

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ  
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)

УТВЕРЖДАЮ:

Директор \_\_\_\_\_ В.П.Гергель

« \_\_\_\_ » \_\_\_\_\_ 2018 г.

**Рабочая программа дисциплины (модуля)  
АЛГОРИТМЫ И СТРУКТУРЫ ДАННЫХ**

\_\_\_\_\_  
(наименование дисциплины (модуля))

Уровень высшего образования

**бакалавриат**

\_\_\_\_\_  
(бакалавриат / магистратура / специалитет)

Направление подготовки / специальность

**01.03.02 Прикладная математика и информатика**

\_\_\_\_\_  
(указывается код и наименование направления подготовки / специальности)

Направленность образовательной программы

**Системный анализ, исследование операций и управление**

\_\_\_\_\_  
(указывается профиль / магистерская программа / специализация)

Квалификация (степень)

**бакалавр**

\_\_\_\_\_  
(бакалавр / магистр / специалист)

Форма обучения

**очно-заочная**

\_\_\_\_\_  
(очная / очно-заочная / заочная)

Нижегород

2018

## 1. Место и цели дисциплины в структуре ОПОП

Дисциплина «Алгоритмы и структуры данных» относится к базовой части ОПОП 01.03.02 - Прикладная математика и информатика. Данная дисциплина читается в третьем и четвертом семестрах бакалавриата (Б1.Б.11 – базовая часть). Материал дисциплины опирается на знание курсов «Языки и методы программирования» (1-2 семестры) и «Дискретная математика» (1-2 семестры).

**Целями освоения дисциплины являются:**

*Целями освоения дисциплины являются:*

- углубление знаний в области языков программирования C/C++
- знакомство с основами объектно-ориентированного программирования на языке C++
- изучение базовых структур данных и алгоритмов

## 2. Планируемые результаты обучения по дисциплине, соотнесенные с планируемыми результатами освоения образовательной программы (компетенциями выпускников)

Формируемые компетенции (код компетенции, уровень освоения – при наличии в карте компетенции)	Планируемые результаты обучения по дисциплине (модулю), характеризующие этапы формирования компетенций
<b>ОПК-3</b> <i>Способность к разработке алгоритмических и программных решений в области системного и прикладного программирования, математических, информационных и имитационных моделей, созданию информационных ресурсов глобальных сетей, образовательного контента, прикладных баз данных, тестов и средств тестирования систем и средств на соответствие стандартам и исходным требованиям (начальный (низкий) этап)</i>	<b>ЗНАТЬ 31(ОПК3)</b> основные алгоритмические конструкции и программные решения <b>32(ОПК3)</b> основы построения математических, имитационных и информационных моделей <b>УМЕТЬ У1(ОПК3)</b> разрабатывать программные решения на базе математических, имитационных и информационных моделей <b>У2(ОПК3)</b> создавать ресурсы для глобальной сети, БД <b>У3(ОПК3)</b> создавать тесты для тестирования систем на соответствие требованиям и стандартам <b>ВЛАДЕТЬ В1(ОПК3)</b> навыками тестирования информационных систем
<b>ПК-7</b> <i>Способность к разработке и применению алгоритмических и программных решений в области системного и прикладного программного обеспечения (начальный (низкий) этап)</i>	<b>ЗНАТЬ 31(ПК7)</b> основные способы разработки информационных систем, парадигмы программирования <b>УМЕТЬ У1(ПК7)</b> профессионально разрабатывать и использовать программное обеспечение для поддержки информационных систем и процессов. <b>У2(ПК7)</b> проводить процедуры тестирования информационных систем <b>ВЛАДЕТЬ В1(ПК7)</b> современными инструментальными вычислительными средствами.

## 3. Структура и содержание дисциплины "Алгоритмы и структуры данных"

Объем дисциплины (модуля) составляет

10 зачетных единиц, всего 360 часов, из которых

116 часов составляет контактная работа обучающегося с преподавателем:

64 часа занятия лекционного типа,

32 часа занятия семинарского типа,

16 часов лабораторные работы,

4 часа мероприятия промежуточной аттестации

244 часа составляет самостоятельная работа обучающегося (в т.ч. включая 36 часов подготовки к экзамену).

### Содержание дисциплины (модуля)

Наименование и краткое содержание разделов и тем дисциплины,  форма промежуточной аттестации по дисциплине	Всего (часы)	В том числе				Самостоятельная работа студента часы
		контактная работа (работа во взаимодействии с преподавателем), часы из них				
		Занятия лекционного типа	Занятия семинарского типа	Лабораторные работы	Всего контактных часов	СРС
Технологии программирования. Базовые концепции ООП.	24	4	2		6	18
Классы C++. Стандартные члены класса.	24	4	2		6	18
Перегрузка операций.	24	4	2		6	18
Наследование и полиморфизм	36	12	6		18	18
Шаблоны функций и классов	35	8	4		12	23
<b>В т.ч. текущий контроль</b>	2					
<b>Промежуточная аттестация: зачет</b>						
Понятие структуры данных. Схема и экземпляр СД. Понятие структуры хранения	35	6	2	2	10	25
Структура хранения множества	36	5	3	3	11	25
Структура хранения матриц специального вида	36	5	3	3	11	25
Динамические структуры данных (Стек, Очередь)	37	6	3	3	12	25
Линейные списки	36	5	3	3	11	25
Таблицы	33	5	2	2	9	24
<b>В т.ч. текущий контроль</b>	2					
<b>Итоговая аттестация – зачёт, экзамен</b>						

#### 4. Образовательные технологии

Используются активные и интерактивные образовательные технологии в форме лекций, практических занятий, лабораторных работ.

**Лекция-информация.** Ориентирована на изложение и объяснение студентам научной информации, подлежащей осмыслению и запоминанию.

**Лекция-беседа,** или «диалог с аудиторией», является наиболее распространенной и сравнительно простой формой активного вовлечения студентов в учебный процесс. Эта лекция предполагает непосредственный контакт преподавателя с аудиторией. Преимущество лекции-беседы состоит в том, что она позволяет привлекать внимание слушателей к наиболее важным вопросам темы, определять содержание и темп изложения учебного материала с учетом особенностей обучаемых.

К участию в лекции-беседе можно привлечь различными приемами, так, например, озадачивание слушателей вопросами в начале лекции и по ее ходу. Вопросы могут, быть информационного и проблемного характера, для выяснения мнений и уровня осведомленности по рассматриваемой теме, степени их готовности к восприятию

последующего материала. Вопросы адресуются всей аудитории. Слушатели отвечают с мест. Если преподаватель замечает, что кто-то из обучаемых не участвует в ходе беседы, то вопрос можно адресовать лично тому слушателю, или спросить его мнение по обсуждаемой проблеме. Для экономии времени вопросы рекомендуется формулировать так, чтобы на них можно было давать однозначные ответы. С учетом разногласий или единодушия в ответах преподаватель строит свои дальнейшие рассуждения, имея при этом возможность, наиболее доказательно изложить очередное понятие лекционного материала.

Вопросы могут быть как простыми для того, чтобы сосредоточить внимание слушателей на отдельных аспектах темы, так и проблемные. Обучаемый, продумывая ответ на заданный вопрос, получает возможность самостоятельно прийти к тем выводам и обобщениям, которые преподаватель должен был сообщить им в качестве новых знаний, либо понять важность обсуждаемой темы, что повышает интерес, и степень восприятия материала слушателями.

Во время проведения лекции-беседы задаваемые вопросы не должны оставаться без ответов, иначе они будут носить риторический характер, не обеспечивая достаточной активизации мышления обучаемых. Наиболее проблемные вопросы могут быть вынесены на самостоятельную работу студентов, проверку которой преподаватель осуществляет в рамках текущего контроля успеваемости и/или промежуточной аттестации.

Эффективность лекции-беседы в условиях группового обучения снижается из-за того, что не всегда удается каждого обучаемого вовлечь в двусторонний обмен мнениями.

**Лекция с заранее запланированными ошибками** - рассчитана на стимулирование студентов к постоянному контролю предлагаемой информации (поиск ошибки: содержательной, методологической, методической, орфографической). Предварительно студенты могут быть разбиты на малые группы. В конце лекции или на следующем занятии проводится диагностика групп и разбор сделанных ошибок.

Данный вид лекции проводится в завершение темы или раздела учебной дисциплины, когда у слушателей сформированы основные понятия и представления. Заключительный анализ ошибок развивает у слушателей теоретическое мышление.

**Лекция-консультация** – по типу «**вопросы—ответы—дискуссия**», является тройным сочетанием: изложение новой учебной информации лектором, постановка вопросов и организация дискуссии в поиске ответов на поставленные вопросы». Проводится в период работы обучающихся над проектной работой.

**Лекция-консультация** по типу «**вопросы—ответы**». Лектор отвечает в течение лекционного времени на вопросы студентов по всем разделу или всему курсу. Проводится перед защитой проектных работ и промежуточной аттестацией.

Для выполнения лабораторных работ предусматривается работа в терминал-классах.

## **5. Учебно-методическое обеспечение самостоятельной работы обучающихся**

### **5.1. Виды самостоятельной работы студентов**

- проработка теоретического материала лекционных занятий;
- выполнение домашних заданий по практическим занятиям;
- подготовка к выполнению лабораторных работ;

Выполнение лабораторных работ на следующие темы:

Лабораторная работа «Верхнетреугольные матрицы на шаблонах»

Лабораторная работа «Реализация класса Стек на списке»

Лабораторная работа «Реализация класса Очередь на динамическом массиве»

Лабораторная работа «Представление полиномов на основе списка»

Лабораторная работа «Таблица на сбалансированных деревьях»

Лабораторная работа «Реализация хеш-таблицы»

## 5.2. Вопросы для контроля

1. Ссылки. Операторы new и delete..
2. Функции как члены структуры..
3. Классы. Конструкторы.
4. Структуры и классы..
5. Ссылка на себя
6. Деструкторы. .
7. Область действия имен..
8. Заголовочные файлы.
9. Понятие стека.
10. Пространства имен.
11. Перегрузка операторов.
12. Программа моделирования комплексных чисел..
13. Конструктор копирования.
14. Вывод и Ввод в языке C++.
15. Перегрузка операторов ввода и вывода для пользовательских типов.
16. Файловый ввод и вывод.
17. Сравнение текстового и бинарного файлов.
18. Объекты как члены класса.
19. Конструкторы встроенных типов.
20. Наследование классов.
21. Управление доступом при наследовании.
22. Наследование и конструкторы
23. Производный класс личных данных.
24. Виртуальные функции.
25. Абстрактные классы
26. Локальные классы.
27. Совместимость типов при наследовании.
28. Множественное наследование.
29. Класс алгебраических векторов Vector.
30. Класс прямоугольных матриц.
31. Класс систем линейных уравнений.
32. Шаблоны функций и классов
33. Шаблон классов векторов и динамических массивов.
34. Обработка исключений.
35. Стандартная библиотека шаблонов..
36. Динамическое создание объектов класса. Виртуальные деструкторы.

## 5.3. Образовательные материалы для студентов

- Белоцерковская И., Галина Н., Катаева Л. Интернет-курс "Алгоритмизация . Введение в язык программирования C++" // Национальный Открытый Университет «ИНТУИТ». URL: <http://www.intuit.ru/studies/courses/16740/1301/info> (дата обращения 8.06.2017).
- Седжвик Р. "Алгоритмы на C++". Национальный Открытый Университет «ИНТУИТ». URL: <http://www.intuit.ru/studies/courses/12181/1174/info>
- Петров А. "Программирование на C++" . Национальный Открытый Университет «ИНТУИТ». URL: <http://www.intuit.ru/studies/courses/3492/734/info>

## 6. Фонд оценочных средств для промежуточной аттестации по дисциплине

### 6.1 Перечень компетенций выпускников образовательной программы с указанием результатов обучения

Индикаторы компетенции	Критерии оценивания (дескрипторы)	Шкала оценивания
<p><b>ЗНАТЬ</b>  <b>З1(ОПК3)</b> основные алгоритмические конструкции и программные решения  <b>З2(ОПК3)</b> основы построения математических, имитационных и информационных моделей</p> <p><b>УМЕТЬ</b>  <b>У1(ОПК3)</b> разрабатывать программные решения на базе математических, имитационных и информационных моделей  <b>У2(ОПК3)</b> создавать ресурсы для глобальной сети, БД  <b>У3(ОПК3)</b> создавать тесты для тестирования систем на соответствие требованиям и стандартам</p> <p><b>Владеть</b>  <b>В1(ОПК3)</b> навыками тестирования информационных систем</p> <p>Знать:  <b>З1(ПК7)</b> Информационное обеспечение ИС, Методы анализа прикладной области</p> <p>Уметь  <b>У1(ПК7)</b> профессионально разрабатывать и использовать программное обеспечение для поддержки информационных систем и процессов.  <b>У2(ПК7)</b> проводить процедуры тестирования информационных систем</p> <p><b>ВЛАДЕТЬ</b>  <b>В1(ПК7)</b> современными инструментальными вычислительными средствами.</p>	Отсутствие знаний материала, отсутствует способность решения стандартных задач, полное отсутствие навыков, предусмотренных компетенцией.	Плохой уровень формирования компетенции. «Плохо»
	Наличие грубых ошибок в основном материале, наличие грубых ошибок при решении стандартных задач, отсутствие навыков, предусмотренных данной компетенцией	Неудовлетворительный уровень формирования компетенции. «неудовлетворительно»
	<b>Знать</b> некоторые основные понятия и свойства изучаемых алгоритмов <b>Уметь</b> У1(ОПК3,ПК7) с погрешностями. <b>Уметь</b> У2(ОПК3,ПК7) с погрешностями. <b>Владеть</b> некоторыми основными навыками, демонстрируя их в стандартных ситуациях	Удовлетворительный уровень формирования компетенции. «Удовлетворительно»
	<b>Знать</b> большинство основных понятий и свойств изучаемых последовательных алгоритмов и их параллельных обобщений. <b>Уметь</b> У1(ОПК3,ПК7) с незначительными погрешностями. <b>Уметь</b> У2(ОПК3,ПК7) с незначительными погрешностями. <b>Владеть</b> основными навыками, демонстрируя их в стандартных ситуациях	Хороший уровень формирования компетенции. «Хорошо»
	<b>Знать</b> алгоритмы и структуры данных. <b>Уметь</b> У1(ОПК3,ПК7) с незначительными погрешностями. <b>Уметь</b> У2(ОПК3,ПК7) с незначительными погрешностями <b>Владеть</b> всеми основными навыками, демонстрируя их в стандартных ситуациях	Очень хороший уровень формирования компетенции «Очень хорошо»
	<b>Знать</b> основные методы и алгоритмы, предусмотренные компетенцией без ошибок и погрешностей. <b>Уметь</b> У1(ОПК3,ПК7) в полном объеме. <b>Уметь</b> У2(ОПК3,ПК7) в полном объеме. <b>Владеть</b> всеми навыками, демонстрируя их в стандартных ситуациях.	Отличный уровень формирования компетенции «Отлично»
	<b>Знать</b> основной и дополнительный материал без ошибок и погрешностей. <b>Уметь</b> У1(ОПК3,ПК7) в полном объеме. <b>Уметь</b> У2(ОПК3,ПК7) в полном объеме. <b>Владеть</b> свободно всеми навыками, демонстрируя их в стандартных и нестандартных	Превосходный уровень формирования компетенции «Превосходно»

	ситуациях.	
--	------------	--

### Карта компетенций для оценивания умений и навыков

Индикаторы компетенции	Критерии оценивания (дескрипторы)						
	«плохо»	«неудовлетворительно»	«удовлетворительно»	«хорошо»	«очень хорошо»	«отлично»	«превосходно»
Умения У1(ОПК3) У2(ОПК3) У3(ОПК3) У1(ПК7), У2(ПК7)	отсутствует способность решения стандартных задач	наличие грубых ошибок при решении стандартных задач	способность решения основных стандартных задач с негрубыми ошибками	способность решения всех стандартных задач с незначительными погрешностями	способность решения всех стандартных задач без ошибок и погрешностей	Способность решения стандартных и некоторых нестандартных задач	способность решения стандартных и широкого круга нестандартных задач
Навыки В1(ОПК3), В1(ПК7),	полное отсутствие навыков, предусмотренных компетенцией	отсутствие ряда важнейших навыков, предусмотренных данной компетенцией	наличие минимального необходимого множества навыков	наличие большинства основных навыков, продемонстрированное в стандартных ситуациях	наличие всех основных навыков, продемонстрированных в стандартных ситуациях	наличие всех навыков, продемонстрированное в стандартных ситуациях	Наличие всех навыков, продемонстрированное в стандартных и нестандартных ситуациях
Личностные качества ОПК3 ПК7	соответствующие личностные качества не сформированы	сформированность личностных качеств недостаточная для достижения основных целей обучения	сформированность личностных качеств минимальная для достижения основных целей обучения	личностные качества в целом сформированы	сформированные личностные качества достаточны для достижения целей обучения	Личностные качества сформированы на высоком уровне	Сформированность личностных качеств выше обязательных требований

### 6.2 Описание шкал оценивания

Для оценивания результатов учебной деятельности студентов при изучении дисциплины «Алгоритмы и структуры данных» используется балльная система оценки учебной работы

студентов. По результатам промежуточной аттестации проставляются оценки «Зачтено» (соответствует уровням оценки компетенций «удовлетворительно» и выше) и «Не зачтено» (соответствует уровням оценки компетенций «плохо» и «неудовлетворительно»). По результатам итоговой аттестации проставляются оценки «Превосходно», «Отлично», «Очень хорошо», «Хорошо», «Удовлетворительно» и «Неудовлетворительно» в соответствии со шкалой оценивания в таблице компетенций. По результатам итоговой аттестации в 4 семестре на экзамене выставляются оценки по семибалльной системе.

Индикаторы компетенции	ОЦЕНКИ СФОРМИРОВАННОСТИ КОМПЕТЕНЦИЙ						
	плохо	неудовлетворительно	удовлетворительно	хорошо	очень хорошо	отлично	превосходно
<b>Полнота знаний</b>	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.	Уровень знаний в объеме, превышающем программу подготовки.
<b>Наличие умений</b>	Отсутствие минимальных умений. Невозможность оценить наличие умений вследствие отказа обучающегося от ответа	При решении стандартных задач не продемонстрированы основные умения. Имели место грубые ошибки.	Продемонстрированы основные умения. Решены типовые задачи с негрубыми ошибками. Выполнены все задания, но не в полном объеме.	Продемонстрированы все основные умения. Решены все основные задачи с негрубыми ошибками. Выполнены все задания, в полном объеме, но некоторые с недочетами.	Продемонстрированы все основные умения. Решены все основные задачи. Выполнены все задания, в полном объеме, но некоторые с недочетами.	Продемонстрированы все основные умения, решены все основные задачи с отдельными несущественными недочетами, выполнены все задания в полном объеме.	Продемонстрированы все основные умения,. Решены все основные задачи. Выполнены все задания, в полном объеме без недочетов
<b>Наличие навыков (владение опытом)</b>	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	Продемонстрированы базовые навыки при решении стандартных задач с некоторыми недочетами	Продемонстрированы базовые навыки при решении стандартных задач без ошибок и недочетов.	Продемонстрированы навыки при решении нестандартных задач без ошибок и недочетов.	Продемонстрирован творческий подход к решению нестандартных задач
<b>Мотивация (лично учебное)</b>	Полное отсутствие учебной	Учебная активность и	Учебная активность и	Учебная активность и мотивация	Учебная активность и мотивация	Учебная активность и мотивация	Учебная активность и мотивация

<b>отношение)</b>	активности и мотивации	мотивация слабо выражены, готовность решать поставленные задачи качественно отсутствуют	мотивация низкие, слабо выражены, стремление решать задачи качественно	проявляются на среднем уровне, демонстрируется готовность выполнять поставленные задачи на среднем уровне качества	проявляются на уровне выше среднего, демонстрируется готовность выполнять большинство поставленных задач на высоком уровне качества	проявляются на высоком уровне, демонстрируется готовность выполнять все поставленные задачи на высоком уровне качества	проявляются на очень высоком уровне, демонстрируется готовность выполнять нестандартные дополнительные задачи на высоком уровне качества
<b>Характеристика сформированности и компетенции</b>	Компетенция в не сформирована. отсутствуют знания, умения, навыки, необходимые для решения практических (профессиональных) задач. Требуется повторное обучение	Компетенция в полной мере не сформирована. Имеющихся знаний, умений, навыков недостаточно для решения практических (профессиональных) задач. Требуется повторное обучение	Сформированность компетенции и соответствующим минимальным требованиям. Имеющихся знаний, умений, навыков в целом достаточно для решения практических (профессиональных) задач, но требуется дополнительная практика по большинству практических задач.	Сформированность компетенции в целом соответствует требованиям, но есть недочеты. Имеющихся знаний, умений, навыков и мотивации в целом достаточно для решения практических (профессиональных) задач, но требуется дополнительная практика по некоторым профессиональным задачам.	Сформированность компетенции в целом соответствует требованиям. Имеющихся знаний, умений, навыков и мотивации в целом достаточно для решения стандартных практических (профессиональных) задач.	Сформированность компетенции полностью соответствует требованиям. Имеющихся знаний, умений, навыков и мотивации в полной мере достаточно для решения сложных практических (профессиональных) задач.	Сформированность компетенции превышает стандартные требования. Имеющихся знаний, умений, навыков и мотивации в полной мере достаточно для применения творческого подхода к решению сложных практических (профессиональных) задач
<b>Уровень сформированности и компетенций</b>	Нулевой	Низкий	Ниже среднего	Средний	Выше среднего	Высокий	Очень высокий
<b>Баллы, %</b>	0-30	31-50	51-70	71-85	86-90	91-98	99-100

**Критерии оценок при проведении зачета**

<b>Зачтено</b>	Студент свободно ориентируется в понятиях и основных фактах курса, отвечает на контрольные вопросы, решает предложенные задания из перечня заданий.
<b>Не зачтено</b>	Студент не ориентируется в понятиях и фактах курса, не отвечает на контрольные вопросы, не решает задания из перечня заданий

### **6.3 Критерии и процедуры оценивания результатов обучения по дисциплине , характеризующих этапы формирования компетенций. Дисциплина формирует начальный уровень компетенции ПК7.**

Для оценивания результатов обучения в виде знаний используются следующие процедуры и технологии:

- Устные и письменные ответы на вопросы.
- Тестирование
- Индивидуальное собеседование

Для оценивания результатов обучения в виде умений и владений используются следующие процедуры и технологии:

- практические контрольные задания (далее – ПКЗ), включающих одну или несколько задач (вопросов) в виде краткой формулировки действий (комплекса действий), которые следует выполнить, или описание результата, который нужно получить.

По сложности ПКЗ разделяются на простые и комплексные задания. Простые ПКЗ предполагают решение в одно или два действия. К ним можно отнести: простые ситуационные задачи с коротким ответом или простым действием; несложные задания по выполнению конкретных действий. Простые задания применяются для оценки умений. Комплексные задания требуют многоходовых решений как в типичной, так и в нестандартной ситуациях. Это задания в открытой форме, требующие поэтапного решения и развернутого ответа. Комплексные практические задания применяются для оценки владений.

### **6.4 Типовые контрольные вопросы или иные материалы, необходимые для оценки результатов обучения, характеризующих этапы формирования компетенций и (или) для итогового контроля сформированности компетенции.**

#### **6.4.1. Вопросы к зачету для оценивания компетенций ОПК-3, ПК-7**

##### **3 семестр**

- Ссылки. Ссылка как аргумент функции. Возврат ссылки из функции.
- Операторы new и delete. Привести примеры использования.
- Функции как члены структуры. Программа моделирования временем в виде структуры.
- Встроенные функции.
- Классы. Инкапсуляция. Члены класса. Скрытие данных. Объекты класса.
- Конструкторы. Программа «Конструкторы в классе дат».
- Размер класса и размер объектов класса.
- Друзья класса. Привести пример.
- Копирование объектов класса.
- Структуры и классы. Правила доступа к членам класса.
- Статические члены класса. Привести пример.
- Ссылка на себя. Программа модификации дат.
- Деструкторы. Деструктор в классе дат.
- Программа моделирования многоугольников на плоскости.
- Область действия имен. Глобальные и локальные имена. Статические имена.
- Заголовочные файлы. Страж включения.
- Понятие стека. Реализация стека в виде массива.
- Программа анализа правильности расстановки скобок в тексте.
- Пространства имен.

- Стандартное пространство имен std.
- Перегрузка операторов. Программа моделирования обыкновенных дробей.
- Правила перегрузки операторов.
- Программа моделирования комплексных чисел.
- Проблемы при копировании объектов, содержащих указатели.
- Конструктор копирования.
- Вывод в языке C++.
- Ввод в языке C++.
- Перегрузка операторов ввода для пользовательских типов.
- Перегрузка операторов вывода для пользовательских типов.
- Файловый ввод и вывод.
- Сравнение текстового и бинарного файлов.
- Объекты как члены класса.
- Конструкторы встроенных типов.
- Наследование классов.

#### 4 семестр

- Управление доступом при наследовании.
- Наследование и конструкторы
- Производный класс личных данных.
- Виртуальные функции.
- Абстрактные классы
- Вызов виртуальных функций.
- Локальные классы.
- Совместимость типов при наследовании.
- Множественное наследование.
- Класс алгебраических векторов Vector.
- Класс прямоугольных матриц.
- Класс систем линейных уравнений.
- Шаблоны функций.
- Шаблоны классов.
- Шаблон классов векторов.
- Шаблон классов динамических массивов.
- Обработка исключений.
- Стандартная библиотека шаблонов. Привести пример использования шаблонов.
- Динамическое создание объектов класса. Виртуальные деструкторы.

#### 6.4.2. Вопросы для экзамена для оценивания компетенций ОПК-3, ПК-7

Вопрос 1.1. Технологии программирования: структурное, модульное, ООП. Общее описание.

Вопрос 1.2. Концепции ООП: абстрагирование, инкапсуляция. Поддержка в языке программирования C++. Примеры.

Вопрос 1.3. Концепции ООП: наследование. Поддержка в языке программирования C++. Примеры.

Вопрос 1.4. Концепции ООП: полиморфизм. Поддержка в языке программирования C++. Примеры.

Вопрос 1.5. Создание объектов. Конструкторы. Виды конструкторов, объявление. Реализация конструкторов по умолчанию и инициализации на примере класса Complex. Примеры использования.

Вопрос 1.6. Создание объектов. Конструкторы. Виды конструкторов, объявление. Реализация конструкторов копирования и преобразования типа на примере класса Complex. Примеры использования.

Вопрос 1.7. Удаление объектов. Деструктор. Пример реализации для класса String. Реализация метода Add для класса Complex.

Вопрос 1.8. Перегрузка функций в C++. Перегрузка унарных и бинарных операций. Перегрузка операций ++ с и с ++ для класса Complex. Примеры использования.

Вопрос 1.9. Перегрузка функций в C++. Перегрузка унарных и бинарных операций. Перегрузка операций + и += для класса Complex. Примеры использования.

Вопрос 1.10. Перегрузка функций в C++. Перегрузка унарных и бинарных операций. Перегрузка оператора присваивания на примере класса String. Отличия оператора присваивания от конструктора копирования. Примеры использования.

Вопрос 1.11. Разработка класса Vector. Объявление (конструкторы, деструктор, перегрузка операций +, -, +=, -=, ++, --, [], =). Реализация конструкторов, деструктора, оператора присваивания.

Вопрос 1.12. Разработка класса Vector. Объявление (конструкторы, деструктор, перегрузка операций +, -, +=, -=, ++, --, [], =). Реализация операций +, +=, ++, [].

Вопрос 1.13. Спецификаторы доступа в объявлении класса. Примеры. Друже-ственные функции на примере операции + для класса Complex.

Вопрос 1.14. Полиморфизм на примере иерархии классов геометрических фигур. Таблица виртуальных функций. Преимущества использования полиморфизма.

Вопрос 1.15. Полиморфизм на примере иерархии классов геометрических фигур. Абстрактные классы. Виртуальный деструктор.

Вопрос 1.16. Макросы и шаблоны функций. Отличия, плюсы и минусы. Примеры.

Вопрос 1.17. Шаблоны классов. Описание. Отличия реализации шаблонного класса от обычного. Пример: динамический массив.

Вопрос 2.1. Понятие структуры данных (СД). Схема и экземпляр СД. Примеры: вектор, матрица. Понятие структуры хранения. Примеры.

Вопрос 2.2. Тип данных: список, реализованный на статическом массиве

Вопрос 2.3. Тип данных: стек, реализованный на статическом массиве

Вопрос 2.4. Тип данных: шаблон стека, реализованный на статическом массиве

Вопрос 2.5. Тип данных: стек, реализованный на динамическом массиве

Вопрос 2.7. Тип данных: шаблон класса очередь с круговым заполнением

Вопрос 2.8. Тип данных: односвязный список

Вопрос 2.9. Тип данных: бинарное дерево поиска

Вопрос 2.10. Тип данных: AVL-дерево

Вопрос 2.11. Тип данных: красно-черное дерево

Вопрос 2.12. Тип данных: хэш-таблица

### 6.4.3. Образец теста для оценивания результатов формирования знаний, умений ОПК-3, ПК-7

1. Что из перечисленного является объектом?

Число	Цвет	Номер кредитной карты	Счет в банке
-------	------	-----------------------	--------------

Правильные ответы:

## 2. Отношение целого и его части, приводящие к соотв. иерархии объектов

Абстракция	Агрегация	Консолидация	Типизация
------------	-----------	--------------	-----------

Правильные ответы:

## 3. Для чего используются классы?

Для создания эффективных программ	Для разработки прототипов	Для создания объектов	Для описания шаблонов
-----------------------------------	---------------------------	-----------------------	-----------------------

Правильные ответы:

### Критерий оценивания результатов тестирования

Баллы, %	Оценка
99-100	Превосходно
91-98	Отлично
86-90	Очень хорошо
71-85	Хорошо
51-70	Удовлетворительно
31-50	Неудовлетворительно
0-30	Плохо

### Образец экзаменационного билета

Национальный исследовательский Нижегородский государственный университет  
им. Н.И. Лобачевского  
Институт ИИТММ  
Кафедра Программная инженерия  
Дисциплина Алгоритмы и структуры данных

#### ЭКЗАМЕНАЦИОННЫЙ БИЛЕТ № 1

1. Технологии программирования: структурное, модульное, ООП. Общее описание.
2. Понятие структуры данных (СД). Схема и экземпляр СД. Примеры: вектор, матрица. Понятие структуры хранения. Примеры.

Зав. кафедрой \_\_\_\_\_  
Экзаменатор \_\_\_\_\_

### 6.5. Методические материалы, определяющие процедуры оценивания

Положение «О проведении текущего контроля успеваемости и промежуточной аттестации обучающихся в ННГУ», утвержденное приказом ректора ННГУ от 13.02.2014 г. №55-ОД, Положение о фонде оценочных средств, утвержденное приказом ректора ННГУ от 10.06.2015 №247-ОД.

### 7. Учебно-методическое и информационное обеспечение дисциплины (модуля)

а) основная литература:

1. Белоцерковская И., Галина Н., Катаева Л. Интернет-курс "Алгоритмизация . Введение в язык программирования C++" // Национальный Открытый Университет «ИНТУИТ». URL: <http://www.intuit.ru/studies/courses/16740/1301/info> (дата обращения 8.06.2017).

2. Седжвик Р. "Алгоритмы на С++". Национальный Открытый Университет «ИНТУИТ». URL: <http://www.intuit.ru/studies/courses/12181/1174/info>

б) дополнительная литература

1. Петров А. "Программирование на С++" . Национальный Открытый Университет «ИНТУИТ». URL: <http://www.intuit.ru/studies/courses/3492/734/info>

- в) программное обеспечение и Интернет-ресурсы  
Microsoft Visual Studio 2010-2015

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

Имеются в наличии учебные аудитории для проведения занятий лекционного типа, занятий семинарского типа, промежуточной аттестации, а также помещения для самостоятельной работы, оснащенные компьютерной техникой с возможностью подключения к сети «Интернет». Учебная и научная литература, учебно-методические материалы, представленные в библиотечном фонде, в электронных библиотеках и на кафедре программной инженерии.

Используемое лицензионное программное обеспечение:

- Операционные системы семейства Microsoft Windows, лицензия по подписке Microsoft Imagine.
- Среда разработки семейства Microsoft Visual Studio, лицензия по подписке Microsoft Imagine.

Программа составлена в соответствии с требованиями ФГОС ВО с учетом рекомендаций и ОПОП ВО по направлению «01.03.02 Прикладная математика и информатика» (профиль «Системный анализ, исследование операций и управление»).

Автор (ы) доцент кафедры программной инженерии \_\_\_\_\_ Штанюк А.А.

Рецензент (ы) \_\_\_\_\_

Заведующий кафедрой программной инженерии проф. \_\_\_\_\_ Гергель В.П.

Программа одобрена на заседании методической комиссии Института информационных технологий, математики и механики ННГУ им. Н.И. Лобачевского  
от \_\_\_\_\_ года, протокол № \_\_\_\_\_.