

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ 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)

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

Рабочая программа дисциплины

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

Математическое моделирование и вычислительная математика

(указывается профиль / магистерская программа / специализация)

Форма обучения

очная

(очная / очно-заочная / заочная)

Нижегород

2023 год

1. Место дисциплины в структуре ООП

Дисциплина относится к обязательной части.

Б1.О.11 «Алгоритмы и структуры данных».

№ варианта	Место дисциплины в учебном плане образовательной программы	Стандартный текст для автоматического заполнения в конструкторе РПД
1	Блок 1. Дисциплины (модули) Обязательная часть	Дисциплина Б1.О.11 «Алгоритмы и структуры данных» относится к обязательной части ООП направления подготовки 01.03.02 <i>Прикладная математика и информатика</i> ».

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

Формируемые компетенции (код, содержание компетенции)	Планируемые результаты обучения по дисциплине (модулю), в соответствии с индикатором достижения компетенции		Наименование оценочного средства
	Индикатор достижения компетенции (код, содержание индикатора)	Результаты обучения по дисциплине	
ПК-2. Способен к разработке и применению алгоритмических и программных решений в области системного и прикладного программного обеспечения	<i>ПК-2.1. Знает приемы и методы разработки алгоритмических решений, программного обеспечения для анализа, распознавания и обработки информации</i>	<i>Знать общие принципы использования вычислительной техники для решения прикладных задач; основные идеи, лежащие в основе современных языков программирования и основные составляющие современного языка программирования C++; технологии разработки программного обеспечения (структурное, модульное и объектно-ориентированное программирование) и способы их выражения в языке программирования C++; основные виды структур данных и методы их эффективной реализации.</i>	<i>Собеседование по вопросам к зачету/экзамену</i>
	<i>ПК-2.2. Умеет разрабатывать и применять алгоритмические и программные решения в области системного и прикладного программного обеспечения</i>	<i>Уметь разрабатывать и отлаживать программы средней сложности с использованием языка программирования C++ и технологии объектно-ориентированного программирования.</i>	<i>Практическая работа</i>

	<i>ПК-2.3. Имеет практические навыки разработки программного обеспечения для анализа, распознавания и обработки информации</i>	<i>Владеть современными интегрированными средами разработки программ; навыками реализации, тестирования и отладки программных систем средней сложности.</i>	<i>Практическая работа</i>
--	--	---	----------------------------

3. Структура и содержание дисциплины

3.1. Трудоемкость дисциплины

Семестр 3

	Очная форма обучения
Общая трудоемкость	4 ЗЕТ
Часов по учебному плану	144
в том числе	
аудиторные занятия (контактная работа):	97
- занятия лекционного типа	32
- занятия семинарского типа	32
- занятия лабораторного типа	32
- текущий контроль (КСР)	1
самостоятельная работа	47
Промежуточная аттестация – зачет	

Семестр 4

	Очная форма обучения
Общая трудоемкость	5 ЗЕТ
Часов по учебному плану	180
в том числе	
аудиторные занятия (контактная работа):	98
- занятия лекционного типа	32
- занятия семинарского типа	32
- занятия лабораторного типа	32
- текущий контроль (КСР)	2
самостоятельная работа	46
Промежуточная аттестация – экзамен	36

3.2. Содержание дисциплины

Наименование и краткое содержание разделов и тем дисциплины	Всего (часы)	В том числе					Самостоятельная работа обучающегося, часы
		Контактная работа (работа во взаимодействии с преподавателем), часы. Из них				Всего	
		Занятия лекционного типа	Занятия семинарского типа	Занятия лабораторного типа	Всего		
Семестр 3							
Введение в предмет. Структура и содержание курса. Литература.	13	2			2	11	
Структуры действия и структуры данных	66	16	16	16	48	18	
Динамические структуры и представление на ЭВМ сложных математических моделей	64	14	16	16	46	18	
Текущий контроль (КСР)	1				1		
Промежуточная аттестация – зачет							
Итого	144	32	32	32	97	47	
Семестр 4							
Организация доступа по имени	48	10	10	10	30	18	
Проблемное языковое обеспечение	48	10	10	10	30	18	
Автоматизация управления ЭВМ и операционные системы	46	12	12	12	36	10	
Текущий контроль (КСР)	2				2		
Промежуточная аттестация – экзамен	36						
Итого	180	32	32	32	98	46	

Практические занятия (лабораторные работы) организуются, в том числе в форме практической подготовки, которая предусматривает участие обучающихся в выполнении отдельных элементов работ, связанных с будущей профессиональной деятельностью.

Практическая подготовка предусматривает: разработку и отладку программ средней сложности с использованием языка программирования C++ и технологии объектно-ориентированного программирования; освоение инструментов разработки программного обеспечения, таких как система контроля версий Git и фреймворк для разработки автоматических тестов Google Test.

На проведение практических занятий (лабораторных работ) в форме практической подготовки отводится 64 часа.

Практическая подготовка направлена на формирование и развитие:

- практических навыков в соответствии с профилем ОП: разработка, отладка, проверка работоспособности, модификация программного обеспечения на основе анализа математических моделей различных естественнонаучных, информационных процессов; разработка технической документации по программному обеспечению;

- компетенций - ПК-2: Способен к разработке и применению алгоритмических и программных решений в области системного и прикладного программного обеспечения (ПК-2.3: Имеет практические навыки разработки программного обеспечения для анализа, распознавания и обработки информации).

Текущий контроль успеваемости реализуется в формах опросов на занятиях семинарского типа, лабораторного типа.

На практических занятиях студенты под руководством преподавателя практики разбирают лекционный материал, постановки лабораторных работ, решают задачи по темам курса (задачи указаны в ФОС в описании к лабораторным).

Промежуточная аттестация проходит в традиционных формах (зачет, экзамен).

4. Учебно-методическое обеспечение самостоятельной работы обучающихся

Самостоятельная работа студентов подразделяется на следующие категории:

- Изучение учебной литературы (см. перечень образовательных материалов).
- Выполнение лабораторных работ на следующие темы:
 - Лабораторная работа 1. Структура хранения множеств.
 - Лабораторная работа 2. Структуры хранения матриц специального вида.
 - Лабораторная работа 3. Вычисление арифметических выражений.
 - Лабораторная работа 4. Обслуживание процессором ЭВМ очереди заданий.
 - Лабораторная работа 5. Система для арифметических действий над полиномами.
 - Лабораторная работа 6. Редактирование текстов.
 - Лабораторная работа 7. Структуры хранения геометрических объектов.
 - Лабораторная работа 8. Сравнительная характеристика способов организации таблиц.

Образовательные материалы для самостоятельной работы студентов:

- Барышева И.В., Мееров И.Б., Сысоев А.В., Шестакова Н.В. Лабораторный практикум (по программе "Алгоритмы и структуры данных") Учебно-методическое пособие. http://www.unn.ru/books/met_files/Pract_ADS.pdf
- Технопарк Mail.ru Group. Курс "Алгоритмы и структуры данных". <http://www.intuit.ru/studies/courses/3496/738/info>
- Седжвик Р. Курс "Алгоритмы на C++". <http://www.intuit.ru/studies/courses/12181/1174/info>
- Алексеев В., Таланов В. Курс "Структуры данных и модели вычислений". <http://www.intuit.ru/studies/courses/100/100/info>
- Meyer В. Курс "Инструменты, алгоритмы и структуры данных". <http://www.intuit.ru/studies/courses/683/539/info>

Также для обеспечения самостоятельной работы обучающихся используются электронные курсы (Алгоритмы и структуры данных <https://e-learning.unn.ru/enrol/index.php?id=1617>, Алгоритмы и структуры данных 2 <https://e-learning.unn.ru/enrol/index.php?id=1618>),

созданные в системе электронного обучения ННГУ - <https://e-learning.unn.ru/>, и в системе открытых онлайн-курсов - МООС - <https://mooc.unn.ru/> (Алгоритмы: теория и практика. Структуры данных. <https://stepik.org/course/1547/>).

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

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

5.1. Описание шкал оценивания результатов обучения по дисциплине

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

	навыков вследствие отказа обучающегося от ответа	навыки. Имели место грубые ошибки.	задач с некоторыми недочетами.	задач с некоторыми недочетами	задач без ошибок и недочетов.	ошибок и недочетов.	ых задач.
--	--	------------------------------------	--------------------------------	-------------------------------	-------------------------------	---------------------	-----------

Шкала оценки при промежуточной аттестации

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

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

5.2.1 Контрольные вопросы Контрольные вопросы к зачёту

вопросы	Код формируемой компетенции
1. Проблема доказательства правильности программ.	ПК-2
2. Способы снижения сложности программного обеспечения.	ПК-2
3. Рекурсивное описание вычислительного процесса и структуры данных.	ПК-2
4. Структуры данных и математические структуры.	ПК-2
5. Переменные структуры и схемы структуры.	ПК-2
6. Понятие экземпляра, схемы структуры.	ПК-2
7. Линейные структуры данных.	ПК-2
8. Структура машинной памяти. Вектор памяти как образ линейной структуры.	ПК-2
9. Динамические структуры.	ПК-2
10. Структуры хранения динамических структур типа стек.	ПК-2
11. Структуры хранения динамических структур типа очередь.	ПК-2
12. Сравнение структур хранения линейных и динамических структур.	ПК-2

13. Статическое и динамическое распределение памяти.	ПК-2
14. Управление памятью путем перепаковки структур хранения.	ПК-2
15. Структура хранения нескольких стеков в общей памяти.	ПК-2
16. Роль гипотез о росте структур при разработке систем управления памятью путем перепаковки.	ПК-2
17. Оценка параметров модели в ходе выполнения программ (адаптация).	ПК-2
18. Линейный список.	ПК-2
19. Способы реализации списков на языках высокого уровня.	ПК-2
20. Управление свободной памятью при использовании сцепления.	ПК-2
21. Реализация структуры хранения нескольких стеков с использованием списков на языке высокого уровня.	ПК-2
22. Сравнение непрерывной и списковой структур хранения.	ПК-2
23. Динамическое распределение памяти в языке C/C++ (выделение и освобождение памяти).	ПК-2
24. Реализация стека с использованием динамически распределяемой памяти.	ПК-2
25. Пример использования стеков: поразрядная сортировка.	ПК-2
26. Пример использования стеков: преобразование арифметических выражений в польскую форму записи.	ПК-2
27. Разработка общего представления линейного списка для обеспечения списковой структуры хранения.	ПК-2
28. Общая характеристика стандартной библиотеки шаблонов.	ПК-2
29. Система для арифметических действий над полиномами (представление полиномов, управление памятью, выполнение операций).	ПК-2
30. Представление многочленов от нескольких переменных. Исключение хранения мономов с нулевыми коэффициентами.	ПК-2
31. Схема наследования программ для обеспечения структуры хранения полиномов.	ПК-2
32. Реализация программ для обеспечения работы с линейным циклическим списком.	ПК-2
33. Структура класса для представления на ЭВМ полиномов от нескольких переменных.	ПК-2
34. Алгоритм сложения многочленов от нескольких переменных.	ПК-2
35. Представление текста связным списком.	ПК-2
36. Операторы объединения списков и расчленения списка.	ПК-2
37. Алгоритм обхода иерархического списка.	ПК-2
38. Копирование списка.	ПК-2
39. Сборка мусора.	ПК-2
40. Плексы как представление рисунков, состоящих из точек и соединяющих их отрезков.	ПК-2
41. Алгоритм обхода плекса.	ПК-2
42. Алгоритм вставки линии.	ПК-2
43. Плекс, как представление арифметического выражения.	ПК-2
44. Организация доступа по имени. Таблицы. Поиск по ключу (просмотр и двоичный поиск).	ПК-2
45. Упорядоченные таблицы. Алгоритм сортировки включением.	ПК-2
46. Упорядоченные таблицы. Алгоритм сортировки слиянием.	ПК-2
47. Представление таблиц с использованием деревьев поиска.	ПК-2
48. Деревья поиска. Алгоритмы обхода.	ПК-2
49. Деревья поиска. Алгоритмы поиска и вставки.	ПК-2
50. Деревья поиска. Алгоритм удаления.	ПК-2
51. Сбалансированные и идеально сбалансированные деревья поиска. Общая схема балансировки при вставке.	ПК-2

52. Таблицы с вычислимым входом. Запись и поиск при переполнении (способ открытого перемешивания).	ПК-2
53. Определение формального языка.	ПК-2
54. БНФ-форма задания грамматики формального языка.	ПК-2
55. Представление грамматик с помощью синтаксических диаграмм. Порождение языковых цепочек в результате обхода диаграмм.	ПК-2
56. Контекстно-свободные грамматики (терминалы и нетерминалы, правила вывода).	ПК-2
57. Распознавание операторов формального языка.	ПК-2
58. Пример описания грамматики языка арифметических выражений.	ПК-2
59. Прохождение задачи в среде операционной системы.	ПК-2
60. Понятие прерывания. Привилегированный режим.	ПК-2
61. Классификация прерываний.	ПК-2
62. Совмещение работы устройств и многопрограммный режим.	ПК-2
63. Понятие процесса и ресурса в операционной системе. Deskрипторы.	ПК-2
64. Понятия состояния операционной системы. Граф "процесс-ресурс".	ПК-2
65. Модель управления процессами и ресурсами в операционной системе в форме асинхронного конечного автомата.	ПК-2

Контрольные вопросы к экзамену

Вопрос	Код компетенции (согласно РПД)
1. Понятие экземпляра, схемы структуры.	ПК-2
2. Способы реализации списков на языках высокого уровня.	ПК-2
3. Линейные структуры данных	ПК-2
4. Управление свободной памятью при использовании сцепления.	ПК-2
5. Структура хранения множеств. Класс TBitField – операции присваивания, побитового умножения И, побитового сложения ИЛИ, отрицания, перегрузка вывода	ПК-2
6. Реализация структуры хранения нескольких стеков с использованием списков на языке высокого уровня.	ПК-2
7. Структуры хранения для матриц специального вида. Класс TVector – конструктор копирования, операция сложения	ПК-2
8. Сравнение непрерывной и списковой структур хранения.	ПК-2
9. Структура хранения множеств. Реализация класса TSet с использованием шаблонов	ПК-2
10. Реализация стека с использованием динамически распределяемой памяти.	ПК-2
11. Структура хранения множеств. Реализация класса TSet через наследование класса TBitField	ПК-2
12. Пример использования стеков: преобразование арифметических выражений в польскую форму записи.	ПК-2
13. Структуры хранения для матриц специального вида. Класс TVector – конструктор копирования, операция сложения	ПК-2
14. Разработка общего представления линейного списка для обеспечения списковой структуры хранения. Организация хранения в списках значений разного типа	ПК-2
15. Структуры хранения для матриц специального вида. Класс TMatrix – конструктор копирования, операция сложения	ПК-2
16. Разработка общего представления линейного списка для обеспечения списковой структуры хранения. Проектирование структуры списка, структура наследования классов	ПК-2
17. Структуры хранения для матриц специального вида. Разработка плотной (непрерывной) структуры хранения верхних треугольных матриц.	ПК-2
18. Разработка общего представления линейного списка для обеспечения списковой структуры хранения. Навигация по списку (итератор)	ПК-2
19. Структуры хранения для матриц специального вида. Разработка структуры хранения матриц как набора векторов разной длины	ПК-2
20. Разработка общего представления линейного списка для обеспечения списковой структуры хранения. Операции вставки звеньев	ПК-2
21. Операции вставки звеньев	ПК-2
22. Разработка общего представления линейного списка для обеспечения списковой структуры хранения. Операции удаления звеньев	ПК-2

23. Структуры хранения динамических структур типа стек и очередь. Разработка структуры хранения стека	ПК-2
24. Разработка общего представления линейного списка для обеспечения списковой структуры хранения. Разработка класса-переходника	ПК-2
25. Структуры хранения динамических структур типа стек и очередь. Разработка структуры хранения очереди	ПК-2
26. Разработка общего представления линейного списка для обеспечения списковой структуры хранения. Расширение набора операций со списком (пример)	ПК-2
27. Структуры хранения динамических структур типа стек и очередь. Разработка структуры хранения дека	ПК-2
28. Разработка общего представления линейного списка для обеспечения списковой структуры хранения. Разработка двунаправленного списка.	ПК-2
29. Сравнение структур хранения линейных и динамических структур.	ПК-2
30. Общая характеристика стандартной библиотеки шаблонов.	ПК-2
31. Статическое и динамическое распределение памяти.	ПК-2
32. Представление многочленов от нескольких переменных. Исключение хранения мономов с нулевыми коэффициентами.	ПК-2
33. Управление памятью путем перепаковки структур хранения.	ПК-2
34. Схема наследования программ для обеспечения структуры хранения полиномов.	ПК-2
35. Структура хранения нескольких стеков в общей памяти. Структура памяти, начальное распределение, оценка свободной памяти, перераспределение свободной памяти	ПК-2
36. Реализация программ для обеспечения работы с линейным циклическим списком.	ПК-2
37. Роль гипотез о росте структур при разработке систем управления памятью путем перепаковки. Гипотеза о сохранении локальной тенденции роста структур.	ПК-2
38. Структура класса для представления на ЭВМ полиномов от нескольких переменных.	ПК-2
39. Роль гипотез о росте структур при разработке систем управления памятью путем перепаковки. Понятие смешанной стратегии распределения свободной памяти.	ПК-2
40. Алгоритм сложения многочленов от нескольких переменных.	ПК-2
41. Адаптивная оценка параметров модели в ходе выполнения программ (на примере системы управления несколькими стеками).	ПК-2
42. Вычисление значения полинома при заданных значениях переменных.	ПК-2
43. Понятие линейного списка.	ПК-2
44. Вычисление частных производных полинома от нескольких переменных.	ПК-2

5.2.2. Типовые задания для оценки сформированности компетенции ПК-2

1) Лабораторная работа 1. Структура хранения множеств.

Цель практической работы – разработка структуры данных для хранения множеств с использованием битовых полей, а также освоение таких инструментов разработки программного обеспечения, как система контроля версий Git и фреймворк для разработки автоматических тестов Google Test.

Перед выполнением работы студенты получают данный проект-шаблон, содержащий следующее:

- Интерфейсы классов битового поля и множества (h-файлы)
- Готовый набор тестов для каждого из указанных классов
- Пример использования класса битового поля и множества для решения задачи поиска простых чисел с помощью алгоритма "Решето Эратосфена"

Выполнение работы предполагает решение следующих задач:

- Реализация класса битового поля TBitField согласно заданному интерфейсу.
- Реализация класса множества TSet согласно заданному интерфейсу.

- Обеспечение работоспособности тестов и примера использования.
- Реализация нескольких простых тестов на базе Google Test.
- Публикация исходных кодов в личном репозитории на GitHub.

6. Учебно-методическое и информационное обеспечение дисциплины

а) основная литература:

- Барышева И.В., Мееров И.Б., Сысоев А.В., Шестакова Н.В. Лабораторный практикум (по программе "Алгоритмы и структуры данных") Учебно-методическое пособие. http://www.unn.ru/books/met_files/Pract_ADS.pdf
- Технопарк Mail.ru Group. Курс «Алгоритмы и структуры данных». <http://www.intuit.ru/studies/courses/3496/738/info>

б) дополнительная литература:

- Седжвик Р. Курс «Алгоритмы на C++». <http://www.intuit.ru/studies/courses/12181/1174/info>
- Алексеев В., Таланов В. Курс «Структуры данных и модели вычислений». <http://www.intuit.ru/studies/courses/100/100/info>
- Meyer В. Курс «Инструменты, алгоритмы и структуры данных». <http://www.intuit.ru/studies/courses/683/539/info>

в) программное обеспечение и Интернет-ресурсы (в соответствии с содержанием дисциплины)

- В системе открытых онлайн-курсов - МООС - <https://mooc.unn.ru/> (Алгоритмы: теория и практика. Структуры данных. <https://stepik.org/course/1547/>).

7. Материально-техническое обеспечение дисциплины

Помещения представляют собой учебные аудитории для проведения учебных занятий, предусмотренных программой, оснащенные оборудованием и техническими средствами обучения.

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

Программа составлена в соответствии с требованиями ОС ННГУ 01.03.02 Прикладная математика и информатика.

Авторы: В.П. Гергель,

А.В. Сысоев

Заведующий кафедрой МОСТ Р.Г. Стронгин

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

от 30.11.2022 года, протокол № 3.