



**ФЕДЕРАЛЬНОЕ АГЕНТСТВО ВОЗДУШНОГО ТРАНСПОРТА
(РОСАВИАЦИЯ)**

**ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
ГРАЖДАНСКОЙ АВИАЦИИ»**



УТВЕРЖДАЮ

Ректор

/ Ю.Ю. Михальчевский

« 17 »

июня

2021 года

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

Алгоритмы и структуры данных

Направление подготовки
25.03.03 Аэронавигация

Направленность программы (профиль)
**Техническая эксплуатация автоматизированных систем
управления воздушным движением**

Квалификация выпускника
бакалавр

Форма обучения
очная

Санкт-Петербург
2021

1 Цели освоения дисциплины

Целями освоения дисциплины «Алгоритмы и структуры данных» являются формирование у студентов комплекса знаний в отношении форм организации данных в программах и методов их обработки применительно к различным классам задач, решаемых в рамках теоретической информатики, а также приобретение ими навыков и умений анализа эффективности существующих классических и вновь разрабатываемых алгоритмов.

Для достижения поставленных целей в рамках дисциплины решаются следующие задачи:

- ознакомить обучающегося с классическими методами решения в области теоретической информатики;
- обучить способам ведения доказательств и установления корректности математических утверждений;
- отработать навыки определения вычислительной сложности (производительности, эффективности) алгоритмов и интерфейсов структур данных.

Дисциплина обеспечивает подготовку выпускника к решению задач профессиональной деятельности эксплуатационно-технологического типа.

2 Место дисциплины в структуре ОПОП ВО

Дисциплина «Алгоритмы и структуры данных» представляет собой дисциплину, относящуюся к обязательной части Блока 1 «Дисциплины (модули)» ОПОП ВО по направлению подготовки 25.03.03 «Аэронавигация» (бакалавриат), профиль «Техническая эксплуатация автоматизированных систем управления воздушным движением».

Дисциплина «Алгоритмы и структуры данных» базируется на результатах обучения, полученных при изучении дисциплин: «Информатика», «Алгоритмические языки и программирование», «Базы данных», «Операционные системы и сети», «Объектно-ориентированное программирование».

Дисциплина «Алгоритмы и структуры данных» является обеспечивающей для дисциплин: «Средства автоматизации управления и планирования воздушного движения», «Математическое обеспечение систем управления воздушным движением», «Визуальное программирование».

Дисциплина изучается в 5 семестре.

3 Компетенции обучающегося, формируемые в результате освоения дисциплины

Процесс освоения дисциплины «Алгоритмы и структуры данных» направлен на формирование следующих компетенций: ОПК-1; ПК-1; ПК-4.

Код компетенции/ индикатора	Результат обучения: наименование компетенции, индикатора компетенции
-----------------------------	--

Код компетенции/ индикатора	Результат обучения: наименование компетенции, индикатора компетенции
ОПК-1	Способен работать с готовыми программными продуктами и стандартными программными средствами при решении профессиональных задач
ИД ¹ _{ОПК-1}	ИД ¹ _{ОПК1} Ориентируется в пакетах прикладных программ, работает со стандартными программными средствами
ИД ² _{ОПК-1}	ИД ² _{ОПК1} Выбирает и использует стандартные программные средства для решения поставленных задач, в том числе в сфере профессиональной деятельности
ПК-1	Способен осуществлять эксплуатацию программного обеспечения автоматизированных систем управления воздушным движением
ИД ¹ _{ПК-1}	ИД ¹ _{ПК1} Знает состав и основные принципы функционирования программного обеспечения АС УВД и использует данную информацию при решении профессиональных задач
ИД ² _{ПК-1}	ИД ² _{ПК1} Применяет на практике все имеющиеся знания, умения и навыки при решении профессиональных задач, связанных с эксплуатацией программного обеспечения АС УВД.
ИД ³ _{ПК-1}	ИД ³ _{ПК1} Ориентируется в условиях изменения правовой базы и эксплуатационных требований, предъявляемых к программному обеспечению автоматизированных систем управления воздушным движением
ПК-4	Способен разрабатывать алгоритмы и программы для решения профессиональных задач
ИД ¹ _{ПК-4}	ИД ¹ _{ПК4} Идентифицирует входную и выходную информацию, а также определяет последовательность действий, необходимых для решения практической задачи
ИД ² _{ПК-4}	ИД ² _{ПК4} Использует инструментальные средства и методики разработки программного обеспечения
ИД ³ _{ПК-4}	ИД ³ _{ПК4} Принимает участие в поддержке всех этапов жизненного цикла программного обеспечения автоматизированных систем управления воздушным движением

Планируемые результаты изучения дисциплины:

Знать:

- основные структуры данных, используемые для представления типовых информационных объектов
- основные алгоритмы и их характеристики сложности для типовых задач, ставших классическими в теоретической информатике
- методы оценки вычислительной сложности
- методики поиска информации по заданной тематике

Уметь:

- разрабатывать алгоритмы, выбирая подходящие структуры данных для представления информационных объектов
- реализовывать алгоритмы и структуры данных как на абстрактном уровне (посредством псевдокода), так и на языках программирования
- экспериментально исследовать эффективность алгоритма и программы

Владеть:

- методами оценки вычислительной сложности алгоритмов
- методами определения корректности разрабатываемого алгоритма
- навыками составления эффективных алгоритмов и структур данных
- навыками использования типовых алгоритмов для реальных прикладных задач

4 Объем дисциплины и виды учебной работы

Общая трудоемкость дисциплины составляет 3 зачетные единицы, 108 академических часов.

Наименование	Всего часов	Семестр
		5
Общая трудоемкость дисциплины	108	108
Контактная работа:	42,5	42,5
лекции	14	14
практические занятия	28	28
семинары	–	–
лабораторные работы	–	–
курсовой проект (работа)	–	–
Самостоятельная работа студента	57	57
Промежуточная аттестация:	9	9
контактная работа	0,5	0,5
самостоятельная работа по подготовке к зачету с оценкой	8,5	8,5

5 Содержание дисциплины

5.1 Соотнесения тем (разделов) дисциплины и формируемых компетенций

Темы дисциплины	Количество часов	Компетенции			Образовательные технологии	Оценочные средства
		ОПК-1	ПК-1	ПК-4		
Тема 1. Элементарные структуры данных	14	+	+	+	ВК, Л,ПЗ, СРС	ВК, УО, ПР
Тема 2. Элементы теории графов и алгоритмы обходов	14	+	+	+	Л,ПЗ, СРС	УО, ПР
Тема 3. Проблема нахождения наикратчайших путей	14	+	+	+	Л,ПЗ, СРС	УО, ПР
Тема 4. Вычислительная геометрия	14	+	+	+	Л,ПЗ, СРС	УО, ПР
Тема 5. Двоичные деревья поиска. Хеш таблицы	14	+	+	+	Л,ПЗ, СРС	УО, ПР
Тема 6. Минимальные покрывающие деревья	14	+	+	+	Л,ПЗ, СРС	УО, ПР
Тема 7. Алгоритмы поиска подстроки	15	+	+	+	Л,ПЗ, СРС	УО, ПР
Промежуточная аттестация	9					
Всего по дисциплине	108					

Сокращения: Л – лекция, ПЗ – практическое занятие, ВК – входной контроль, СРС – самостоятельная работа студента, УО – устный опрос, ПР – разбор практических заданий

5.2 Темы (разделы) дисциплины и виды занятий

Наименование темы (раздела) дисциплины	Л	ПЗ	С	ЛР	СРС	КР	Всего часов
Тема 1. Элементарные структуры данных	2	4			8		14
Тема 2. Элементы теории графов и алгоритмы обходов	2	4			8		14
Тема 3. Проблема нахождения наикратчайших путей	2	4			8		14
Тема 4. Вычислительная геометрия	2	4			8		14

Наименование темы (раздела) дисциплины	Л	ПЗ	С	ЛР	СРС	КР	Всего часов
Тема 5. Двоичные деревья поиска. Хеш таблицы	2	4			8		14
Тема 6. Минимальные покрывающие деревья	2	4			8		14
Тема 7. Алгоритмы поиска подстроки	2	4			9		15
Итого за семестр	14	28	–	–	57	–	99
Промежуточная аттестация							9
Итого по дисциплине							108

Сокращения: Л – лекции, ПЗ – практические занятия, С – семинары, ЛР – лабораторные работы, СРС – самостоятельная работа студента, КР – курсовая работа.

5.3 Содержание дисциплины

Тема 1. Введение.

Определение алгоритма, вычислительной проблемы, размера задачи. Формальное описание алгоритма. Пузырьковый алгоритм сортировки (Insertion-Sort). Время выполнения алгоритма. Худшее, лучшее и среднее время выполнения алгоритма. Асимптотический анализ. Определения для O , Θ , Ω – нотаций. Методы определения верхней и нижней границы скорости роста монотонной функции. Элементы теории множеств. Рекуррентные выражения. Стратегия разделяй и властвуй. Алгоритм сортировки слиянием (Merge-Sort). Методы вычисления рекурсивных выражений: итерационны, подстановки, мастер формула. Примеры использования математической индукции. Алгоритм быстрой сортировки Quick-Sort. Нижняя граница для решения проблемы сортировки. Сортировка за линейное время – сортировка подсчетом (Count-Sort).

Тема 2. Элементарные структуры данных.

Статический массив. Динамический массив. Стек. Очередь. Связанный список (двойной, одиночный). Организация машинной памяти элементарных структур данных. Интерфейсы и их вычислительная сложность. Области применения.

Тема 3. Элементы теории графов и алгоритмы обходов.

Определение графа. Направленный и ненаправленный граф. Степень вершины. Путь и цикл. Методы представления графов в машинной памяти: последовательность граней, массив смежных вершин, список смежных вершин, матрица смежностей. Проблема обхода графа. Обобщенный алгоритм обхода. Обход графа поиском в ширину – алгоритм BFS. Обход графа поиском в глубину – алгоритм DFS. Классификация граней в графе. Топологическая сортировка. Деревья и их свойства. Двоичная куча. Пирамидальная сортировка.

Тема 4. Проблема нахождения наикратчайших путей.

Определение проблемы. Стоимость путей в графах. Стратегии решения проблемы SSSP. Релаксация граней. SSSP в направленном ациклическом графе. SSSP в графах с неотрицательной стоимостью граней – алгоритм Дикстра. Приоритетная очередь. Практическое использование алгоритма Дикстра. Дикстра с различными реализациями приоритетных очередей: наивная реализация, очередь корзин, радикс. SSSP для произвольных стоимостей граней – алгоритм Бельман-Форда.

Тема 5. Вычислительная геометрия.

Область применения. Функция ориентации. Проблема выпуклой оболочки. Алгоритм Грехема. Алгоритм разделяй и властвуй. Алгоритм последовательного построения. Проблема нахождения наименьшей окружности обрамления. Нахождение пересечений отрезков на плоскости – алгоритм скользящей линии (Sweep Line). Пересечение плоскостных разбиений. Триангуляция.

Тема 6. Двоичные деревья поиска.

Проблема двоичного поиска. Определение отсортированной последовательности (ОП) и операции над ними. Примеры использования ОП в практических приложениях. Реализация двоичного дерева. (a,b)-деревья. Операции locate(), insert(), remove(). Амортизационный анализ производительности структуры данных. Метод агрегаций. Банковский метод. Метод потенциалов.

Тема 7. Хеш таблицы.

Прямая адресация. Хеш-функции. Универсальное хеширование. Эвристические методики выбора хеш функции. Открытая адресация.

Тема 8. Минимальные покрывающие деревья.

Проблема построения минимального остова. Практическая область применения. Алгоритм Крускала. Алгоритм Ярника-Прима. Структура данных Union-Find.

Тема 9. Алгоритмы поиска подстроки.

Обозначения и терминология. Простейший алгоритм. Алгоритм Рабина-Карпа. Поиск подстроки с помощью конечных автоматов. Алгоритм Кнута-Морриса-Пратта. Префикс-функция. Алгоритм Бойера-Мура.

5.4 Практические занятия

Номер темы дисциплины	Тематика практических занятий	Трудоемкость (часы)
1	Методы определения вычислительной сложности алгоритмов. Асимптотическая форма записи для временных и пространственных затрат. Способы раскрытия рекуррентных выраже-	4

Номер темы дисциплины	Тематика практических занятий	Трудо-емкость (часы)
	ний. Метод математической индукции. Алгоритмы сортировки - Insertion-Sort, Quick-Sort, Count-Sort. Проблема нижней границы сортировки.. Способы организации графа в машинной памяти. Статический и динамический массив в языках программирования. Способы реализации связанного списка. Организация памяти, интерфейсы. Структуры данных – стек и очередь.	
2	Работа алгоритма обхода графа для заданного графа - поиск в ширину (BFS), входные и выходные параметры, время выполнения. Работа алгоритма обхода графа для заданного графа - поиск в глубину (DFS), входные и выходные параметры, время выполнения. Алгоритм топологической сортировки. Структура данных - двоичная куча и алгоритм пирамидальная сортировка.	4
3	Работа алгоритма поиска кратчайших путей для ациклического направленного графа. Работа алгоритма Дикстра на заданном примере графа. Примеры работы (заполнения) приоритетной очереди: наивная реализация, двоичная куча, очередь корзин, Radix-куча. Работа алгоритмов поиска кратчайших путей на пример сетевых устройств. Практическая организация пары клиент-сервер на примере протокола ТСР.	4
4	Алгоритм Грэхема (поиск выпуклой оболочки) и функция ориентации. Разбор работы алгоритма скользящей линии на примерах. Делони-триангуляция на примере построения функции высоты.	4
5	Реализация двоичного дерева поиска. Реализация структуры данных и интерфейсных методов (a,b)-деревьев. Методы амортизационного анализа: метод агрегации, банковский метод, метод подстановки. Применение деревьев поиска в практических приложениях: алгоритмы вычислительной геометрии, базы данных, планировщик задач операционной системы.	4
6	Реализация таблиц с прямой адресацией. Реализация хеш таблицы с простейшей функцией хеширования. Методики выбора хеш функции. Хеш таблицы с открытой адресацией. Определение вероятности коллизий. Алгоритм Крускала – работа алгоритма на примерных графах. Алгоритм Ярника-Прима – работа алгоритма на примерных графах. Реализация структуры данных Union-Find.	4
7	Алгоритмы поиска подстроки – метод Кнута-Мориса-Прата. Расчеты функции сдвигов. Алгоритм Бойера-Мура. Суффиксные деревья.	4
Итого		28

5.5 Лабораторный практикум

Лабораторный практикум учебным планом не предусмотрен.

5.6 Самостоятельная работа

Номер темы дисциплины	Виды самостоятельной работы	Трудо-емкость (часы)
1	Изучение теоретического материала раздела. Подготовка отчета для защиты практической работы. [1, 3, 4, 7, 8]	8
2	Изучение теоретического материала раздела. Подготовка отчета для защиты практической работы. [1,2]	8
3	Изучение теоретического материала раздела. Подготовка отчета для защиты практической работы. [3]	8
4	Изучение теоретического материала раздела. Подготовка отчета для защиты практической работы. [7,8]	8
5	Изучение теоретического материала раздела. Подготовка отчета для защиты практической работы. [9,10]	8
6	Изучение теоретического материала раздела. Подготовка отчета для защиты практической работы. [1, 7, 8, 10,11]	8
7	Изучение теоретического материала раздела. Подготовка отчета для защиты практической работы. [1,5,6,7]	9
Итого по дисциплине		57

5.7 Курсовые работы

Курсовые работы учебным планом не предусмотрены.

6 Учебно-методическое и информационное обеспечение дисциплины

а) основная литература:

1. **Конова Е.А., Поллак Г.А. Алгоритмы и программы. Язык С++** [Текст]: учеб. для вузов / Конова Е.А., Поллак Г.А. – СПб.:Лань, 2016. – 384 с. – ISBN 978-5-8114-2020-9.

2. **Тюкачев Н.А., Хлебостроев В.Г. С#. Алгоритмы и структуры данных** [Текст]: учеб. для вузов / Тюкачев Н.А., Хлебостроев В.Г. – СПб.:Лань, 2017. – 232 с. – ISBN: 978-5-8114-2566-2.

3. **Асанов М.О., Баранский В.А., Расин В.В. Дискретная математика: графы, матроиды, алгоритмы** [Текст]: учеб. для вузов / Асанов М.О., Баранский В.А., Расин В.В. – СПб.:Лань, 2010. – 368 с. - ISBN 978-5-8114-1068-2.

б) дополнительная литература:

4. **Абрамов, С. А. Лекции о сложности алгоритмов** [Текст]: учебн.-метод. пособие / С. А. Абрамов.– М.: МЦНМО, 2009. – 256 с.

5. **Вирт, Н. Алгоритмы и структуры данных** [Текст]: монография / Н. Вирт.– СПб.: Невский Диалект, 2008. –352 с. – ISBN 978-5-7940-0065-8.

6. **Кнут, Д. Э. Искусство программирования** [Текст]: монография / Д. Э. Курт.– М.: Вильямс, 2012. – 824 с. – ISBN 978-5-8459-0082-1.

7. **Кормен, Т. Х. Алгоритмы: построение и анализ** [Текст]: учеб. для вузов / Т.Х. Кормен, Ч.И. Лейзерсон, Р.Л. Ривест, К. Штайн. – 3-е издание, – М.: «Вильямс», 2012. – 1328 с. – ISBN 978-5-8459-1794-2.

8. **Мяготин А.В. Алгоритмы, структуры данных и численные методы** [Текст]: учеб. пособие / Мяготин А.В. – СПбГУ ГА, С.-Петербург, 2015. - 117с.

9. **Berg, M. Computational Geometry: Algorithms and Applications** [Текст]: монография / M. Berg, M. Kreveld, M. Overmars.– Berlin: Springer-Verlag, 2000. – 367 с. – ISBN 3-540-65620-0.

10. **Mehlhorn, K. LEDA: A Platform for Combinatorial and Geometric Computing** [Текст]: учеб. для вузов / K. Mehlhorn, S. Naeher.– Cambridge University Press, 1999. – 1034 с. – ISBN 978-0521563291.

11. **Mehlhorn, K. Algorithms and Data Structures: The Basic Toolbox** [Текст]: учеб. для вузов / K. Mehlhorn, P. Sanders.– Springer, 2008. – 300 с. – ISBN 978-3642096822.

в) перечень ресурсов информационно-телекоммуникационной сети «Интернет»:

12. **ABYY Lingvo** [Электронный ресурс]. – Режим доступа: <http://www.lingvo.ru>. Дата обращения: 15.05.2021.

13. **Wikipedia** [Электронный ресурс]. – Режим доступа: <http://www.wikipedia.org>. Дата обращения: 15.05.2021.

14. **Алгоритмы и структуры данных - видеоуроки** [Электронный ресурс]. – Режим доступа: <https://stepik.org/course/63/>. Дата обращения: 15.05.2021.

7 Материально-техническое обеспечение дисциплины

Компьютерные классы (ауд. 800-805) с доступом в Интернет, переносной проектор ACER X1261P.

Программное обеспечение: ОС Oracle Linux (GPL), OpenOffice / LibreOffice; Microsoft Windows 7 Professional, Microsoft Windows Office Professional, Oracle VirtualBox (GPL v2).

Информационно-справочные и материальные ресурсы библиотеки СПбГУ ГА.

8 Образовательные и информационные технологии

Лекция составляет основу теоретического обучения в рамках дисциплины и направлена на систематизированное изложение накопленных и актуальных научных знаний.

Практическое занятие: предназначено для отработки навыков использования приобретенных на лекционных занятиях теоретических знаний для решения прикладных и практических задач. На практических занятиях студент по-

лучает очередной блок заданий для самостоятельной работы и имеет возможность отчитаться по ранее выполненным заданиям.

Самостоятельная работа: является составной частью учебной работы. Ее основной целью является формирования навыка самостоятельного приобретения новых знаний по вопросам теоретического курса, закрепление и углубление полученных знаний, работа с периодическими изданиями и научно-популярной литературой, в том числе находящейся в информационных сетях, отработка навыков работы со специализированными программными пакетами. Самостоятельная работа подразумевает выполнение учебных заданий, полученных на практических занятиях.

Решения заданий, выносимых на самостоятельную работу, выполняются в форме отчета. Контроль за выполнением студентов заданий, выносимых на самостоятельную работу, осуществляет преподаватель в процессе их проверки (защиты) в период, отведенный на практические занятия.

IT-методы: является еще одним вариантом образовательной технологии. В рамках представленной дисциплины, под термином скрывается проведение учебного занятия, в рамках которого преподаватель формулирует задачу, сопряженную с теоретическим материалом, разбивает ее на этапы решения, которые последовательно решаются им с использованием компьютерной станции, оборудованной демонстрационным проектором. Студенты, находящиеся на своих рабочих местах, дублируют за преподавателем ходы решения - этап за этапом, вырабатывая в себе компонент «владения» заявленной компетенции, задают сопутствующие вопросы, тем самым закрепляя компоненты «знания» и «умения» формируемой компетенции. Данный вид занятия в полной мере обеспечивает продуктивный и творческий уровень деятельности студента при выполнении практически-значимых заданий.

Консультация: одна из форм руководства самостоятельной работой студентов и оказания им помощи в освоении учебного материала. Консультации может проводиться не реже раза в неделю в часы, свободные от учебных занятий, и носят индивидуальный характер. На консультациях разбираются вопросы, на которых базируется изучаемая дисциплина, и которые требуют более глубокого уровня освоения и понимания.

Учебно-методическое обеспечение самостоятельной работы студентов:

1. Конспект лекций.
2. Соответствующая учебная литература.
3. Офисные программные пакеты.

9 Фонд оценочных средств для проведения текущего контроля успеваемости и промежуточной аттестации по итогам освоения дисциплины

Уровень и качество знаний обучающихся оцениваются по результатам текущего контроля успеваемости и промежуточной аттестации по итогам освоения дисциплины.

Устный опрос проводится на практических занятиях с целью контроля усвоения теоретического материала, излагаемого на лекциях.

Тест проводится по темам в соответствии с данной программой и предназначен для проверки обучающихся на предмет освоения материала лекций.

Дискуссия, являясь одной из наиболее эффективных технологий группового взаимодействия, усиливает развивающие и воспитательные эффекты обучения, создает условия для открытого выражения участниками своих мыслей, позиций, обладает возможностью воздействия на установки ее участников. Принципами организации дискуссии являются содействие возникновению альтернативных мнений, путей решения проблемы, конструктивность критики, обеспечение психологической защищенности участников.

Практические задания выдаются студентам на практических занятиях и предназначены для закрепления теоретических знаний, а также для отработки умений и навыков. Как правило, они подразумевают проработку теоретического материала предыдущих лекций и последующее выполнение определенной последовательности действий на компьютере. При проверке преподавателем правильности выполнения задания студент также должен показать знание соответствующего теоретического материала.

Защита лабораторных работ подразумевает устный опрос студента по основным теоретическим сведениям, необходимым для выполнения работы, методике ее выполнения, полученным при этом результатам и их интерпретации.

Промежуточная аттестация по итогам освоения дисциплины проводится в виде зачета с оценкой в 5 семестре. К моменту сдачи зачета должны быть успешно пройдены предыдущие формы контроля.

Зачет с оценкой позволяют оценить уровень освоения компетенций за весь период изучения дисциплины. Билет включает два теоретических вопроса и задачу.

9.1 Балльно-рейтинговая оценка текущего контроля успеваемости и знаний студентов по дисциплине

Не применяется.

9.2 Методические рекомендации по проведению процедуры оценивания знаний, умений и навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций

Решение практических заданий оценивается:

«зачтено»: обучающийся самостоятельно правильно решает задачу, дает обоснованную оценку по итогу решения;

«не зачтено»: обучающийся отказывается от выполнения задачи или не способен ее решить самостоятельно, а также с помощью преподавателя.

Устный опрос:

«зачтено»: зачитывается в том случае, если получены достаточно полные и аргументированные ответы на вопросы преподавателя;

«не зачтено»: не зачитывается в том случае, если обучающийся не смог ответить на вопросы или ответил правильно менее чем на 61% вопросов.

Тест оценивается на «отлично», если количество правильных ответов 90% и более; «хорошо» – от 76% до 89%; «удовлетворительно» – от 61% до 75%; «неудовлетворительно» – менее 61%.

9.3 Темы курсовых работ (проектов) по дисциплине

В учебном плане курсовых работ не предусмотрено.

9.4 Контрольные вопросы для проведения входного контроля остаточных знаний по обеспечивающим дисциплинам

1. Приведите определения терминов: *алгоритм, рекурсия, множество, натуральное число, граф, дерево.*

2. Докажите методом математической индукции

$$S(n) = \sum_{i=0}^n (a_0 + i\Delta) = \frac{2a_0 + N\Delta}{2} N$$

3. Докажите методом математической индукции

$$\sum_{i=0}^{K-1} 2^i = 2^K - 1$$

4. Напишите на любом известном языке программирования текст программ, решающей уравнение вида $kx + b = 0$.

5. Приведите формулу суммы N членов геометрической последовательности.

6. Приведите примеры алгоритма ветвления и цикла.

9.5 Описание показателей и критериев оценивания компетенций на различных этапах их формирования, описание шкал оценивания

Компетенции	Показатели оценивания (индикаторы достижения) компетенций	Критерии оценивания
I этап		
ОПК-1	ИД ¹ _{ОПК-1}	Знает: - формальные определения для асимптотических O, Θ, Ω –нотаций; - классы вычислительных проблем и существующие алгоритмы их решений;

Компетенции	Показатели оценивания (индикаторы достижения) компетенций	Критерии оценивания
ПК-4	ИД ² _{ОПК-1} ИД ¹ _{ПК-4}	<p>Умеет:</p> <ul style="list-style-type: none"> - выполнять обзор существующих методов решений - обращаться с инструментами по определению вычислительной сложности и используемых объемов машинной памяти для заданного алгоритма (структуры данных); - использовать элементарные структуры данных (статический и динамический массив, стек, очередь) - составлять решение вычислительной проблемы с использованием псевдокода; <p>Владеет:</p> <ul style="list-style-type: none"> - использует асимптотическую нотацию для определения вычислительной сложности алгоритма
ПК-1	ИД ¹ _{ПК-1}	<p>Знает:</p> <ul style="list-style-type: none"> - области применения и базовые характеристики элементарных структур данных – статический и динамический массив, стек, очередь, связанный список;
II этап		
ОПК-1	ИД ¹ _{ОПК-1} ИД ² _{ОПК-1}	<p>Знает:</p> <ul style="list-style-type: none"> - принципы работы и характеристики сложных структур данных и способы их представления в машинной памяти – структуры данных для представления графа и дерева, хеши, сортирующие последовательности, контейнеры компьютерной графики - характеристики производительности заданного алгоритма - формальное доказательство (обоснование) эффективности алгоритма (структуры данных) с использованием асимптотических характеристик <p>Умеет:</p> <ul style="list-style-type: none"> - выбирать наиболее перспективные способы решения - разбивать работу алгоритма на отдельные этапы с возможностью оценки производительности отдельных блоков; - соотносить реальные измеренные временные (пространственные) оценки теоретическим асимптотическим характеристикам - использовать элементарные и сложные структуры данных - составлять решение вычислительной проблемы с использованием заданного языка программирования;
ПК-4	ИД ¹ _{ПК-4}	<p>Владеет:</p> <ul style="list-style-type: none"> - Составлением алгоритмическое решение задачи, используя эффективные (высоко-производительные) алго-

Компетенции	Показатели оценивания (индикаторы достижения) компетенций	Критерии оценивания
		ритмы и методы - Определением асимптотической оценки полученного решения - методами проведения формального доказательства заявленной вычислительной сложности алгоритма;
ПК-1	ИД ¹ _{ПК-1}	Владеть: математическими методами (в том числе индукцию) для доказательства корректности выполнения разрабатываемого алгоритма.

Шкала оценивания при проведении промежуточной аттестации

«Отлично» выставляется обучающемуся, показавшему всесторонние, систематизированные, глубокие знания по рассматриваемой компетенции и умение уверенно применять их на практике при решении задач, свободное и правильное обоснование принятых решений. Отвечая на вопрос, может быстро и безошибочно проиллюстрировать ответ собственными примерами. Обучающийся самостоятельно правильно решает задачу, дает обоснованную оценку итогам решения.

«Хорошо» выставляется обучающемуся, если он твердо знает материал, грамотно и по существу излагает его, умеет применять полученные знания на практике, но допускает в ответе или в решении задачи некоторые неточности, хорошо владеет всем содержанием, видит взаимосвязи, но не всегда делает это самостоятельно без помощи преподавателя. Обучающийся решает задачу верно, но при помощи преподавателя.

«Удовлетворительно» выставляется обучающемуся, показавшему фрагментарный, разрозненный характер знаний, недостаточно правильные формулировки базовых понятий, нарушения логической последовательности в изложении программного материала, но при этом он владеет основными разделами учебной программы в рамках заданной компетенции, необходимыми для дальнейшего обучения и может применять полученные знания по образцу в стандартной ситуации. Отвечает только на конкретный вопрос, соединяет знания из разных разделов курса только при наводящих вопросах преподавателя. Ситуационная задача решена не полностью, или содержатся незначительные ошибки в расчетах.

«Неудовлетворительно» выставляется обучающемуся, который не знает большей части основного содержания учебной программы дисциплины в рамках компетенций, допускает грубые ошибки в формулировках основных понятий дисциплины и не умеет использовать полученные знания при решении типовых практических задач. Не раскрыты глубина и полнота при ответах. Задача не решена даже при помощи преподавателя.

9.6 Типовые контрольные задания для проведения текущего контроля и промежуточной аттестации по итогам обучения по дисциплине

9.6.1 Примерные контрольные задания для проведения текущего контроля успеваемости

- Докажите корректность следующих утверждений
 - Если $f_1(n) \in O(g_1(n))$ и $f_2(n) \in O(g_2(n))$, то $f_1(n) + f_2(n) \in O(g_1(n) + g_2(n))$
 - Если $f_1(n) \in O(g_1(n))$ и $f_2(n) \in O(g_2(n))$, то $f_1(n) \cdot f_2(n) \in O(g_1(n) \cdot g_2(n))$
- Необходимо расположить приведенные функции в соответствии с их порядком роста, т.е. необходимо расположить функции $g_1(n), g_2(n), g_3(n)$ так, чтобы удовлетворялось условие $g_1(n) = O(g_2(n)), g_2(n) = O(g_3(n))$. В верхней строке указан номер варианта.

№	1	2	3	4	5	6
	$\lg n$	$(\sqrt{2})^{\lg n}$	n^2	$n \cdot 2^n$	$5 \cdot 2^n$	$\lg(n^n)$
	2^n	n^4	$\lg^2 n$	$\lg(n!)$	$n^{1/\lg n}$	$n^{2+\sin(n)}$
	n	$\lg(\lg n)$	$n!$	n^3	$\ln n$	$\ln \ln n$
	$n\sqrt{n}$	2	$n/2$	$n \log n$	6	$2^{\lg n}$

В отчете обязательно следует привести математический вывод полученного результата.

Примечание 1: Несколько функций могут иметь одинаковый порядок роста, т.е. функции $f(n)$ и $g(n)$ эквивалентны если $f = \Theta(g)$.

Примечание 2: Вы можете использовать различные формулы приближения и упрощения, например те, что приведены в книге Cormen T. и др. «Introduction to Algorithms» стр. 32-37.

- На лекции рассматривались худший и лучший случаи (best and worst cases) времени выполнения $T(n)$ алгоритма сортировки Insertion-Sort. Выведите формулу $T(n)$ для среднего случая (average case), принимая во внимание что для элемента массива A на позиции i в среднем производится по-

рядка $\lfloor i/2 \rfloor$ сдвигов во внутреннем цикле. Запишите полученное время используя Θ -нотацию.

4. Для следующего рекуррентного выражения

$$T(n) = \begin{cases} a & n = 1 \\ T(\lfloor n/4 \rfloor) + d \lfloor n/2 \rfloor & n > 1 \end{cases}$$

найдите форму $T(n)$ в явном виде при помощи итерационного метода,

5. При помощи метода математической индукции докажите, что сумма геометрической прогрессии a_1, a_2, \dots, a_n , где $a_i = a_1 \cdot q^{i-1}$ определяется как

$$S_n = \sum_{i=1}^n a_i = \frac{a_1(1-q^n)}{1-q}$$

6. Запишите псевдокод рекурсивной процедуры `sum_geom_series(n, a1, q)`, которая для заданных параметров находит сумму геометрической последовательности. Проведите анализ времени выполнения данной процедуры и запишите $T(n)$, используя рекуррентную и явную зависимость.

7. Приведите псевдокод процедуры `Merge(A, p, q, r)`. Покажите линейность времени выполнения Вашего алгоритма.

8. Покажите методом индукции, что число $5^{2n-1} + 1$ делится на 6 без остатка для $\forall n > 0$.

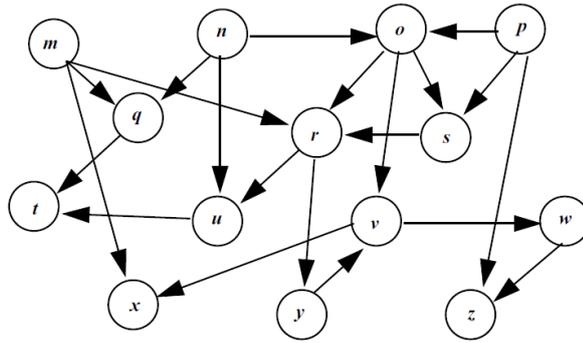
9. При рассмотрении времени выполнения процедуры Quick-Sort мы выяснили, что худшее время выполнения возникает при несбалансированном разбиении массива в процедуре Partition на подмассивы, содержащие 1 и $n-1$ элементов. Приведите пример массива для которого процедура Partition будет выполнять именно такой вид разбиения. Выведите общее правило того, как должен выглядеть массив, чтобы время работы Quick-Sort было квадратичным.

10. Покажите содержимое массива B по окончанию процедуры `Count-Sort()` для входного массива $A = \{5, 2, 3, 1, 2, 1, 5, 1, 7, 2, 3, 5, 7\}$ и $K = 10$.

11. Напишите псевдокод процедуры `Reverse()` для структуры данных связанный список L . Данная процедура обращает содержимое L , так что первый элемент становится последним и наоборот. Время выполнения процедуры должно быть линейным. Дополнительного места в памяти процедура должна требовать не более чем для одного элемента списка и нескольких вспомогательных переменных.

12. Напишите псевдокод процедуры `Size()` для структуры данных очередь Q , представленную на лекции. Приведите асимптотическую оценку времени выполнения данной процедуры.

13. Для графа $G = (V, E)$ представленного на рисунке



▪ покажите результат выполнения алгоритма BFS запущенного из вершины x , т.е. приведите $d[v]$ для всех $v \in V$, нарисуйте BFS-дерево

▪ покажите результат выполнения алгоритма DFS, т.е. укажите $d[v]$ и $f[v]$ для всех $v \in V$, нарисуйте DFS-дерево.

▪ используя результаты предыдущего задания, покажите результат выполнения алгоритма Topological-Sort.

14. Докажите, что если n - это число вершин в двоичном дереве, то высота дерева $h \in [\lceil \log n \rceil, n]$.

15. Перечислите все возможные представления множества чисел $\{6, 3, 1, 3, 8\}$ двоичной кучей.

16. Пусть граф $G = (V, E)$ - дерево. Докажите, что добавление любого ребра в множество E приведет к порождению цикла.

17. Для ненаправленного ациклического графа (дерево) докажите справедливость утверждения $|E| = |V| - 1$.

18. Рассмотрим ненаправленный граф $G = (V, E)$, в котором вершина $v \in V$ представляет пользователя некоторой социальной сети, а ребро $(v_i, v_j) \in E$ свидетельствует о том, что пользователи v_i, v_j обозначили себя друзьями. Для заданного графа социальной сети опишите детали алгоритма, который находит количество обособленных социальных групп, т.е. таких, пользователи которых не имеют общих друзей. Покажите вычислительную сложность предложенного алгоритма.

19. Докажите, что в процессе выполнения алгоритма Дикстра в приоритетной очереди находятся вершины с расчетными расстояниями в диапазоне $[\min \dots \min + c]$, где \min величина ключа последнего удаленного из очереди элемента и $C > 0$ максимальная стоимость грани.

20. Пусть \overline{ab} и \overline{cd} - два отрезка, заданные конечными точками, т.е. известны точки a, b, c и d .

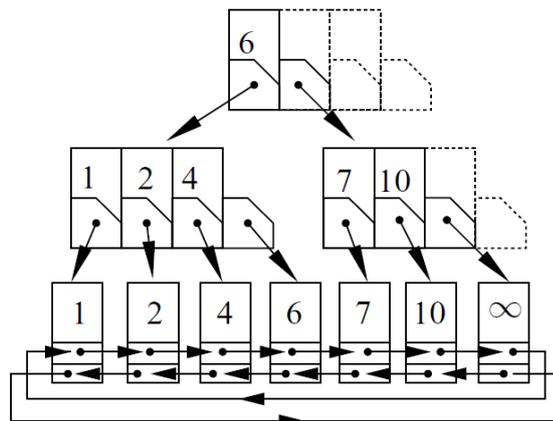
- Покажите как можно установить факт пересечения данных отрезков используя только функцию $\text{orientation}(p,q,r)$;
- Предположим, что отрезки пересекаются. Покажите как можно рассчитать координаты точки пересечения двух отрезков.

21. Пусть P_1 и P_2 - два выпуклых полигона на плоскости, каждый из которых представлен списком вершин (точек). Предложите алгоритм, который определяет выпуклую оболочку для объединения $P_1 \cup P_2$ за время $O(n_1 + n_2)$, где n_1 и n_2 число вершин в P_1 и P_2 , соответственно. (Для удобства можете предположить, что все координаты x точек P_1 меньше чем у вершин полигона P_2).

22. Покажите структуру двоичного дерева поиска высотой 2, 3, 4, 5 и 6 на $\{1, 4, 5, 10, 16, 17, 21\}$.

23. Докажите, что двоичное дерево поиска с $n \geq 2$ может быть построено таким образом, что его высота составит $\lceil \log n \rceil$.

24. Покажите результат вставки (insert) элемента e с $\text{key}(e)=3$ в (2,4)-дерево, представленное ниже.



25. Докажите, что для $a \geq 2$ и $b \geq 2a - 1$ вершины v и t , полученные после разделения некоторой вершины со степенью $b+1$, имеют степень в интервале $[a, b]$.

26. Покажите справедливость утверждения, что функция locate в (a, b) -дереве имеет вычислительную сложность $O(\log n)$.

27. Предложите процедуру построения (a, b) -дерева для сортированного связанного списка L . Покажите результат ее работы для (2,4)-дерева и списка с элементами $[1, 17]$.

28. Предположим, что наряду с операцией MultiPop в интерфейс стека добавлена операция MultiPush. Сохранится ли в этом случае амортизационная стоимость операций стека равной $O(1)$?

29. Рассмотрим последовательность операций, в которой стоимость операции с индексом i равна i , если i является степенью двойки и 1 в противном случае. Оцените амортизационную стоимость одной операции в последовательности из n методом агрегаций, банковским методом и методом потенциалов.

30. При помощи метода потенциалов покажите общую стоимость n операций стека Push, Pop, MultiPop, предполагая, что стек начинает с S_0 объектов и оканчивает работу с S_n .

31. Объясните, как можно реализовать очередь на базе двух стеков. Каково время работы операций Enqueue и Dequeue?

9.6.2 Контрольные вопросы промежуточной аттестации по итогам освоения дисциплины

Примерный перечень вопросов к зачету с оценкой

1. Формальное определение алгоритма.
2. Пример вычислительной проблемы.
3. Формальное описание алгоритма. Отличия от кода языка высокого уровня.
4. Алгоритм пузырьковой сортировки.
5. Худшее время выполнения пузырьковой сортировки.
6. Асимптотический анализ – назначение и пример.
7. Определение и примеры рекуррентного выражения. Пример реализации стратегии раздай и властвуй.
8. Алгоритм сортировка слиянием (Merge-Sort).
9. Концепция алгоритма быстрой сортировки Quicksort. Анализ времени выполнения.
10. Нижняя граница проблемы сортировки для общего случая.
11. Сортировка за линейное время методом Count-Sort.
12. Интерфейс и реализация динамического массива (vector).
13. Интерфейс и реализация стека. Стратегия LIFO.
14. Интерфейс и реализация очередь. Стратегия FIFO.
15. Формальное определение графа.
16. Представление графа последовательностью граней.
17. Представление графа матрицей смежностей.
18. Представление графа массивом смежных вершин.
19. Представление графа списком смежных вершин.
20. Проблема нахождения кратчайшего пути.
21. Алгоритм Дикстра. Общее время выполнения.

22. Определение приоритетная очередь.
23. Наивная реализация приоритетной очереди.
24. Реализация приоритетной очереди очередью корзин.
25. Проблема выпуклой оболочки и алгоритмы ее построения.

Примерный перечень вопросов к зачету с оценкой

1. Роль асимптотической нотации в определении производительности алгоритмов и структур данных.
2. Алгоритмы сортировки – принципы действия и производительность. Нижняя граница проблемы сортировки в общем случае.
3. Алгоритмы обхода графа. Сравнение и анализ.
4. Проблема поиска кратчайших путей.
5. Алгоритм Дикстры с различными приоритетными очередями.
6. Методы нахождения выпуклой оболочки.
7. Двоичное дерево поиска пример и назначение.
8. Структура (a,b) – дерева поиска.
9. Формальное определение хэш-функции.
10. Амортизационный анализ – назначение и примеры использования.
11. Метод агрегаций и банковский метод.
12. Минимальные покрывающие деревья.
13. Поиск подстроки. Алгоритмы поиска и анализ времени выполнения.
14. Проблема максимального потока.
15. Примеры полиномиальных и неполиномиальных алгоритмов.

Требования к содержанию экзаменационных билетов

Экзаменационные билеты включают, как правило, два теоретических вопроса и расчетную задачу, аналогичную текущему контролю успеваемости.

10 Методические рекомендации для обучающихся по освоению дисциплины

Методика преподавания дисциплины «Алгоритмы и структуры данных» характеризуется совокупностью методов, приемов и средств обучения, обеспечивающих реализацию содержания и учебно-воспитательных целей дисциплины, которая может быть представлена как некоторая методическая система, включающая методы, приемы и средства обучения. Такой подход позволяет более качественно подойти к вопросу освоения дисциплины обучающимися.

Одну из основных ролей в организации учебного процесса по данной дисциплине играют лекционные занятия. В ходе занятий осуществляется теоретическое обучение студентов, привитие им необходимых умений и практических навыков, приобретаемых при изучении дисциплины.

Учебные занятия начинаются и заканчиваются по времени в соответствии с утвержденным режимом университета в аудиториях согласно семестровым расписаниям теоретических занятий. Допуск в аудиторию студентов, опоздавших на 15 минут от начала пары и более, запрещается. На занятиях, предусмотр-

ренных расписанием, обязаны присутствовать все обучающие. Освобождение студентов от занятий может проводиться только по письменным распоряжениям представителей деканатом. Преподаватель обязан лично контролировать наличие студентов на занятиях.

Основными видами учебных занятий по дисциплине являются лекции, практические занятия, консультации. Объем и виды учебных занятий определены представленной рабочей программой дисциплины.

Лекции являются одним из важнейших видов образовательных технологий и составляют основу теоретической подготовки студентов по дисциплине. Они должны давать систематизированные основы научных знаний, концентрировать внимание студентов на наиболее сложных, проблемных вопросах, стимулировать их активную познавательную деятельность и способствовать формированию творческого и профессионального мышления.

Каждая лекция должна представлять собой устное изложение лектором основных теоретических положений изучаемой дисциплины или отдельной темы как логически законченное целое и иметь конкретную целевую установку. Лекции должны носить, как правило, проблемный характер. Основным методом в лекции выступает устное изложение лектором учебного материала, сопровождающееся демонстрацией схем, мультимедийных презентаций, диаграмм.

Порядок изложения материала лекции отражается в плане ее проведения.

Особое место в лекционном курсе по дисциплине занимают вводная и заключительная лекции.

Вводная лекция должна давать общую характеристику изучаемой дисциплины, подчеркивать новизну проблем, указывать ее роль и место в системе (структурно-логической схеме) изучения других дисциплин, раскрывать учебные и воспитательные цели и кратко знакомить студентов с содержанием и структурой курса, а также с организацией учебной работы по нему. На вводной лекции проводится входной контроль с целью установления общего уровня компетенций, освоенных студентом в ранее изученных дисциплинах.

Заключительная лекция должна давать научно-практическое обобщение изученной дисциплины, показывать перспективы развития изучаемой области знаний, навыков и практических умений.

Практические занятия по дисциплине имеют целью:

- углубление, расширение и конкретизацию теоретических знаний, полученных на лекции, до уровня, на котором возможно их практическое использование;

- отработку навыков и умений в пользовании соответствующем математическим и алгоритмическим аппаратом;

- отработку умения решения реальных прикладных задач;

- проверку теоретических знаний.

Основу практических занятий составляет работа каждого обучаемого (индивидуальная и (или) коллективная, по приобретению умений и навыков использования закономерностей, принципов, методов, форм и средств, составляющих содержание дисциплины в профессиональной деятельности и в подготовке к изучению дисциплин, формирующих компетенции выпускника). Прак-

тическим занятиям предшествуют лекции и целенаправленная самостоятельная подготовка студентов, поэтому практические занятия нужно начать с краткого обзора цели занятия, напоминания о его связи с лекциями. На практическом занятии студентам выдается в печатном виде блок вопросов (заданий). В ходе практического занятия студентом решаются указанные вопросы и на очередном практическом занятии проводится индивидуальная защита решений, оформленных отчетом, с целью установления их корректности и степени овладения той или иной компетенции. В случае нехватки времени, отведенного на практическое занятие, для нахождения решения задач, студент обязан выполнить их в рамках самостоятельной работы.

По результатам контроля знаний и умений преподаватель должен провести анализ хода и итогов практических занятий, отметить успехи студентов в решении учебной задачи, а также недостатки и ошибки, разобрать их причины и дать методические указания к их устранению. Таким образом, практические занятия являются важной формой обучения, в ходе которых знания студентов превращаются в профессиональные необходимые умения, навыки и компетенции.

Консультации являются одной из форм руководства работой студентов и оказания им помощи в самостоятельном изучении учебного материала. Они проводятся регулярно в процессе всего периода обучения (по мере возникновения потребности) по предварительной договоренности студентов с лектором (преподавателем) в часы самостоятельной работы и носят индивидуальный характер. При необходимости разъяснения общих вопросов нескольким или всем обучающимся учебной группы проводятся групповые консультации.

Преподаватель имеет право вызывать на консультацию тех студентов, которые не показывают глубоких знаний и не пользуются консультациями по своей инициативе. В этих случаях, преподаватель выясняет, работает ли студент систематически над учебным материалом, в какой степени усваивает его, в чем встречает наибольшие трудности. Установив фактическое положение дела, преподаватель дает рекомендации по самостоятельному изучению материала, решению трудных вопросов и при необходимости назначает срок повторной консультации.

Зачет и экзамен являются заключительными оценочными средствами в 5 и 6 семестре, по итогам которых выявляется общий уровень овладения студентом предусмотренных компетенций по тематическим вопросам всего курса.

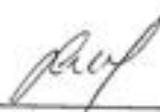
Рабочая программа дисциплины составлена в соответствии с требованиями ФГОС ВО по направлению подготовки 25.03.03 «Аэронавигация».

Программа рассмотрена и утверждена на заседании кафедры № 8 «Прикладной математики и информатики»

« 18 » 05 2021 года, протокол № 8.

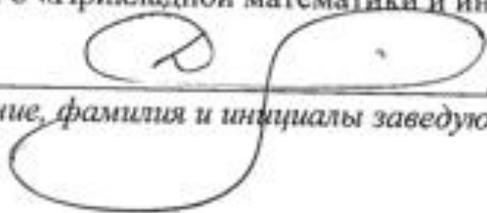
Разработчик:

к.т.н.


Земсков Ю.В.
(ученая степень, ученое звание, фамилия и инициалы разработчика)

Заведующий кафедрой № 8 «Прикладной математики и информатики»

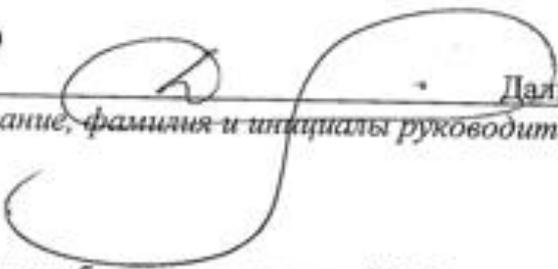
к.т.н., доцент


Далингер Я.М.
(ученая степень, ученое звание, фамилия и инициалы заведующего кафедрой)

Программа согласована:

Руководитель ОПОП ВО

к.т.н., доцент


Далингер Я.М.
(ученая степень, ученое звание, фамилия и инициалы руководителя ОПОП)

Программа рассмотрена и одобрена на заседании Учебно-методического совета Университета « 16 » июни 2021 года, протокол № 7.