическое моделирование ИС, ER-диаграммы. Нормализация ER-диаграмм, построение структуры базы данных ИС</p> | 14 | | 9 | | 7 |
| <p>Раздел 3. Объектно-ориентированное программирование (ООП) и библиотека STL языка C++.</p> <p>Тема 3.1. Основные принципы и понятия ООП</p> <p>Тема 3.2. Библиотека STL языка C++: контейнеры, итераторы, алгоритмы.</p> | 10 | | 5 | | 7 |
| Итого в семестре: | 34 | | 17 | | 21 |
| Семестр 5 | | | | | |

| | | | | | |
|--|----|---|----|---|----|
| <p>Раздел 4. Методологии объектно-ориентированного анализа и проектирования программного обеспечения.</p> <p>Тема 4.1. Основные принципы объектно-ориентированного проектирования и модели ИС. Универсальный язык моделирования UML, основные понятия, обозначения и диаграммы.</p> <p>Тема 4.2. Диаграмма вариантов использования. Назначение, связь с другими диаграммами, обозначения, рассмотрение примера.</p> <p>Тема 4.3. Разработка программы по принципам объектно-ориентированного программирования.</p> <p>Тема 4.4. Диаграмма классов. Назначение, связь с другими диаграммами, обозначения, рассмотрение примера.</p> <p>Тема 4.5. Диаграммы последовательностей, кооперации, состояния, пакетов и размещения.</p> <p>Тема 4.6. Автоматическая генерация кодов программ, обратное проектирование (реинжиниринг).</p> <p>Тема 4.7. Технологические средства разработки программного обеспечения. CASE-технологии и CASE-пакеты.</p> | 14 | | 13 | | 15 |
| <p>Раздел 5. Шаблоны проектирования.</p> <p>Тема 5.1. Шаблоны проектирования – основные понятия, назначение.</p> <p>Тема 5.2. Знакомство с примерами порождающих, структурных и поведенческих шаблонов.</p> | 3 | | 4 | | 23 |
| Итого в семестре: | 17 | | 17 | | 38 |
| Итого | 51 | 0 | 34 | 0 | 59 |
| | | | | | |

Практическая подготовка заключается в непосредственном выполнении обучающимися определенных трудовых функций, связанных с будущей профессиональной деятельностью.

4.2. Содержание разделов и тем лекционных занятий.

Содержание разделов и тем лекционных занятий приведено в таблице 4.

Таблица 4 – Содержание разделов и тем лекционного цикла

| Номер раздела | Название и содержание разделов и тем лекционных занятий |
|---------------|---|
| 1 | <p align="center">Информационные системы, подходы и методологии проектирования.</p> <p>Лекция 1.1. Этапы развития вычислительной техники, задач и средств Технологии программирования. Понятие информационной системы. Цели и задачи проектирования. Организация проекта программного обеспечения.</p> <p>Лекция 1.2. Жизненный цикл программного обеспечения. Линейная, каскадная и спиралевидная модели жизненного цикла программного обеспечения. Основные этапы жизненного цикла и инструментальные средства поддержки проекта.</p> <p>Лекция 1.3 Тестирование программного обеспечения: классификация видов, области применения, инструментальные средства.</p> <p>Лекция 1.4 Стандарты в области информационных технологий</p> <p>Лекция 1.5. Документация программного обеспечения</p> |
| 2 | <p align="center">Методология структурного анализа и проектирования программного обеспечения.</p> <p>Лекция 2.1. Сущность и базовые принципы структурного подхода. Основные этапы и средства структурного анализа и проектирования информационных систем. Основные диаграммы.</p> <p>Лекция 2.2. Диаграммы потоков данных. Декомпозиция ДПД, построение словаря проекта. Построение словаря данных. Атрибуты описания потоков данных. БНФ-определение. Примеры описания потоков данных. Методы спецификации процессов. Структурированный естественный язык (псевдокод), визуальные языки проектирования, FLOW-формы и диаграммы Насси-Шнейдермана (структурограммы).</p> <p>Лекция 2.3. Инфологическое моделирование. ИС. Проектирование структуры базы данных ИС. ER-диаграммы. Базовые элементы и их свойства. Нормализация ER-диаграмм. Пример построения ER-диаграммы. Примеры нормализации, приведение диаграмм к 1НФ, 2НФ, 3НФ, устранение связей типа М:М.</p> |
| 3 | <p align="center">Объектно-ориентированное программирование (ООП) и библиотека STL языка C++.</p> <p>Лекция 3.1. Основные принципы и понятия ООП: класс, объект, поля, методы, уровни доступа, наследование.</p> <p>Лекция 3.2. Библиотека STL языка C++: контейнеры, итераторы, алгоритмы.</p> |

| | |
|---|---|
| 4 | <p style="text-align: center;">Методологии объектно-ориентированного анализа и проектирования программного обеспечения.</p> <p>Лекция 4.1. Основные принципы объектно-ориентированных методов проектирования. Инкапсуляция, наследование, полиморфизм. Статическая и динамическая модели информационной системы. Универсальный язык моделирования UML. Структура языка, основные понятия и обозначения.</p> <p>Лекция 4.2. Основные диаграммы языка UML. Взаимосвязь диаграмм. Диаграмма вариантов использования. Основные понятия и обозначения. Пример построения диаграммы вариантов использования.</p> <p>Лекция 4.3. Принципы объектно-ориентированного программирования. Разработка программы с классами.</p> <p>Лекция 4.4. Диаграмма классов. Основные понятия и обозначения. Пример построения диаграммы классов.</p> <p>Лекция 4.5. Диаграммы последовательностей и кооперации. Рассмотрение примеров.</p> <p>Лекция 4.6. Диаграммы состояния и видов деятельности. Рассмотрение примеров</p> <p>Лекция 4.7. Диаграммы пакетов и размещения. Рассмотрение примеров</p> <p>Лекция 4.8. Автоматическая генерация кодов программ, обратное проектирование (реинжиниринг).</p> <p>Лекция 4.9. Технологические средства разработки программного обеспечения ПО. CASE-пакеты. Структура типового CASE-средства.</p> |
| 5 | <p style="text-align: center;">Шаблоны проектирования.</p> <p>Лекция 5.1. Шаблоны проектирования – основные понятия, назначение.</p> <p>Лекция 5.2. Знакомство с примерами порождающих, структурных и поведенческих шаблонов.</p> |

4.3. Практические (семинарские) занятия

Темы практических занятий и их трудоемкость приведены в таблице 5.

Таблица 5 – Практические занятия и их трудоемкость

| № п/п | Темы практических занятий | Формы практических занятий | Трудоемкость, (час) | Из них практической подготовки, (час) | № раздела дисциплины |
|---------------------------------|---------------------------|----------------------------|---------------------|---------------------------------------|----------------------|
| Учебным планом не предусмотрено | | | | | |
| | | | | | |
| Всего | | | | | |

4.4. Лабораторные занятия

Темы лабораторных занятий и их трудоемкость приведены в таблице 6.

Таблица 6 – Лабораторные занятия и их трудоемкость

| № п/п | Наименование лабораторных работ | Трудоемкость, (час) | Из них практической подготовки, (час) | № раздела дисциплины |
|-----------|--|---------------------|---------------------------------------|----------------------|
| Семестр 4 | | | | |
| 1 | Разработка диаграммы потоков данных для предметной области. | 4 | 2 | 2 |
| 2 | Разработка спецификаций для информационных процессов | 2 | 2 | 2 |
| 3 | Разработка программной документации | 3 | 2 | 1 |
| 4 | Разработка диаграммы «Сущность-связь» для предметной области. | 3 | 2 | 2 |
| 5 | Объектно-ориентированное программирование | 2 | 2 | 3 |
| 6 | Знакомство с библиотекой STL | 3 | 2 | 3 |
| Семестр 5 | | | | |
| 1 | Язык UML. Разработка диаграммы вариантов использования. | 2 | 2 | 4 |
| 2 | Язык UML. Разработка диаграммы последовательностей. | 2 | 2 | 4 |
| 3 | Язык UML. Разработка диаграммы классов | 4 | 2 | 4 |
| 4 | Язык UML. Разработка диаграмм пакетов и развертывания | 2 | 2 | 4 |
| 5 | Автоматическая генерация кода программы по диаграмме классов и обратное проектирование | 3 | 2 | 4 |
| 6 | Шаблоны проектирования | 4 | 2 | 5 |
| | | | | |
| Всего | | 34 | | |

4.5. Выполнение курсового проекта/ курсовой работы
Учебным планом не предусмотрено

4.6. Самостоятельная работа обучающихся
Виды самостоятельной работы и ее трудоемкость приведены в таблице 7.

Таблица 7 – Виды самостоятельной работы и ее трудоемкость

| Вид самостоятельной работы | Всего, час | Семестр 4, час | Семестр 5, час |
|---|------------|----------------|----------------|
| 1 | 2 | 3 | 4 |
| Изучение теоретического материала дисциплины (ТО) | 30 | 10 | 20 |
| Подготовка к текущему контролю успеваемости (ТКУ) | 15 | 7 | 8 |

| | | | |
|--|----|----|----|
| Подготовка к промежуточной аттестации (ПА) | 14 | 4 | 10 |
| Всего: | 59 | 21 | 38 |

5. Перечень учебно-методического обеспечения для самостоятельной работы обучающихся по дисциплине (модулю)
Учебно-методические материалы для самостоятельной работы обучающихся указаны в п.п. разделов 6-11.

6. Перечень печатных и электронных учебных изданий
Перечень печатных и электронных учебных изданий приведен в таблице 8.
Таблица 8– Перечень печатных и электронных учебных изданий

| Шифр/ URL адрес | Библиографическая ссылка | Количество экземпляров в библиотеке (кроме электронных экземпляров) |
|---|--|---|
| URL: https://e.lanbook.com/book/176670 (дата обращения: 11.02.2026) | Волк, В. К. Базы данных. Проектирование, программирование, управление и администрирование : учебник для вузов / В. К. Волк. — 2-е изд., стер. — Санкт-Петербург : Лань, 2021. — 244 с. — ISBN 978-5-8114-8412-6. — Текст : электронный // Лань : электронно-библиотечная система Режим доступа: для авториз. пользователей. | |
| URL: https://e.lanbook.com/book/175503 (дата обращения: 11.02.2026) | Маран, М. М. Программная инженерия : учебное пособие для вузов / М. М. Маран. — 2-е изд., стер. — Санкт-Петербург : Лань, 2021. — 196 с. — ISBN 978-5-8114-8367-9. — Текст : электронный // Лань : электронно-библиотечная система. Режим доступа: для авториз. пользователей. | |
| URL: https://e.lanbook.com/book/111721 (дата обращения: 11.02.2026) | Забродин, А. В. Основы проектирования информационных систем с помощью языка UML : учебное пособие / А. В. Забродин, В. П. Бубнов. — Санкт-Петербург : ПГУПС, 2018. — 46 с. — ISBN 978-5-7641-1133-9. — Текст : электронный // Лань : электронно-библиотечная система. Режим доступа: для авториз. пользователей. | |

| | | |
|---|---|--|
| URL: https://e.lanbook.com/book/168336 (дата обращения: 11.02.2026) | Гринченко, Н. Н. Разработка моделей информационных систем на языке UML : учебное пособие / Н. Н. Гринченко, Ю. В. Конкин. — Рязань : РГРТУ, 2015. — 48 с. — Текст : электронный // Лань : электронно-библиотечная система. Режим доступа: для авториз. пользователей. | |
| URL: https://e.lanbook.com/book/493547 (дата обращения: 11.02.2026) | Объектно-ориентированное программирование : учебное пособие / В. В. Лозовский, Е. Н. Штрекер, Е. С. Данилович [и др.]. — Москва : РТУ МИРЭА, 2025. — 484 с. — ISBN 978-5-7339-2498-4. — Текст : электронный // Лань : электронно-библиотечная система. — Режим доступа: для авториз. пользователей. | |
| URL: https://e.lanbook.com/book/404348 (дата обращения: 11.02.2026) | Зайцев, М. Г. Контейнеры STL языка C++ : учебное пособие / М. Г. Зайцев. — Новосибирск : НГТУ, 2023. — 134 с. — ISBN 978-5-7782-4966-0. — Текст : электронный // Лань : электронно-библиотечная система. — Режим доступа: для авториз. пользователей. | |
| URL: https://e.lanbook.com/book/382706 (дата обращения: 11.02.2026) | Гусев, К. В. Технология разработки программных приложений : учебное пособие / К. В. Гусев, М. Б. Туманова, Е. А. Чернов. — Москва : РТУ МИРЭА, 2023. — 146 с. — ISBN 978-5-7339-1938-6. — Текст : электронный // Лань : электронно-библиотечная система. — Режим доступа: для авториз. пользователей. | |

7. Перечень электронных образовательных ресурсов информационно-телекоммуникационной сети «Интернет»

Перечень электронных образовательных ресурсов информационно-телекоммуникационной сети «Интернет», необходимых для освоения дисциплины приведен в таблице 9.

Таблица 9 – Перечень электронных образовательных ресурсов информационно-телекоммуникационной сети «Интернет»

| URL адрес | Наименование |
|---|--|
| https://intuit.ru/studies/courses/3632/874/lecture/14297 | Лекция «Жизненный цикл программных систем» |

8. Перечень информационных технологий

8.1. Перечень программного обеспечения, используемого при осуществлении образовательного процесса по дисциплине.

Перечень используемого программного обеспечения представлен в таблице 10.

Таблица 10– Перечень программного обеспечения

| № п/п | Наименование |
|-------|---|
| 1 | Редактор для рисования диаграмм уEd. Распространяется свободно |
| 2 | Онлайн компилятор для языка C++ (https://coddy.tech/playground/ru/cpp) |

8.2. Перечень информационно-справочных систем, используемых при осуществлении образовательного процесса по дисциплине

Перечень используемых информационно-справочных систем представлен в таблице 11.

Таблица 11– Перечень информационно-справочных систем

| № п/п | Наименование |
|-------|------------------|
| | Не предусмотрено |

9. Материально-техническая база

Состав материально-технической базы, необходимой для осуществления образовательного процесса по дисциплине, представлен в таблице 12.

Таблица 12 – Состав материально-технической базы

| № п/п | Наименование составной части материально-технической базы | Номер аудитории (при необходимости) |
|-------|---|-------------------------------------|
| 1 | Лекционная аудитория | |
| 2 | Компьютерный класс | |

10. Оценочные средства для проведения промежуточной аттестации

10.1. Состав оценочных средств для проведения промежуточной аттестации обучающихся по дисциплине приведен в таблице 13.

Таблица 13 – Состав оценочных средств для проведения промежуточной аттестации

| Вид промежуточной аттестации | Перечень оценочных средств |
|------------------------------|---------------------------------------|
| Экзамен | Список вопросов к экзамену; Тесты. |
| Дифференцированный зачёт | Список вопросов; Тесты. |

10.2. В качестве критериев оценки уровня сформированности (освоения) компетенций обучающимися применяется 5-балльная шкала оценки сформированности компетенций, которая приведена в таблице 14. В течение семестра может использоваться 100-балльная шкала модульно-рейтинговой системы Университета, правила использования которой, установлены соответствующим локальным нормативным актом ГУАП.

Таблица 14 –Критерии оценки уровня сформированности компетенций

| Оценка компетенции | Характеристика сформированных компетенций |
|---------------------------------------|--|
| 5-балльная шкала | |
| «отлично» «зачтено» | Обучающийся: – глубоко и всесторонне усвоил программный материал; – уверенно, логично, последовательно и грамотно его излагает; – опираясь на знания основной и дополнительной литературы, тесно связывает усвоенные научные положения с практической деятельностью направления; – умело обосновывает и аргументирует выдвигаемые им идеи; – делает выводы и обобщения; – свободно владеет системой специализированных понятий. – правильно выполнил от 90% до 100% тестовых заданий**. |
| «хорошо» «зачтено» | Обучающийся: – твердо усвоил программный материал, грамотно и по существу излагает его, опираясь на знания основной литературы; – не допускает существенных неточностей; – увязывает усвоенные знания с практической деятельностью направления; – аргументирует научные положения; – делает выводы и обобщения; – владеет системой специализированных понятий. – правильно выполнил от 70% до 89% тестовых заданий**. |
| «удовлетворительно» «зачтено» | – обучающийся усвоил только основной программный материал, по существу излагает его, опираясь на знания только основной литературы; – допускает несущественные ошибки и неточности; – испытывает затруднения в практическом применении знаний направления; – слабо аргументирует научные положения; – затрудняется в формулировании выводов и обобщений; – частично владеет системой специализированных понятий. – правильно выполнил от 51% до 69% тестовых заданий**. |
| «неудовлетворительно» «не зачтено» | – обучающийся не усвоил значительной части программного материала; – допускает существенные ошибки и неточности при рассмотрении проблем в конкретном направлении; – испытывает трудности в практическом применении знаний; – не может аргументировать научные положения; – не формулирует выводов и обобщений. – правильно выполнил менее 51% тестовых заданий**. |

10.3. Типовые контрольные задания или иные материалы.

Вопросы (задачи) для экзамена представлены в таблице 15.

Таблица 15 – Вопросы (задачи) для экзамена

| № п/п | Перечень вопросов (задач) для экзамена | Код индикатора |
|-------|---|------------------------|
| 1 | Объектно-ориентированный подход к разработке ПО. Основные понятия, принципы, особенности и достоинства. | ОПК-2.3.1 ОПК-2.У.1 |
| 2 | Объектно-ориентированный анализ. Основные модели | ОПК-2.3.1 ОПК-2.У.1 |
| 3 | Объектно-ориентированное проектирование. Основные этапы и средства. | ОПК-2.3.1 ОПК-2.У.1 |
| 4 | Язык UML. Основные понятия, обозначения, диаграммы. Взаимосвязь диаграмм. | ОПК-2.3.1 ОПК-2.У.1 |

| | | |
|----|--|------------------------|
| 5 | Диаграмма вариантов использования. Основные элементы и их свойства. Пример. | ОПК-2.В.1 |
| 6 | Диаграмма классов. Основные элементы и их свойства. Пример. | ОПК-2.В.1 |
| 7 | Диаграмма последовательностей. Основные элементы и их свойства. Пример | ОПК-2.В.1 |
| 8 | Диаграмма кооперации. Основные элементы и их свойства. Пример. | ОПК-2.В.1 |
| 9 | Диаграммы состояний и видов деятельности. Основные элементы и их свойства. Пример. | ОПК-2.В.1 |
| 10 | Диаграмма пакетов. Основные элементы и их свойства. Пример. | ОПК-2.В.1 |
| 11 | Диаграмма размещения. Основные элементы и их свойства. Пример. | ОПК-2.В.1 |
| 12 | Автоматическая генерация кодов программ. Возможности и ограничения. Пример реализации. | ОПК-2.3.1 ОПК-2.У.1 |
| 13 | Реинжиниринг. Примеры использования. Пример реализации. | ОПК-2.3.1 ОПК-2.У.1 |
| 14 | CASE-технологии. Сравнительный анализ. | ОПК-2.3.1 ОПК-2.У.1 |
| 15 | CASE-средства. Классификация CASE-средств. Примеры современных CASE- средств и их возможности. | ОПК-2.3.1 ОПК-2.У.1 |
| 16 | Пример структуры типового CASE-средства | ОПК-2.3.1 ОПК-2.У.1 |
| 17 | Обзор шаблонов проектирования, назначение | ОПК-2.3.1 ОПК-2.У.1 |
| 18 | Порождающие шаблоны для языка C++ | ОПК-2.3.1 ОПК-2.У.1 |
| 19 | Структурные шаблоны для языка C++ | ОПК-2.3.1 ОПК-2.У.1 |
| 20 | Поведенческие шаблоны для языка C++ | ОПК-2.3.1 ОПК-2.У.1 |

Вопросы (задачи) для зачета / дифф. зачета представлены в таблице 16.
Таблица 16 – Вопросы (задачи) для зачета / дифф. зачета

| № п/п | Перечень вопросов (задач) для зачета / дифф. зачета | Код индикатора |
|-------|--|------------------------|
| 1 | Эволюция вычислительной техники и задач Технологии программирования. Этапы развития методологий проектирования ПО. | ОПК-2.3.1 ОПК-2.У.1 |
| 2 | Жизненный цикл программного обеспечения. Модели ЖЦ ПО. | ОПК-2.3.1 ОПК-2.У.1 |
| 3 | Информационные системы. Определение, основные задачи и цели создания. | ОПК-2.3.1 ОПК-2.У.1 |
| 4 | Классификация современных методологий анализа и проектирования ИС. | ОПК-2.3.1 ОПК-2.У.1 |
| 5 | Сущность структурного подхода к разработке ПО ИС. | ОПК-2.3.1 ОПК-2.У.1 |

| | | |
|----|---|-------------------------------------|
| 6 | Структурный анализ. Определения, основные этапы и средства структурного анализа. | ОПК-2.3.1 ОПК-2.У.1 |
| 7 | Диаграммы потоков данных. Определения, основные элементы, этапы разработки. | ОПК-2.3.1 ОПК-2.У.1 |
| 8 | Построение словаря данных. Способы описания структур данных | ОПК-2.3.1 ОПК-2.У.1 |
| 9 | Методы задания спецификаций процессов. Определения, структура спецификации, правила описания. Сравнение методов спецификации процессов. | ОПК-2.В.1 |
| 10 | Методы задания спецификаций процессов. Структурный естественный язык (псевдокод). | ОПК-2.В.1 |
| 11 | Методы задания спецификаций процессов. FLOW-формы, структурограммы. | ОПК-2.В.1 |
| 12 | Методы задания спецификаций процессов. Деревья и таблицы решений | ОПК-2.В.1 |
| 13 | Проектирование Баз данных. ER-диаграммы. Основные элементы и их свойства. | ОПК-2.3.1 ОПК-2.У.1 |
| 14 | Нормализация ER-диаграммы ИС. 1, 2, 3 нормальные формы. | ОПК-2.В.1 |
| 15 | Нормализация ER-диаграммы ИС. Устранение связей типа М:М. | ОПК-2.В.1 |
| 16 | Разработка структуры Базы данных. Табличные формы БД и их связь с ER- диаграммой. | ОПК-2.3.1 ОПК-2.У.1 |
| 17 | Тестирование ПО | ОПК-2.3.1 ОПК-2.У.1 |
| 18 | Основные принципы ООП и их реализация в языке С++ | ОПК-2.3.1 ОПК-2.У.1 |
| 19 | Уровни доступа к элементам класса в языке С++ | ОПК-2.3.1 ОПК-2.У.1 |
| 20 | Структура библиотеки STL языка С++ | ОПК-2.3.1 ОПК-2.У.1 |
| 21 | Библиотека STL: контейнеры | ОПК-2.3.1 ОПК-2.У.1 |
| 22 | Библиотека STL: итераторы | ОПК-2.3.1 ОПК-2.У.1 |
| 23 | Библиотека STL: алгоритмы | ОПК-2.3.1 ОПК-2.У.1 |
| 24 | Виды стандартов для программного обеспечения | ОПК-4.3.1 ОПК-4.У.1 ОПК-4.В.1 |
| 25 | Виды документации программного обеспечения | ОПК-4.3.1 ОПК-4.У.1 ОПК-4.В.1 |

Перечень тем для выполнения курсового проекта/ курсовой работы представлены в таблице 17.

Таблица 17 – Перечень тем для выполнения курсового проекта / курсовой работы

| № п/п | Примерный перечень тем для выполнения курсового проекта/ курсовой работы |
|-------|--|
| | Учебным планом не предусмотрено |

Вопросы для проведения промежуточной аттестации в виде тестирования представлены в таблице 18.

Таблица 18 – Примерный перечень вопросов для тестов

| № п/п | Примерный перечень вопросов для тестов | Код индикатора | | | | | | | | |
|-------|---|----------------|--|---|--|---|-------------------|---|--|--|
| 1 | <p>Инструкция: выберите один ответ</p> <p>Как называется период от момента появления идеи создания некоторого программного обеспечения до момента завершения его поддержки фирмой- разработчиком или фирмой, выполняющей сопровождение?</p> <ol style="list-style-type: none"> 1. Эпоха 2. Жизненный план 3. Существование 4. Жизненный цикл | ОПК-2.3.1 | | | | | | | | |
| 2 | <p>Инструкция: выберите несколько ответов</p> <p>Какие из перечисленных терминов являются названиями моделей жизненного цикла программного обеспечения?</p> <ol style="list-style-type: none"> 1. Спиральная 2. Водопроводная 3. Каскадная 4. Поточковая 5. Постоянная 6. Ключевая | ОПК-2.3.1 | | | | | | | | |
| 3 | <p>Инструкция:</p> <p>Для каждой диаграммы языка UML из левого столбца подберите фразу из правого столбца, которая характеризует эту диаграмму.</p> <table border="1" data-bbox="363 1402 1121 2051"> <tbody> <tr> <td data-bbox="363 1402 432 1816">А</td> <td data-bbox="432 1402 751 1816">Диаграмма вариантов использования (use cases)</td> <td data-bbox="751 1402 820 1816">1</td> <td data-bbox="820 1402 1121 1816">Показывает последовательность событий, которые генерируются в системе при выполнении определенного сценария варианта использования</td> </tr> <tr> <td data-bbox="363 1816 432 2051">В</td> <td data-bbox="432 1816 751 2051">Диаграмма пакетов</td> <td data-bbox="751 1816 820 2051">2</td> <td data-bbox="820 1816 1121 2051">Показывает аппаратные и программные компоненты системы</td> </tr> </tbody> </table> | А | Диаграмма вариантов использования (use cases) | 1 | Показывает последовательность событий, которые генерируются в системе при выполнении определенного сценария варианта использования | В | Диаграмма пакетов | 2 | Показывает аппаратные и программные компоненты системы | ОПК-2.В.1 ОПК-4.3.1 ОПК-4.У.1 ОПК-4.В.1 |
| А | Диаграмма вариантов использования (use cases) | 1 | Показывает последовательность событий, которые генерируются в системе при выполнении определенного сценария варианта использования | | | | | | | |
| В | Диаграмма пакетов | 2 | Показывает аппаратные и программные компоненты системы | | | | | | | |

| | | | | | |
|---|--|---|---|--|------------------------|
| | С | Диаграмма последовательностей | 3 | Представляет ожидаемое поведение системы с точки зрения пользователя | |
| | D | Диаграмма размещения (англ. deployment) | 4 | Позволяет разбить большую диаграмму на части. | |
| 4 | <p>Инструкция: Запишите лексемы, используемые для объявления словаря в программе на языке C++, в правильном порядке:</p> <ol style="list-style-type: none"> 1. тип данных ключа 2. map 3. тип данных значения 4. > 5. имя словаря 6. < 7. , | | | | ОПК-2.В.1 |
| 5 | <p>Инструкция: Прочитайте текст и запишите развернутый обоснованный ответ.</p> <p>Что показывает контекстная диаграмма в рамках иерархии диаграмм потоков данных и какие элементы содержит?</p> | | | | ОПК-2.В.1 ОПК-4.В.1 |
| 6 | <p>Инструкция: выберите один ответ</p> <p>Какой графический элемент используется в flow-форме?</p> <ol style="list-style-type: none"> 1. Волнистая линия 2. Стрелка 3. Прямоугольник 4. Трапеция | | | | ОПК-2.В.1 ОПК-4.В.1 |
| 7 | <p>Инструкция: Запишите лексемы, используемые для объявления вектора в программе на языке C++, в правильном порядке:</p> <ol style="list-style-type: none"> 1. тип данных значения 2. vector 3. > 4. имя вектора 5. < | | | | ОПК-2.В.1 |

| | | | | | | |
|----|---|-----------|--|---|--|-----------|
| 8 | <p>Инструкция: Запишите в правильном порядке лексемы, используемые для объявления итератора в программе на языке C++:</p> <ol style="list-style-type: none"> 1. iterator 2. :: 3. описание контейнерного класса 4. имя итератора | ОПК-2.В.1 | | | | |
| 9 | <p>Инструкция: выберите несколько ответов</p> <p>Какие из перечисленных ключевых слов управляют доступом к элементам класса в языке C++?</p> <ol style="list-style-type: none"> 1. new 2. public 3. private 4. void 5. protected | ОПК-2.В.1 | | | | |
| 10 | <p>Инструкция: выберите один ответ</p> <p>Как называется процесс, который сокращает избыточность хранения и устраняет аномалии в базах данных?</p> <ol style="list-style-type: none"> 1. Верификация 2. Нормализация 3. Стабилизация 4. Оптимизация | ОПК-2.В.1 | | | | |
| 11 | <p>Инструкция: выберите один ответ</p> <p>Как называется группа шаблонов (паттернов) проектирования, предназначенных для создания новых объектов?</p> <ol style="list-style-type: none"> 1. Завершающие 2. Структурные 3. Оптимизирующие 4. Порождающие | ОПК-2.В.1 | | | | |
| 12 | <p>Инструкция:</p> <p>Для каждой связи диаграммы классов языка UML из левого столбца подберите фразу из правого столбца, которая характеризует эту связь.</p> <table border="1" data-bbox="363 1733 1235 1986"> <tr> <td data-bbox="363 1733 443 1986">А</td> <td data-bbox="443 1733 746 1986">Агрегация</td> <td data-bbox="746 1733 826 1986">1</td> <td data-bbox="826 1733 1235 1986">Отношение «часть-целое» между двумя объектами, при котором контейнер управляет существованием своего содержимого. При уничтожении контейнера</td> </tr> </table> | А | Агрегация | 1 | Отношение «часть-целое» между двумя объектами, при котором контейнер управляет существованием своего содержимого. При уничтожении контейнера | ОПК-2.В.1 |
| А | Агрегация | 1 | Отношение «часть-целое» между двумя объектами, при котором контейнер управляет существованием своего содержимого. При уничтожении контейнера | | | |

| | | | | |
|----|---|------------|--------------------------------|---|
| | | | содержимое также уничтожается. | |
| | В | Обобщение | 2 | отношение «часть-целое» между двумя равноправными объектами, которые существуют независимо. При уничтожении контейнера его содержимое остается. |
| | С | Композиция | 3 | Связь между интерфейсом и классом. |
| | Д | Реализация | 4 | Отношение, показывающее, что один класс получает все элементы другого класса. |
| 13 | Инструкция: выберите несколько ответов | | | ОПК-2.В.1 |
| | <p>Выберите названия типов связей (отношений) между классами в диаграмме классов языка UML</p> <ol style="list-style-type: none"> 1. привязанность 2. зависимость 3. нейтральность 4. ассоциация 5. защищенность | | | |
| 14 | Инструкция: Прочитайте текст и запишите развернутый обоснованный ответ. | | | ОПК-2.У.1 |
| | Объясните, почему исчерпывающее (полное) тестирование обычно невозможно. | | | |
| 15 | Инструкция: выберите несколько ответов | | | ОПК-2.В.1 |
| | <p>Выберите названия методов тестирования программного обеспечения</p> <ol style="list-style-type: none"> 1. красный ящик 2. дырявый ящик 3. белый ящик 4. картонный ящик 5. черный ящик | | | |
| 16 | Инструкция: | | | ОПК-2.В.1 |
| | Для каждого вида тестирования из левого столбца подберите соответствующую фразу из правого столбца. | | | |

| | | | | | |
|----|---|--------------------|---|---|--|
| | A | Позитивное | 1 | Используются программные средства для выполнения тестов и проверки их результатов | |
| | B | Автоматизированное | 2 | Использование почти готовой версии программного продукта добровольцами | |
| | C | Бета-тестирование | 3 | Тестирование изолированных частей программы | |
| | D | Модульное | 4 | В тесте используются только корректные данные. | |
| 17 | <p>Инструкция: выберите один ответ</p> <p>Какой графический символ используется для обозначения варианта использования на диаграммах языка UML?</p> <p>1. Овал 2. Прямоугольник 3. Ромб 4. Круг</p> | | | | ОПК-2.В.1 ОПК-4.У.1 |
| 18 | <p>Инструкция: выберите несколько ответов</p> <p>Выберите названия диаграмм языка UML.</p> <p>1. диаграмма коробок 2. диаграмма деятельности 3. диаграмма пакетов 4. диаграмма компонентов 5. контекстная диаграмма</p> | | | | ОПК-2.В.1 ОПК-4.3.1 ОПК-4.У.1 ОПК-4.В.1 |
| 19 | <p>Инструкция: выберите один ответ</p> <p>Как называется класс, который содержит или наследует хотя бы одну чистую виртуальную функцию (язык C++)?</p> <p>1. Контейнер 2. Конструктор 3. Шаблон 4. Абстрактный</p> | | | | ОПК-2.В.1 |
| 20 | <p>Инструкция: выберите несколько ответов</p> <p>Какие шаблоны можно создать в программе на языке C++?</p> <p>1. отношения 2. деятельности</p> | | | | ОПК-2.В.1 |

| | | |
|----|---|-------------------------------------|
| | 3. программы 4. функции 5. класса | |
| 21 | Инструкция: Прочитайте текст и запишите развернутый обоснованный ответ. Какие требования предъявляются к технической документации? | ОПК-4.3.1 |
| 22 | Инструкция: выберите несколько ответов Что должен устанавливать стандарт проектной документации? 1. Состав и структуру документации на каждой стадии проектирования; 2. Состав программных модулей 3. Требования к оформлению 4. Словарь проекта | ОПК-4.3.1 ОПК-4.У.1 ОПК-4.В.1 |

10.4. Методические материалы, определяющие процедуры оценивания индикаторов, характеризующих этапы формирования компетенций, содержатся в локальных нормативных актах ГУАП, регламентирующих порядок и процедуру проведения текущего контроля успеваемости и промежуточной аттестации обучающихся ГУАП.

11. Методические указания для обучающихся по освоению дисциплины

Целью дисциплины «Технология программирования» является получение студентами необходимых знаний, умений и навыков в области проектирования информационных систем в различных прикладных областях с использованием современных методологий и технологических средств проектирования.

Одной из целей является приобрести умение провести декомпозицию крупной прикладной задачи на более мелкие, с целью организации работы коллектива разработчиков, а также знать современные методики и уметь оценить качество разработанного программного обеспечения, и, используя различные модели ЖЦ ПО, знать возможности организации эффективной работы такого коллектива разработчиков. Важным также является получение студентами практических навыков работы с CASE-средствами различного уровня.

11.1. Методические указания для обучающихся по освоению лекционного материала

Основное назначение лекционного материала – логически стройное, системное, глубокое и ясное изложение учебного материала. Назначение современной лекции в рамках дисциплины не в том, чтобы получить всю информацию по теме, а в освоении фундаментальных проблем дисциплины, методов научного познания, новейших достижений научной мысли. В учебном процессе лекция выполняет методологическую, организационную и информационную функции. Лекция раскрывает понятийный аппарат конкретной области знания, её проблемы, дает цельное представление о дисциплине, показывает взаимосвязь с другими дисциплинами.

Планируемые результаты при освоении обучающимися лекционного материала:

- получение современных, целостных, взаимосвязанных знаний в области проектирования ИС;
- развитие профессионально–деловых качеств и самостоятельного творческого мышления;
- получение знаний о современном уровне развития науки «Технология программирования», а также о прогнозе и потребностях ее развития на ближайшие годы;
- получение практического понимания всех используемых терминов и понятий.

Лекционные материалы доступны в электронном виде на сервере кафедры №43 по адресу: \\Dcbm\учебные пособия\Технология программирования

11.2. Методические указания для обучающихся по выполнению лабораторных работ

В ходе выполнения лабораторных работ обучающийся должен развить творческие навыки и умение переходить от практической прикладной проблемы к научной, формализованной постановке задачи проектирования ИС, овладеть современными методиками проектирования, а также получить навыки работы с современными пакетами программ, позволяющими частично автоматизировать процесс проектирования ПО ИС. Выполнение лабораторных работ состоит из аналитической и проектной частей при разработке различного типа диаграмм, а также экспериментально-практической при использовании современных CASE-средств.

Выполнение лабораторных работ обучающимся является неотъемлемой частью изучения дисциплины, определяемой учебным планом, и относится к средствам, обеспечивающим решение следующих основных задач обучающегося:

- приобретение навыков исследования процессов, явлений и объектов, изучаемых в рамках данной дисциплины;
- закрепление, развитие и детализация теоретических знаний, полученных на лекциях.

Задание и требования к проведению лабораторных работ

Лабораторные работы в основном связаны с созданием диаграмм того или иного вида. Требования содержат необходимое количество элементов на диаграмме. Также ряд лабораторных работ содержат задания на написание программы на языке высокого уровня.

Структура и форма отчета о лабораторной работе

Отчет должен содержать следующие разделы: титульный лист, цель работы, постановка задачи, результаты выполнения заданий по лабораторной работе

Требования к оформлению отчета о лабораторной работе

Отчет о выполнении лабораторной работы должен быть представлен в электронном виде (файл). В отчет включается диаграмма и комментарии к ней.

Методические указания по выполнению лабораторных работ доступны в электронном виде на сервере кафедры №43 по адресу: \\Dcbm\учебные пособия\Технология программирования

11.3. Методические указания для обучающихся по прохождению самостоятельной работы

В ходе выполнения самостоятельной работы, обучающийся выполняет работу по заданию и при методическом руководстве преподавателя, но без его непосредственного участия.

В процессе выполнения самостоятельной работы, у обучающегося формируется целесообразное планирование рабочего времени, которое позволяет им развивать умения и навыки в усвоении и систематизации приобретаемых знаний, обеспечивает высокий уровень успеваемости в период обучения, помогает получить навыки повышения профессионального уровня.

Методическими материалами, направляющими самостоятельную работу обучающихся являются:

- учебно-методический материал по дисциплине;
- методические указания по выполнению контрольных работ (для обучающихся по заочной форме обучения).

11.4. Методические указания для обучающихся по прохождению текущего контроля успеваемости.

Текущий контроль успеваемости предусматривает контроль качества знаний обучающихся, осуществляемого в течение семестра с целью оценивания хода освоения дисциплины.

Текущий контроль успеваемости заключается в устной защите лабораторных работ. На защите обучающийся должен: прокомментировать все термины, встречающиеся в тексте отчета; объяснить алгоритм решения задачи; описать инструменты, с помощью которых был достигнут результат; по просьбе преподавателя внести изменения в текст программы, чтобы получить решение измененной задачи. Баллы, полученные за выполнение и защиту каждой работы, суммируются и учитываются при выставлении итоговой оценки.

11.5. Методические указания для обучающихся по прохождению промежуточной аттестации.

Промежуточная аттестация обучающихся предусматривает оценивание промежуточных и окончательных результатов обучения по дисциплине. Она включает в себя:

- экзамен – форма оценки знаний, полученных обучающимся в процессе изучения всей дисциплины или ее части, навыков самостоятельной работы, способности применять их для решения практических задач. Экзамен, как правило, проводится в период экзаменационной сессии и завершается аттестационной оценкой «отлично», «хорошо», «удовлетворительно», «неудовлетворительно».

- дифференцированный зачет – это форма оценки знаний, полученных обучающимся при изучении дисциплины, при выполнении курсовых проектов, курсовых работ, научно-исследовательских работ и прохождении практик с аттестационной оценкой «отлично», «хорошо», «удовлетворительно», «неудовлетворительно».

Оценка за дифференцированный зачет и экзамен выставляется на основе суммы баллов, которые получены за контрольную работу, лабораторные работы и устный ответ на вопросы (см. Таблицы 15 и 16).

Лист внесения изменений в рабочую программу дисциплины

| Дата внесения изменений и дополнений. Подпись внесшего изменения | Содержание изменений и дополнений | Дата и № протокола заседания кафедры | Подпись зав. кафедрой |
|---|-----------------------------------|--------------------------------------|-----------------------|
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |