

Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное образовательное  
учреждение высшего образования  
«Комсомольский-на-Амуре государственный университет»

УТВЕРЖДАЮ

Декан факультета

компьютерных технологий

Григорьев Я.Ю.

«02» 06 2021 г.

## РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

### Технологии и методы программирования

Направление подготовки	01.03.04 "Прикладная математика"
Направленность (профиль) образовательной программы	Математическое и компьютерное моделирование
Квалификация выпускника	бакалавр
Год начала подготовки (по учебному плану)	2022
Форма обучения	очная
Технология обучения	традиционная

Курс	Семестр	Трудоемкость, з.е.
2	3	5

Вид промежуточной аттестации	Обеспечивающее подразделение
Экзамен	Кафедра «Проектирование, управление и разработка информационных систем»

Разработчик рабочей программы:

Старший преподаватель

(должность, степень, ученое звание)



(подпись)

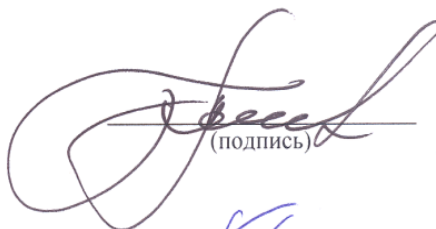
Александров С.Ю.

(ФИО)

СОГЛАСОВАНО:

Заведующий кафедрой ПУРИС

(наименование кафедры)



(подпись)

Тихомиров В.А.

(ФИО)

Заведующий выпускающей  
кафедрой ПМ

(наименование кафедры)



(подпись)

Григорьева А.Л.

(ФИО)

## 1 Общие положения

Рабочая программа дисциплины «Технологии и методы программирования» составлена в соответствии с требованиями федерального государственного образовательного стандарта, утвержденного приказом Министерства образования и науки Российской Федерации № 11 от 10.01.2018, и основной профессиональной образовательной программы подготовки «Математическое и компьютерное моделирование» по направлению 01.03.04 "Прикладная математика.

Практическая подготовка реализуется на основе профессионального стандарта 06.022 Системный аналитик. Обобщенная трудовая функция: С. Концептуальное, функциональное и логическое проектирование систем среднего и крупного масштаба и сложности..

Задачи дисциплины	<ul style="list-style-type: none"><li>• изучить базовые понятия теории множеств;</li><li>• изучить теорию булевой алгебры;</li><li>• изучить базовые понятия математической логики и исчисления высказываний;</li><li>• изучить базовые понятия нечеткой логики;</li><li>• изучить базовые понятия теории алгоритмов..</li></ul>
Основные разделы / темы дисциплины	<ol style="list-style-type: none"><li>1. Множества и отношения</li><li>2. Булевы функции</li><li>3. Исчисление высказываний</li><li>4. Нечеткая логика</li><li>5. Алгоритмы и рекурсивные функции</li></ol>

## 2 Перечень планируемых результатов обучения по дисциплине (модулю), соотнесенных с индикаторами достижения компетенций

Процесс изучения дисциплины «Технологии и методы программирования» направлен на формирование следующих компетенций в соответствии с ФГОС ВО и основной образовательной программой (таблица 1):

Таблица 1 – Компетенции и индикаторы их достижения

Код по ФГОС	Индикаторы достижения	Планируемые результаты обучения по дисциплине
<b>Общепрофессиональные</b>		
ОПК-4. Способен разрабатывать и использовать современные методы и программные средства информационно-коммуникационных технологий	ОПК-4.1 Знает основные принципы работы современного программного обеспечения, принципы использования современных методов и программных средств информационно-коммуникационных технологий ОПК-4.2 Умеет разрабатывать и использовать современные методы и программные средства информационно-коммуникационных технологий ОПК-4.3 Владеет навыками разработки и использования современ-	Знает основные языки программирования и работы с базами данных, операционные системы и оболочки, современные программные среды разработки информационных систем и технологий Умеет применять языки программирования и работы с базами данных, современные программные среды разработки информационных систем и технологий для автоматизации бизнес- процессов, решения

Код по ФГОС	Индикаторы достижения	Планируемые результаты обучения по дисциплине
	ных методов и программных средств информационно-коммуникационных технологий	прикладных задач различных классов, ведения баз данных и информационных хранилищ Владеет навыками программирования, отладки и тестирования прототипов программно-технических комплексов задач

### **3 Место дисциплины (модуля) в структуре образовательной программы**

Дисциплина «Технологии и методы программирования» изучается на 3 курсе в 5 семестре.

Дисциплина входит в состав блока 1 «Дисциплины (модули)» и относится к обязательной части.

Для освоения дисциплины необходимы знания, умения, навыки и (или) опыт практической деятельности, сформированные в процессе изучения дисциплин / практик: «Языки программирования».

Знания, умения и навыки, сформированные при изучении дисциплины «Технологии и методы программирования», будут востребованы при изучении последующих дисциплин «Операционные системы», «Сети и телекоммуникации», «Интернет-технологии», «Учебная практика (ознакомительная практика)».

Дисциплина «Технологии и методы программирования» частично реализуется в форме практической подготовки. Практическая подготовка организуется путем выполнения лабораторных работ, выполнения расчётно-графической работы.

Дисциплина «Технологии и методы программирования» в рамках воспитательной работы направлена на формирование у обучающихся умения самостоятельно мыслить, развивает профессиональные умения

### **4 Объем дисциплины (модуля) в зачетных единицах с указанием количества академических часов, выделенных на контактную работу обучающихся с преподавателем (по видам учебных занятий) и на самостоятельную работу обучающихся**

Общая трудоемкость (объем) дисциплины составляет 5 з.е., 180 акад. час.

Распределение объема дисциплины (модуля) по видам учебных занятий представлено в таблице 2.

Таблица 2 – Объем дисциплины (модуля) по видам учебных занятий

Объем дисциплины	Всего академических часов
Общая трудоемкость дисциплины	180
<b>Контактная аудиторная работа обучающихся с преподавателем (по видам учебных занятий), всего</b>	38
В том числе:	
<b>занятия лекционного типа</b> (лекции и иные учебные занятия, предусматривающие преимущественную передачу учебной информации педагогическими работниками)	16
<b>занятия семинарского типа</b> (семинары, практические занятия, практикумы, лабораторные работы, коллоквиумы и иные аналогичные занятия)	32
<b>Самостоятельная работа обучающихся и контактная работа</b> , включающая групповые консультации, индивидуальную работу обучающихся с преподавателями (в том числе индивидуальные консультации); взаимодействие в электронной информационно-образовательной среде вуза	96
Промежуточная аттестация обучающихся – Экзамен	36

**5 Содержание дисциплины (модуля), структурированное по темам (разделам) с указанием отведенного на них количества академических часов и видов учебной работы**

Таблица 3 – Структура и содержание дисциплины (модуля)

Наименование разделов, тем и содержание материала	Виды учебной работы, включая самостоятельную работу обучающихся и трудоемкость (в часах)			
	Контактная работа преподавателя с обучающимися			СРС
	Лекции	Семинарские (практические занятия)	Лабораторные занятия	
<b>Раздел 1. Объектно-ориентированное программирование</b>				
<b>Тема 1. Классы</b> Описание класса. Описание объектов. Конструкторы объектов. Статические элементы класса. Дружественные функции. Деструкторы. Операции класса. Указатели на элементы класса. Вложенные и локальные классы.	4	-	4	8
<b>Тема 2. Наследование</b> Формы наследования. Формы вложения.	2	-	2	6

<b>Тема 3. Шаблоны классов</b> Создание шаблона классов. Использование шаблона классов. Специализация шаблона классов.	1	-	2	6
<b>Тема 4. Обработка исключительных ситуаций</b> Принцип обработки исключений. Генерация исключений. Перехват исключений. Список исключений функции. Исключения в конструкторах и деструкторах. Стандартные исключения.	1	-	2	6
<b>Тема 5. Преобразования типов</b> Операции приведения типов. Динамическое определение типа.	1	-	2	6
<b>Раздел 2. Стандартная библиотека шаблонов</b>				
<b>Тема 6. Поточковые классы</b> Стандартные потоки. Форматирование данных. Методы обмена с потоками. Состояние потока. Файловые потоки. Строковые потоки. Прямой доступ. Перегрузка операций ввода-вывода.	1	-	4	8
<b>Тема 7. Строки и регулярные выражения</b> Строки. Регулярные выражения.	1	-	2	14
<b>Тема 8. Контейнерные классы</b> Последовательные контейнеры. Ассоциативные контейнеры.	1	-	4	8
<b>Тема 9. Итераторы и функторы</b> Итераторы. Функторы.	1	-	2	6
<b>Тема 10. Алгоритмы</b> Немодифицирующие операции с последовательностями. Модифицирующие операции с последовательностями. Алгоритмы связанные с сортировкой.	1	-	4	8
<b>Тема 11. Средства для численных расчетов</b> Обобщенные численные алгоритмы. Класс <code>valarray</code> . Комплексные числа. Генерация случайных последовательностей.	1	-	2	14
<b>Тема 12. Стандарты языка и сторонние библиотеки</b> Стандарты и совместимость. Библиотеки Boost.	1	-	2	6

<b>ИТОГО по дисциплине</b>	<b>16</b>		<b>32</b>	<b>96</b>
--------------------------------	-----------	--	-----------	-----------

## **6 Внеаудиторная самостоятельная работа обучающихся по дисциплине (модулю)**

При планировании самостоятельной работы студенту рекомендуется руководствоваться следующим распределением часов на самостоятельную работу (таблица 4):

Таблица 4 – Рекомендуемое распределение часов на самостоятельную работу

<b>Компоненты самостоятельной работы</b>	<b>Количество часов</b>
Изучение теоретических разделов дисциплины	16
Подготовка к занятиям семинарского типа	32
Подготовка и оформление РГР	28
	96

## **7 Оценочные средства для проведения текущего контроля и промежуточной аттестации обучающихся по дисциплине (модулю)**

Фонд оценочных средств для проведения текущего контроля успеваемости и промежуточной аттестации представлен в Приложении 1.

Полный комплект контрольных заданий или иных материалов, необходимых для оценивания результатов обучения по дисциплине (модулю), практике хранится на кафедре-разработчике в бумажном и электронном виде.

## **8 Учебно-методическое и информационное обеспечение дисциплины (модуля)**

### **8.1 Основная литература**

1 Немцова, Т. И. Программирование на языке С++ [Электронный ресурс]: учеб. пособие / Т.И. Немцова, С.Ю. Голова, А.И. Терентьев; Под ред. Л.Г. Гагариной. - М.: ИД ФОРУМ: ИНФРА-М, 2012. - 512 с. // ZNANIUM.COM : электронно-библиотечная система. – Режим доступа: <http://znanium.com/catalog.php#>, ограниченный. – Загл. с экрана.

### **8.2 Дополнительная литература**

1. Павловская, Т. А. С/С++. Программирование на языке высокого уровня: учебник для вузов / Т. А. Павловская. – СПб. : Питер, 2010; 2003; 2001. – 460 с.

2. Павловская, Т. А. С++. Объектно-ориентированное программирование: Практикум / Т.А. Павловская, Ю.А. Щупак. — СПб.: Питер, 2006 — 265 с. : ил.– (Серия «Учебное пособие»)

### **8.3 Методические указания для студентов по освоению дисциплины**

Методические указания к выполнению индивидуальных заданий приведены в учебнике [3] из списка основной литературы (раздел 8.2); теоретический материал в учебнике [3] из списка дополнительной литературы (раздел 8.2).

#### **8.4 Современные профессиональные базы данных и информационные справочные системы, используемые при осуществлении образовательного процесса по дисциплине**

1 Электронно-библиотечная система ZNANIUM.COM. Договор ЕП 44 № 003/10 эбс ИКЗ 191272700076927030100100120016311000 от 17 апреля 2019 г.

2 Электронно-библиотечная система IPRbooks. Лицензионный договор № ЕП44 № 001/9 на предоставление доступа к электронно-библиотечной системе IPRbooks ИКЗ 191272700076927030100100090016311000 от 27 марта 2019 г.

3 Электронно-библиотечная система eLIBRARY.RU. Договор № ЕП 44 № 004/13 на оказание услуг доступа к электронным изданиям ИКЗ 91272700076927030100100150016311000 от 15 апреля 2019 г.

4 Информационно-справочные системы «Кодекс»/ «Техэксперт». Соглашение о сотрудничестве № 25/19 от 31 мая 2019 г.

#### **8.5 Лицензионное и свободно распространяемое программное обеспечение, используемое при осуществлении образовательного процесса по дисциплине**

Таблица 7 – Перечень используемого программного обеспечения

Наименование ПО	Реквизиты / условия использования
Microsoft Imagine Premium	Лицензионный договор АЭ223 №008/65 от 11.01.2019
OpenOffice	Свободная лицензия, условия использования по ссылке: <a href="https://www.openoffice.org/license.html">https://www.openoffice.org/license.html</a>
Visual Studio Community 2019 и более поздние версии	Visual Studio Community может использовать неограниченное число пользователей в организации в следующих случаях: в учебных аудиториях, для научных исследований или участия в проектах с открытым кодом.

### **9 Организационно-педагогические условия**

Организация образовательного процесса регламентируется учебным планом и расписанием учебных занятий. Язык обучения (преподавания) — русский. Для всех видов аудиторных занятий академический час устанавливается продолжительностью 45 минут.

При формировании своей индивидуальной образовательной траектории обучающийся имеет право на перезачет соответствующих дисциплин и профессиональных модулей, освоенных в процессе предшествующего обучения, который освобождает обучающегося от необходимости их повторного освоения.

#### **9.1 Образовательные технологии**

Учебный процесс при преподавании курса основывается на использовании традиционных, инновационных и информационных образовательных технологий. Традиционные образовательные технологии представлены лекциями и семинарскими (практическими) занятиями. Инновационные образовательные технологии используются в виде широкого применения активных и интерактивных форм проведения занятий. Информационные образовательные технологии реализуются путем активизации самостоятельной работы студентов в информационной образовательной среде.

#### **9.2 Занятия лекционного типа**

Лекционный курс предполагает систематизированное изложение основных вопросов учебного плана.



На первой лекции лектор обязан предупредить студентов, применительно к какому базовому учебнику (учебникам, учебным пособиям) будет прочитан курс.

Лекционный курс должен давать наибольший объем информации и обеспечивать более глубокое понимание учебных вопросов при значительно меньшей затрате времени, чем это требуется большинству студентов на самостоятельное изучение материала.

### **9.3 Занятия семинарского типа**

Семинарские занятия представляют собой детализацию лекционного теоретического материала, проводятся в целях закрепления курса и охватывают все основные разделы.

Основной формой проведения семинаров является обсуждение наиболее проблемных и сложных вопросов по отдельным темам, а также разбор примеров и ситуаций в аудиторных условиях. В обязанности преподавателя входят: оказание методической помощи и консультирование студентов по соответствующим темам курса.

Активность на семинарских занятиях оценивается по следующим критериям:

- ответы на вопросы, предлагаемые преподавателем;
- участие в дискуссиях;
- выполнение проектных и иных заданий;
- ассистирование преподавателю в проведении занятий.

Ответ должен быть аргументированным, развернутым, не односложным, содержать ссылки на источники.

Доклады и оппонирование докладов проверяют степень владения теоретическим материалом, а также корректность и строгость рассуждений.

Оценивание заданий, выполненных на семинарском занятии, входит в накопленную оценку.

### **9.4 Самостоятельная работа обучающихся по дисциплине (модулю)**

Самостоятельная работа студентов – это процесс активного, целенаправленного приобретения студентом новых знаний, умений без непосредственного участия преподавателя, характеризующийся предметной направленностью, эффективным контролем и оценкой результатов деятельности обучающегося.

Цели самостоятельной работы:

- систематизация и закрепление полученных теоретических знаний и практических умений студентов;
- углубление и расширение теоретических знаний;
- формирование умений использовать нормативную и справочную документацию, специальную литературу;
- развитие познавательных способностей, активности студентов, ответственности и организованности;
- формирование самостоятельности мышления, творческой инициативы, способностей к саморазвитию, самосовершенствованию и самореализации;
- развитие исследовательских умений и академических навыков.

Самостоятельная работа может осуществляться индивидуально или группами студентов в зависимости от цели, объема, уровня сложности, конкретной тематики.

Технология организации самостоятельной работы студентов включает использование информационных и материально-технических ресурсов университета.

Перед выполнением обучающимися внеаудиторной самостоятельной работы преподаватель может проводить инструктаж по выполнению задания. В инструктаж включается:

- цель и содержание задания;
- сроки выполнения;
- ориентировочный объем работы;
- основные требования к результатам работы и критерии оценки;

- возможные типичные ошибки при выполнении.

Инструктаж проводится преподавателем за счет объема времени, отведенного на изучение дисциплины.

Контроль результатов внеаудиторной самостоятельной работы студентов может проходить в письменной, устной или смешанной форме.

Студенты должны подходить к самостоятельной работе как к наиважнейшему средству закрепления и развития теоретических знаний, выработке единства взглядов на отдельные вопросы курса, приобретения определенных навыков и использования профессиональной литературы.

Помещения для самостоятельной работы обучающихся оснащены компьютерной техникой с возможностью подключения к сети «Интернет» и обеспечением доступа в электронную информационно-образовательную среду организации.

### **9.5 Методические указания для обучающихся по освоению дисциплины**

При изучении дисциплины обучающимся целесообразно выполнять следующие рекомендации:

1. Изучение учебной дисциплины должно вестись систематически.
2. После изучения какого-либо раздела по учебнику или конспектным материалам рекомендуется по памяти воспроизвести основные термины, определения, понятия раздела.
3. Особое внимание следует уделить выполнению отчетов по практическим занятиям и индивидуальным комплексным заданиям на самостоятельную работу.
4. Вся тематика вопросов, изучаемых самостоятельно, задается на лекциях преподавателем. Им же даются источники (в первую очередь вновь изданные в периодической научной литературе) для более детального понимания вопросов, озвученных на лекции.

При самостоятельной проработке курса обучающиеся должны:

- просматривать основные определения и факты;
- повторить законспектированный на лекционном занятии материал и дополнить его с учетом рекомендованной по данной теме литературы;
- изучить рекомендованную литературу, составлять тезисы, аннотации и конспекты наиболее важных моментов;
- самостоятельно выполнять задания, аналогичные предлагаемым на занятиях;
- использовать для самопроверки материалы фонда оценочных средств.

## **10 Описание материально-технического обеспечения, необходимого для осуществления образовательного процесса по дисциплине (модулю)**

### **10.1 Учебно-лабораторное оборудование**

Таблица 6 – Перечень оборудования лаборатории

Аудитория	Наименование аудитории (лаборатории)	Используемое оборудование
228-1	Мультимедийная аудитория	Проектор, экран, компьютер/ноутбук
313-5	Учебная лаборатория ФКТ	20 персональных ЭВМ, каждая из которых оснащена процессором Intel(R) Core (TM) i5- и оперативной памятью 8ГБ. Операционная система - Windows 10.

### **10.2 Технические и электронные средства обучения**

### **Лекционные занятия**

Аудитории для лекционных занятий укомплектованы мебелью и техническими средствами обучения, служащими для представления учебной информации большой аудитории (наборы демонстрационного оборудования (проектор, экран, компьютер/ноутбук), учебно-наглядные пособия, тематические иллюстрации).

### **Практические занятия**

Аудитории для практических занятий укомплектованы специализированной мебелью и техническими средствами обучения (проектор, экран, компьютер/ноутбук).

### **Самостоятельная работа.**

Помещения для самостоятельной работы оснащены компьютерной техникой с возможностью подключения к сети «Интернет» и доступом к электронной информационно-образовательной среде КНАГУ:

- читальный зал НТБ КНАГУ;
- компьютерный класс (ауд. 225 корпус № 3).

## **11 Иные сведения**

### **Методические рекомендации по обучению лиц с ограниченными возможностями здоровья и инвалидов**

Освоение дисциплины обучающимися с ограниченными возможностями здоровья может быть организовано как совместно с другими обучающимися, так и в отдельных группах. Предполагаются специальные условия для получения образования обучающимися с ограниченными возможностями здоровья.

Профессорско-педагогический состав знакомится с психолого-физиологическими особенностями обучающихся инвалидов и лиц с ограниченными возможностями здоровья, индивидуальными программами реабилитации инвалидов (при наличии). При необходимости осуществляется дополнительная поддержка преподавания тьюторами, психологами, социальными работниками, прошедшими подготовку ассистентами.

В соответствии с методическими рекомендациями Минобрнауки РФ (утв. 8 апреля 2014 г. N АК-44/05вн) в курсе предполагается использовать социально-активные и рефлексивные методы обучения, технологии социокультурной реабилитации с целью оказания помощи в установлении полноценных межличностных отношений с другими студентами, создании комфортного психологического климата в студенческой группе. Подбор и разработка учебных материалов производятся с учетом предоставления материала в различных формах: аудиальной, визуальной, с использованием специальных технических средств и информационных систем.

Освоение дисциплины лицами с ОВЗ осуществляется с использованием средств обучения общего и специального назначения (персонального и коллективного использования). Материально-техническое обеспечение предусматривает приспособление аудиторий к нуждам лиц с ОВЗ.

Форма проведения аттестации для студентов-инвалидов устанавливается с учетом индивидуальных психофизических особенностей. Для студентов с ОВЗ предусматривается доступная форма предоставления заданий оценочных средств, а именно:

- в печатной или электронной форме (для лиц с нарушениями опорно-двигательного аппарата);
- в печатной форме или электронной форме с увеличенным шрифтом и контрастностью (для лиц с нарушениями слуха, речи, зрения);
- методом чтения ассистентом задания вслух (для лиц с нарушениями зрения).

Студентам с инвалидностью увеличивается время на подготовку ответов на контрольные вопросы. Для таких студентов предусматривается доступная форма предоставления ответов на задания, а именно:

- письменно на бумаге или набором ответов на компьютере (для лиц с нарушениями слуха, речи);
- выбором ответа из возможных вариантов с использованием услуг ассистента (для лиц с нарушениями опорно-двигательного аппарата);
- устно (для лиц с нарушениями зрения, опорно-двигательного аппарата).

При необходимости для обучающихся с инвалидностью процедура оценивания результатов обучения может проводиться в несколько этапов.

## ФОНД ОЦЕНОЧНЫХ СРЕДСТВ

по дисциплине

### Технологии и методы программирования

Направление подготовки	<i>01.03.04 "Прикладная математика"</i>
Направленность (профиль) образовательной программы	<i>Математическое и компьютерное моделирование</i>
Квалификация выпускника	<i>бакалавр</i>
Год начала подготовки (по учебному плану)	<i>2022</i>
Форма обучения	<i>очная</i>
Технология обучения	<i>традиционная</i>

Курс	Семестр	Трудоемкость, з.е.
2	3	5

Вид промежуточной аттестации	Обеспечивающее подразделение
<i>Экзамен</i>	<i>Кафедра «Проектирование, управление и разработка информационных систем»</i>

**1 Перечень планируемых результатов обучения по дисциплине (модулю), соотнесенных с индикаторами достижения компетенций**

Таблица 1 – Компетенции и индикаторы их достижения

Код по ФГОС	Индикаторы достижения	Планируемые результаты обучения по дисциплине
<b>Общепрофессиональные</b>		
ОПК-4. Способен разрабатывать и использовать современные методы и программные средства информационно-коммуникационных технологий	ОПК-4.1 Знает основные принципы работы современного программного обеспечения, принципы использования современных методов и программных средств информационно-коммуникационных технологий ОПК-4.2 Умеет разрабатывать и использовать современные методы и программные средства информационно-коммуникационных технологий ОПК-4.3 Владеет навыками разработки и использования современных методов и программных средств информационно-коммуникационных технологий	Знает основные языки программирования и работы с базами данных, операционные системы и оболочки, современные программные среды разработки информационных систем и технологий Умеет применять языки программирования и работы с базами данных, современные программные среды разработки информационных систем и технологий для автоматизации бизнес-процессов, решения прикладных задач различных классов, ведения баз данных и информационных хранилищ Владеет навыками программирования, отладки и тестирования прототипов программно-технических комплексов задач

Таблица 2 – Паспорт фонда оценочных средств

Контролируемые разделы (темы) дисциплины	Формируемая компетенция	Наименование оценочного средства	Показатели оценки
<b>Тема 1. Классы</b>	ОПК-4	Задание 1. Классы, РГР, Экзамен	Владеет навыками разработки ПО на базе классов
<b>Тема 2. Наследование</b>	ОПК-4	Задание 2. Наследование, РГР, Экзамен	Владеет навыками разработки ПО на базе объектно-ориентированного программирования
<b>Тема 3. Шаблоны классов</b>	ОПК-4	Задание 3. Шаблоны классов и обработка исключительных ситуаций, РГР, Экзамен	Владеет навыками разработки ПО на базе классов Владеет навыками разработки ПО на базе объектно-ориентированного программирования

<b>Тема 4. Обработка исключительных ситуаций</b>	ОПК-4	Задание 3. Шаблоны классов и обработка исключительных ситуаций, РГР, Экзамен	Умеет применять методы и средства проектирования программного обеспечения, структур данных, баз данных, программных интерфейсов
<b>Тема 5. Преобразования типов</b>	ОПК-4	Задание 2. Наследование, РГР, Экзамен	Умеет применять методы и средства проектирования программного обеспечения, структур данных, баз данных, программных интерфейсов
<b>Тема 6. Поточковые классы</b>	ОПК-4	Задание 4. Потoki и контейнеры, РГР, Экзамен	Знает типовые решения, библиотеки программных модулей, шаблоны, классы объектов, используемые при разработке программного обеспечения
<b>Тема 7. Строки и регулярные выражения</b>	ОПК-4	Задание 5. Строки и регулярные выражения, РГР, Экзамен	Знает типовые решения, библиотеки программных модулей, шаблоны, классы объектов, используемые при разработке программного обеспечения
<b>Тема 8. Контейнерные классы</b>	ОПК-4	Задание 6. Контейнеры и алгоритмы, РГР, Экзамен	Знает типовые решения, библиотеки программных модулей, шаблоны, классы объектов, используемые при разработке программного обеспечения
<b>Тема 9. Итераторы и функторы</b>	ОПК-4	Задание 6. Контейнеры и алгоритмы, РГР, Экзамен	Умеет применять методы и средства проектирования программного обеспечения, структур данных, баз данных, программных интерфейсов
<b>Тема 10. Алгоритмы</b>	ОПК-4	Задание 6. Контейнеры и алгоритмы, РГР, Экзамен	Владеет навыками использования библиотек программных модулей, шаблонов, классов объектов при разработке
<b>Тема 11. Средства для численных расчетов</b>	ОПК-4	Задание 7. Класс <code>valarray</code> , РГР, Экзамен	Владеет навыками использования библиотек программных модулей, шаблонов, классов объектов при разработке
<b>Тема 12. Стандарты языка и сторонние библиотеки</b>	ОПК-4	РГР, Экзамен	Знает типовые решения, библиотеки программных модулей, шаблоны, классы объектов, используемые при разработке программного обеспечения

**2 Методические материалы, определяющие процедуры оценивания знаний, умений, навыков и (или) опыта деятельности, характеризующие процесс формирования компетенций**

Методические материалы, определяющие процедуры оценивания знаний, умений, навыков и (или) опыта деятельности, представлены в виде технологической карты дисциплины (таблица 3).

Таблица 3 – Технологическая карта

	<b>Наименование оценочного средства</b>	<b>Сроки выполнения</b>	<b>Шкала оценивания</b>	<b>Критерии оценивания</b>
3 семестр <i>Промежуточная аттестация в форме экзамена</i>				
1	Задание 1 «Классы»	4 неделя	5	0 баллов – задание не выполнено или выполнено не верно 3 балла – задание выполнено с недочетами и не в срок 4 балла – задание выполнено без недочетов и не в срок 5 баллов – задание выполнено без недочетов и в срок
2	Задание 2 «Наследование»	6 неделя	5	0 баллов – задание не выполнено или выполнено не верно 3 балла – задание выполнено с недочетами и не в срок 4 балла – задание выполнено без недочетов и не в срок 5 баллов – задание выполнено без недочетов и в срок



	<b>Наименование оценочного средства</b>	<b>Сроки выполнения</b>	<b>Шкала оценивания</b>	<b>Критерии оценивания</b>
3	Задание 3 «Шаблоны классов и обработка исключительных ситуаций»	8 неделя	5	0 баллов – задание не соответствует варианту, не верно оформлено или отсутствует 3 балла – задание оформлено с недочетами и не в срок 4 балла – задание оформлено без недочетов и не в срок 5 баллов – задание оформлено без недочетов и в срок
4	Задание 4 «Потоки и контейнеры»	10 неделя	5	0 баллов – задание не соответствует варианту, не верно оформлено или отсутствует 3 балла – задание оформлено с недочетами и не в срок 4 балла – задание оформлено без недочетов и не в срок 5 баллов – задание оформлено без недочетов и в срок

	<b>Наименование оценочного средства</b>	<b>Сроки выполнения</b>	<b>Шкала оценивания</b>	<b>Критерии оценивания</b>
5	Задание 5 «Строки и регулярные выражения»	12 неделя	5	0 баллов – за- дание не соот- ветствует вари- анту, не верно оформлено или отсутствует 3 балла – зада- ние оформлено с недочетами и не в срок 4 балла – зада- ние оформлено без недочетов и не в срок 5 баллов – задание оформлено без недочетов и в срок
6	Задание 6 «Контейнеры и алгоритмы»	14 неделя	5	0 баллов – за- дание не соот- ветствует вари- анту, не верно оформлено или отсутствует 3 балла – зада- ние оформлено с недочетами и не в срок 4 балла – зада- ние оформлено без недочетов и не в срок 5 баллов – задание оформлено без недочетов и в срок

	<b>Наименование оценочного средства</b>	<b>Сроки выполнения</b>	<b>Шкала оценивания</b>	<b>Критерии оценивания</b>
7	Задание 7 «Класс valarray»	15 неделя	5	0 баллов – задание не соответствует варианту, не верно оформлено или отсутствует 3 балла – задание оформлено с недочетами и не в срок 4 балла – задание оформлено без недочетов и не в срок 5 баллов – задание оформлено без недочетов и в срок
8	Расчетно-графическая работа (7 разделов)	16 неделя	35	0 баллов – задание не соответствует варианту, не верно оформлено или отсутствует 3 балла – задание оформлено с недочетами и не в срок 4 балла – задание оформлено без недочетов и не в срок 5 баллов – задание оформлено без недочетов и в срок
Текущий контроль:		-	70 баллов	-

	<b>Наименование оценочного средства</b>	<b>Сроки выполнения</b>	<b>Шкала оценивания</b>	<b>Критерии оценивания</b>
	Экзамен (6 вопросов)	сессия	30	0 баллов – ответ на вопрос билета отсутствует или неверен 3 балла – дан не полный ответ, допущены ошибки 4 балла – дан полный ответ, допущены неточности 5 баллов – дан полный ответ, приведены примеры
	Экзамен: -	30 баллов	-	
	ИТОГО: -	100 баллов	-	
<p><b>Критерии оценки результатов обучения по дисциплине:</b>  0 – 64 % от максимально возможной суммы баллов – «неудовлетворительно» (недостаточный уровень для промежуточной аттестации по дисциплине);  65 – 74 % от максимально возможной суммы баллов – «удовлетворительно» (пороговый (минимальный) уровень);  75 – 84 % от максимально возможной суммы баллов – «хорошо» (средний уровень);  85 – 100 % от максимально возможной суммы баллов – «отлично» (высокий (максимальный) уровень)</p>				

### **3 Типовые контрольные задания или иные материалы, необходимые для оценки знаний, умений, навыков и (или) опыта деятельности, характеризующие процесс формирования компетенций в ходе освоения образовательной программы**

#### **3.1 Задания для текущего контроля успеваемости**

**Задание 1.** Описать класс в нотации UML и разработать программу, демонстрирующую все возможности класса. Программа должна содержать меню, позволяющее выполнить проверку всех методов класса.

##### 1. Вариант 1

Описать класс, реализующий стек. Написать программу, использующую этот класс для моделирования Т-образного сортировочного узла на железной дороге.

Программа должна разделять на два направления состав, состоящий из вагонов двух типов (на каждое направление формируется состав из вагонов одного типа). Предусмотреть возможность формирования состава из файла и с клавиатуры.

##### 2. Вариант 2

Описать класс, реализующий бинарное дерево, обладающее возможностями добавления новых элементов, удаления существующих, поиска элемента по ключу, а также последовательного доступа ко всем элементам.

Написать программу, использующую этот класс для представления англо-русского словаря. Программа должна содержать меню, позволяющее выполнить проверку всех методов класса. Предусмотреть возможность создания словаря из файла и с клавиатуры.

**Задание 2.** Реализовать абстрактный базовый класс и производные от него. Во всех классах должно быть хотя бы одно поле, память под которое выделяется динамически. Используя механизм виртуальных методов, продемонстрировать работу с производным классом через указатель на базовый класс. Обеспечить удобные средства получения доступа к состоянию класса. Программа должна содержать меню, позволяющее выполнить проверку всех методов класса.

##### 1. Вариант 1

Создать абстрактный класс File, инкапсулирующий в себе методы Open, Close, Seek, Read, Write, GetPosition и GetLength. Создать производные классы MyDataFile1 и MyDataFile2 — файлы, содержащие в себе данные некоторого определенного типа MyData1 и MyData2, а также заголовки, облегчающие доступ к этим файлам.

Создать класс Folder, содержащий список (или массив) объектов этих классов в динамической памяти. Предусмотреть возможность вывода списка имен и длин файлов. Написать демонстрационную программу, в которой будут использоваться все методы классов.

##### 2. Вариант 2

Создать абстрактный класс Point (точка). На его основе создать классы Colored Point и Line. На основе класса Line создать классы Colored Line и Poly Line (многоугольник). Все классы должны иметь виртуальные методы установки и получения значений всех координат, а также изменения цвета и получения текущего цвета.

Создать класс Picture, содержащий список (или массив) объектов этих классов в динамической памяти. Предусмотреть возможность вывода характеристик объектов списка. Написать демонстрационную программу, в которой будут использоваться все методы классов.

**Задание 3.** Реализовать шаблонный класс. Программа должна содержать меню, позволяющее выполнить проверку работоспособности шаблона класса как для встроенных, так и для пользовательских типов данных. Предусмотреть обработку ошибок с помощью механизма исключительных ситуаций.

1. Вариант 1

Создать шаблон класса Stack. Использовать его при решении задачи варианта 1 задания 1.

2. Вариант 2

Создать шаблон класса BinaryTree. Использовать его при решении задачи варианта 2 задания 1.

**Задание 4.** Выполнить задание 2, используя файловые и стандартные потоки, а также контейнеры. Виды контейнеров выбрать исходя из потребностей задачи. Во всех вариантах использовать работу с файлами и обработку ошибок.

**Задание 5.** Выполнить задание с использованием класса string и регулярных выражений.

1. Вариант 1

Написать программу, которая считывает из текстового файла три предложения и выводит их в обратном порядке.

2. Вариант 2

Написать программу, которая считывает текст из файла и выводит на экран только предложения, содержащие введенное с клавиатуры слово.

**Задание 6.** Выполнить задание с использованием подходящих контейнерных классов и алгоритмов библиотеки. Содержимое контейнеров (не менее миллиона элементов) формировать с помощью генераторов случайных чисел.

### 1. Вариант 1

Найти сумму отрицательных элементов массива.

Найти произведение элементов массива, расположенных между максимальным и минимальным элементами.

Упорядочить элементы массива по возрастанию.

### 2. Вариант 2

Найти сумму положительных элементов массива.

Найти произведение элементов массива, расположенных между максимальным по модулю и минимальным по модулю элементами.

Упорядочить элементы массива по убыванию.

**Задание 7.** Выполнить задания с использованием класса `valarray`, срезов и генераторов случайных чисел.

### 1. Вариант 1

Дана целочисленная прямоугольная матрица. Определить:

- количество строк, не содержащих ни одного нулевого элемента;
- максимальное из чисел, встречающихся в заданной матрице более одного раза.

### 2. Вариант 2.

Дана целочисленная прямоугольная матрица. Определить количество столбцов, не содержащих ни одного нулевого элемента.

Характеристикой строки целочисленной матрицы назовем сумму ее положительных четных элементов. Переставляя строки заданной матрицы, расположить их в соответствии с ростом характеристик.

## ***Расчетно-графическая работа***

1. Выполнить верификацию и тестирование кода, полученного в ходе текущей аттестации при выполнении индивидуальных заданий, путем проведения тестовых запусков программ на заранее подготовленных наборах данных.

2. Описать решения индивидуальных заданий полученные в ходе прохождения текущей аттестации в соответствии с требованиями единой системы программной документации (ЕСПД) и РД 013-2016 «Текстовые студенческие работы. Правила оформления».

Структурными элементами данной работы должны быть:

- титульный лист;
- текст всех заданий (в соответствии с вариантом);
- содержание;
- введение
- основная часть;
- заключение;
- список использованных источников;
- приложения.

Во введении дается краткое описание изучаемой дисциплины, которой посвящена данная работа, а также приводится обзор выполненной работы.

Каждое выполненное индивидуальное задание оформляется отдельным разделом основной части отчета. Каждый раздел включает теоретические сведения по теме раздела с обязательным указанием ссылок на источники информации, в случае её заимствования. Обязательными подразделами каждого отчета являются «Описание программы», «Текст программы», «Программа и методика испытаний». Требования к содержанию подразделов приведены в таблице:

<b>Вид программного документа</b>	<b>Содержание программного документа</b>
Описание программы	Сведения о логической структуре и функционировании программы
Текст программы	Запись программы с необходимыми комментариями
Программа и методика испытаний	Требования, подлежащие проверке при испытании программы, а также порядок и методы их контроля

В заключении приводятся общие выводы и предложения, а также краткое описание проделанной работы; даются практические рекомендации. (1,5 - 2 страницы).

Список литературы состоит из нормативно-правовых актов, учебников и учебных пособий, научных статей, использованных в ходе выполнения индивидуального задания.

Приложения помещают после списка литературы в порядке их отсылки или обращения к ним в тексте. В качестве приложений рекомендуется предоставлять копии документов, бланков договоров, организационно-распорядительных документов, аналитических таблиц, иных документов, иллюстрирующих содержание основной части.

Отчет предварительно оценивается и допускается к защите после проверки его соответствия требованиям, предъявляемым данными методическими указаниями. Защита отчетов организуется в форме собеседования.



**Задания для промежуточной  
аттестации Контрольные  
вопросы к экзамену (семестр 3)**

1. Абстрактные структуры данных.
2. Динамические структуры данных.
3. Описание класса.
4. Описание объектов.
5. Указатель `this`.
6. Конструкторы объектов.
7. Статические элементы класса.
8. Дружественные функции классов.
9. Деструкторы.
10. Операции класса.
11. Указатели на элементы класса.
12. Вложенные и локальные классы.
13. Каноническая форма класса в C++11.
14. Формы наследования.
15. Формы вложения.
16. Создание шаблона классов.
17. Использование шаблона классов.
18. Специализация шаблона классов.
19. Принцип обработки исключений.
20. Генерация исключений.
21. перехват исключений.
22. Список исключений функции.
23. Исключения в конструкторах и деструкторах.
24. Стандартные исключения.
25. Операции приведения типов в C.
26. Операции приведения типов в C++.
27. Динамическое определение типа.
28. Стандартные потоки.
29. Файловые потоки.
30. Строковые потоки.
31. Перегрузка операций ввода-вывода.
32. Строки.
33. Регулярные выражения.
34. Последовательные контейнеры.
35. Ассоциативные контейнеры.
36. Итераторы.
37. Функторы.
38. Немодифицирующие операции с последовательностями.
39. Модифицирующие операции с последовательностями.
40. Алгоритмы, связанные с сортировкой.
41. Обобщенные численные алгоритмы.
42. Класс `valarray`.

***Примерные варианты экзаменационных***

***билетов (семестр 3) Билет №1***

1. Абстрактные структуры данных.
2. Дружественные функции классов.
3. Формы вложения. Список исключений функции.

***Билет №2***

1. Статические элементы класса.
2. Формы наследования.
3. Перехват исключений.

