

Аннотация к рабочей программы дисциплины
Б1.О.25 «Алгоритмы и структуры данных»

Объем трудоемкости: 4 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 (О-нотация)

Место дисциплины в структуре образовательной программы

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

Дисциплина «Алгоритмы и структуры данных» в качестве пререквизитов требует уверенного владения программированием из первого семестра (циклы, условия, функции, массивы, рекурсия), а также базовых знаний дискретной математики (множества, графы). Постреквизитами выступают дисциплины второго курса, среди которых «Численные методы» (вычислительная сложность, итерационные алгоритмы), «Базы данных» (В-деревья, хеш-таблицы для индексов), «Операционные системы» (очереди процессов, управление памятью), «Современные технологии машинного обучения» (деревья решений, графы вычислений), «WEB-разработка» (структуры данных для бэкенда), «Кроссплатформные десктоп приложения» (UI и кэширование), а также «Технологии компьютерного зрения» (многомерные массивы, алгоритмы свёртки). Смежными дисциплинами, изучаемыми параллельно, являются «Дискретная математика» (графы, деревья, комбинаторика).

Требования к уровню освоения дисциплины

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

- | | |
|----------------|---|
| УК-1 | Способен осуществлять поиск, критический анализ и синтез информации, применять системный подход для решения поставленных задач |
| УК-1.1 | <i>Осуществляет поиск необходимой информации, опираясь на результаты анализа поставленной задачи</i> |
| Знать | Типовые компьютерные методы и современное программное обеспечение для решения задач профессиональной деятельности на языке программирования высокого уровня. |
| Уметь | Установить и использовать программное обеспечение для решения задач профессиональной деятельности.
Применять методы компьютерного проектирования, разработки структур данных с использованием основных алгоритмических и программных решений в области информационно-коммуникационных технологий на языке высокого уровня. |
| Владеть | Проектирование структур данных при реализации алгоритмических и программных решений на языке программирования высокого уровня. |
| УК-1.2 | <i>Выбирает оптимальный вариант решения задачи, аргументируя свой выбор</i> |

Знать	Создание и интеграция программных модулей и компонент программного продукта с использованием языка программирования C++. Типовые алгоритмические и программные решения, библиотеки программных модулей, шаблоны, классы объектов, используемые при разработке программного обеспечения на языке C++. Основные алгоритмические и программные решения, методы и средства проектирования программного обеспечения разработанного на языке C++.
Уметь	Разрабатывать и интегрировать программные модули и компоненты в единый программный продукт с использованием языка программирования C++ в среде Microsoft Visual Studio. Использовать существующие алгоритмические и программные решения и шаблоны проектирования программного обеспечения на языке C++. Применять методы и средства проектирования программного обеспечения, структур данных, интерфейсов с использованием основных алгоритмических и программных решений в области информационно-коммуникационных технологий на языке C++ в среде Microsoft Visual Studio.
Владеть	Разработка и интегрирование программных модулей и компонент в единый программный продукт с использованием языка программирования C++ в среде Microsoft Visual Studio. Проектирование структур данных при реализации алгоритмических и программных решений в среде Microsoft Visual Studio. Проектирование программных продуктов при реализации алгоритмических и программных решений в среде Microsoft Visual Studio с использованием конструкций языка C++.
ОПК-1	Способен применять фундаментальные знания, полученные в области математических и (или) естественных наук, и использовать их в профессиональной деятельности
ОПК-1.2	<i>Применяет фундаментальные знания, полученные в области математических и (или) естественных наук при выборе методов решения задач профессиональной деятельности</i>
Знать	Основные математические концепции, используемые в программировании. Основы алгоритмизации и вычислительной сложности. Математические модели, применяемые в машинном обучении.
Уметь	Применять математические методы для решения задач программирования: выбирать оптимальные структуры данных на основе математического анализа, реализовывать алгоритмы, основанные на математических моделях (например, алгоритмы на графах). Анализировать вычислительную сложность алгоритмов: сравнивать разные подходы к решению задачи с точки зрения эффективности, оптимизировать код, используя математические методы. Использовать математический инструментарий в прикладных задачах.
Владеть	Навыками применения математического аппарата в программировании: умение формализовать задачу в математических терминах, использование библиотек для численных вычислений. Методами оптимизации алгоритмов.
ОПК-2	Способен использовать и адаптировать существующие математические методы и системы программирования для разработки и реализации алгоритмов решения прикладных задач
ОПК-2.1	<i>Способен применять системный подход к анализу предметной (проблемной) области, выявлению требований к реализации алгоритмов решения прикладных задач</i>
Знать	Основы системного анализа и принципы декомпозиции сложных задач. Методы формализации требований к программным системам. Критерии выбора алгоритмов для решения прикладных задач

Уметь	Анализировать предметную область и выделять ключевые сущности. Формулировать требования к алгоритмам на основе анализа задачи. Сравнивать и выбирать оптимальные алгоритмы для реализации.
Владеть	Навыками системного проектирования программных решений. Методами оценки эффективности алгоритмов. Техниками документирования требований и проектных решений.
ОПК-2.2	<i>Применяет современный математический аппарат при построении моделей в различных областях человеческой деятельности</i>
Знать	Основные классы математических моделей (детерминированные, стохастические, дискретные, непрерывные). Современные методы математического моделирования (машинное обучение, оптимизация, теория игр). Математические основы выбранной предметной области (физика, экономика, биология и др.)
Уметь	Формализовывать прикладные задачи в терминах математических моделей. Выбирать и адаптировать математические методы для конкретных практических задач. Критически оценивать адекватность и ограничения построенных моделей
Владеть	Навыками реализации математических моделей с использованием современных вычислительных инструментов. Методами верификации и валидации математических моделей. Техниками визуализации и интерпретации результатов моделирования
ОПК-3	Способен применять и модифицировать математические модели для решения задач в области профессиональной деятельности
ОПК-3.2	<i>Ориентируется в современных положениях и концепциях прикладной математики и программного обеспечения</i>
Знать	Основы прикладной математики: элементы линейной алгебры (матрицы, векторы), Основы математического анализа (производные, интегралы), Простейшие численные методы. Базовые концепции ПО: Основные парадигмы программирования (ООП, процедурное), Принципы работы алгоритмов (сложность, эффективность), Основные структуры данных (массивы, списки, деревья)
Уметь	Применять простейшие математические методы в программах. Реализовывать базовые алгоритмы на языке программирования. Анализировать простые технические задания. Работать с учебной литературой и документацией
Владеть	Навыками реализации математических формул в коде. Основными приемами отладки программ. Простейшими методами оптимизации кода
ОПК-5	Способен разрабатывать алгоритмы и компьютерные программы, пригодные для практического применения
ОПК-5.1	<i>Аргументировано применяет методы проектирования, разработки и реализации программных продуктов и программных комплексов в различных областях человеческой деятельности</i>
Знать	Принципы структурного программирования (функции, модули). Базовые алгоритмы и структуры данных (линейные, древовидные)
Уметь	Формулировать техническое задание для простых программ. Проектировать архитектуру элементарных приложений. Реализовывать алгоритмы по заданным условиям. Проводить отладку и тестирование кода. Аргументировать выбор методов решения
Владеть	Навыками разработки консольных приложений. Техник разбиения задачи на подзадачи. Методами простого тестирования (юнит-тесты). Основами документирования кода. Приемами работы в команде
PL-4	Способен применять языки программирования C/C++ для решения задач в области ИИ

PL-4.2 Разрабатывает и отлаживает системы ИИ на C++ под конкретные аппаратные платформы с ограничениями по вычислительной мощности, в том числе для встроенных систем

Знать фундаментальные алгоритмы и структуры данных C/C++, принципы разработки систем ИИ (машинное обучение, нейросети), а также специализированные библиотеки (TensorFlow/C++ API, OpenCV, Boost) для их эффективной реализации.

Уметь:

Уметь проектировать, кодировать и отлаживать на C/C++ вычислительно сложные модули ИИ-приложений (например, алгоритмы оптимизации, обработку данных, инференс моделей), используя низкоуровневые возможности языка для оптимизации производительности и памяти.

Владеть:

Владеть продвинутыми навыками отладки и профилирования C/C++ кода, методами интеграции AI-компонент в прикладные решения и оптимизации вычислений для работы в реальном времени или с большими данными.

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
			Л	ПЗ	ЛР	
1	2	3	4	5	6	7
1.	Анализ сложности алгоритмов.	16	4		8	4
2.	Рекурсия. Деревья (BST, обходы).	16	4		8	4
3.	Алгоритмы сортировки.	16	4		8	4
4.	Работа с файлами	16	4		8	4
5.	Хеш-таблицы, обработка коллизий.	16	4		8	4
6.	Графы	20	6		10	4
7.	Основы ООП: классы и объекты.	16	4		8	4
8.	Контейнерные классы	15,8	2		8	5,8
9.	Базовые алгоритмы машинного обучения	8	2		2	4
ИТОГО по разделам дисциплины		139,8	34		68	37,8
Контроль самостоятельной работы (КСР)		4				
Промежуточная аттестация (ИКР)		0,2				
Подготовка к текущему контролю						
Общая трудоемкость по дисциплине		144				

Курсовые работы: не предусмотрена

Форма проведения аттестации по дисциплине: зачет

Автор

Добровольская Наталья Юрьевна