

Аннотация к рабочей программы дисциплины  
Б1.О.19«Алгоритмы и структуры данных»

**Объем трудоемкости:** 3 зачетных единиц

**Цель дисциплины:**

Изучение базовых алгоритмов и структур данных, их реализация на C++ с акцентом на эффективность и применение в машинном обучении.

**Задачи дисциплины**

Освоение структур данных: списки, стеки, очереди, деревья, графы.

Изучение алгоритмов сортировки и поиска.

Применение рекурсии и динамического программирования.

Анализ сложности алгоритмов (O-нотация)

**Место дисциплины в структуре образовательной программы**

Дисциплина «Алгоритмы и структуры данных» относится к «Обязательная часть» Блока 1 «Дисциплины (модули)» учебного плана.

**Требования к уровню освоения дисциплины**

Изучение данной учебной дисциплины направлено на формирование у обучающихся следующих компетенций:

**ОПК-2** **Способен проводить под научным руководством исследование на основе существующих методов в конкретной области профессиональной деятельности**

**ОПК-2.1** **Применяет современные математические и вычислительные методы для решения научных задач в рамках поставленной проблемы**

**Знать** Знать современный математический аппарат и численные методы, необходимые для формализации, анализа и алгоритмического решения типовых задач в профессиональной научной области.

**Уметь** Уметь выбирать и корректно применять соответствующие математические модели и вычислительные алгоритмы для получения численного решения поставленной научной или инженерной проблемы.

**Владеть** Владеть навыками реализации вычислительных методов на практике с использованием специализированного программного обеспечения или языков программирования, а также анализа достоверности и точности полученных результатов.

**ОПК-2.2** **Формулирует гипотезы, планирует и выполняет эксперименты, обрабатывает и интерпретирует полученные данные с использованием специализированного ПО**

**Знать** Знать методологию научного эксперимента: принципы формулирования проверяемых гипотез, планирования эксперимента, сбора данных, их статистической обработки и интерпретации результатов с использованием соответствующего программного обеспечения.

**Уметь** Уметь формулировать гипотезы относительно поведения алгоритмов или систем, планировать серии вычислительных экспериментов для их проверки, автоматизировать сбор метрик (время выполнения, потребление памяти) и анализировать полученные данные.

**Владеть** Владеть навыками работы со специализированным ПО (среды разработки, профайлеры, системы визуализации данных, статистические пакеты) для проведения экспериментов, обработки эмпирических данных и представления результатов в виде графиков и выводов.

<b>ОПК-4</b>	<b>Способен находить, анализировать, реализовывать программно и использовать на практике математические алгоритмы, в том числе с применением современных вычислительных систем</b>
<b>ОПК-4.1</b>	<b><i>Разрабатывает и оптимизирует алгоритмы с учетом вычислительной сложности и аппаратных ограничений</i></b>
<b>Знать</b>	Знать критерии оценки алгоритмов (временная и пространственная сложность, амортизационный анализ), принципы их оптимизации (жадные алгоритмы, динамическое программирование, эвристики), а также влияние архитектуры вычислительных систем (иерархия памяти, параллелизм) на производительность.
<b>Уметь</b>	Уметь анализировать задачу, разрабатывать и сравнивать алгоритмические решения с разной вычислительной сложностью, модифицировать и оптимизировать алгоритмы с учетом выявленных «узких мест» и ограничений целевой аппаратной платформы.
<b>Владеть</b>	Владеть практическими навыками реализации эффективных алгоритмов, их профилирования для поиска критических участков кода, а также применения техник оптимизации (кэширование, оптимизация циклов, выбор структур данных) для повышения производительности в заданных аппаратных условиях.
<b>ОПК-5</b>	<b>Способен понимать принципы работы современных информационных технологий и использовать их для решения задач профессиональной деятельности</b>
<b>ОПК-5.1</b>	<b>Применяет современные языки программирования и технологии для решения математических и вычислительных задач</b>
<b>Знать</b>	Знать синтаксис, парадигмы, основные библиотеки и возможности современных языков программирования, а также технологии, используемые для решения математических и вычислительных задач (символьные вычисления, численные методы, параллельные вычисления)
<b>Уметь</b>	Уметь выбирать и применять подходящий язык программирования, библиотеки численных методов и вычислительные технологии для формализации математической задачи, реализации алгоритма её решения, верификации и визуализации результатов
<b>Владеть</b>	Владеть навыками практического использования выбранных языков программирования и их специализированных библиотек для реализации, отладки и анализа эффективности алгоритмов решения типовых математических и вычислительных задач
<b>ОПК-6</b>	<b>Способен разрабатывать алгоритмы и компьютерные программы, пригодные для практического применения</b>
<b>ОПК-6.1</b>	<b><i>Разрабатывает эффективные алгоритмы и реализует их в виде программного кода с учетом временной и пространственной сложности</i></b>
<b>Знать</b>	Знать методологию разработки и анализа алгоритмов, принципы оценки их временной (O-нотация) и пространственной сложности, а также основные паттерны проектирования эффективных алгоритмов (жадные алгоритмы, «разделяй и властвуй», динамическое программирование)
<b>Уметь</b>	Уметь формулировать задачу в терминах входных/выходных данных, проектировать эффективный алгоритм её решения, анализировать его сложность и транслировать алгоритмическое решение в корректный, читаемый и эффективный программный код.
<b>Владеть</b>	Владеть навыками практической реализации, отладки и эмпирической проверки разработанных алгоритмов, уметь проводить профилирование кода для оценки его реальной производительности и оптимизировать решения с учетом компромисса между временем выполнения и использованием памяти.

- ОПК-6.2** *Применяет методы структурного и объектно-ориентированного программирования, создает модульные и масштабируемые программы*
- Знать** Знать фундаментальные принципы, методологии и синтаксические конструкции структурного (процедурного) и объектно-ориентированного программирования, а также критерии модульности, слабой связанности и высокой связности кода, обеспечивающие его масштабируемость.
- Уметь** Уметь анализировать предметную область задачи и применять адекватные парадигмы (процедурную декомпозицию или объектное моделирование) для проектирования архитектуры программного решения, состоящего из модулей с четкими интерфейсами.
- Владеть** Владеть навыками реализации, интеграции и рефакторинга программных модулей, разработанных с применением принципов структурного и объектно-ориентированного программирования, обеспечивая возможность их повторного использования и относительно легкого расширения функциональности.
- PL-3** **Способен применять языки программирования C/C++ для решения задач в области ИИ**
- PL-3.1** *"Осуществляет выбор инструментов разработки на языке C/C++, приемлемых для создания прикладной системы ИИ с заданными требованиями"*

**Содержание дисциплины:**

Распределение видов учебной работы и их трудоемкости по разделам дисциплины.  
Разделы (темы) дисциплины, изучаемые в 2 семестре

№	Наименование разделов (тем)	Количество часов				
		Всего	Аудиторная работа			Внеаудиторная работа СРС
			Л	ПЗ	ЛР	
1	2	3	4	5	6	7
1.	Анализ сложности алгоритмов.	12	4		4	4
2.	Рекурсия. Деревья (BST, обходы).	12	4		4	4
3.	Алгоритмы сортировки.	12	4		4	4
4.	Работа с файлами	12	4		4	4
5.	Хеш-таблицы, обработка коллизий.	12	4		4	4
6.	Графы	16	6		6	4
7.	Основы ООП: классы и объекты.	14	4		4	4
8.	Контейнерные классы	7,8	2		2	3,8
9.	Базовые алгоритмы машинного обучения	8	2		2	4
<b>ИТОГО по разделам дисциплины</b>		<b>103,8</b>	<b>34</b>		<b>34</b>	<b>35,8</b>
Контроль самостоятельной работы (КСР)		2				
Промежуточная аттестация (ИКР)		0,2				
Подготовка к текущему контролю						
<b>Общая трудоемкость по дисциплине</b>		<b>108</b>				

**Курсовые работы:** *не предусмотрена*

**Форма проведения аттестации по дисциплине:** *зачет*

Автор Добровольская Наталья Юрьевна