

Аннотация к рабочей программы дисциплины Б1.О.14 Методы программирования

Объем трудоемкости: 5 зачетных единиц

Цель дисциплины: изучение фундаментальных структур данных (линейных, двунаправленных и кольцевых списков, двоичных деревьев, графов) и методов их обработки, а также приобретение практических навыков их реализации на языке программирования. Дисциплина формирует основы для применения современных средств вычислительной техники и программных технологий при решении задач в естественнонаучных, общепрофессиональных и специальных дисциплинах.

Задачи дисциплины:

- Освоение принципов организации и обработки связанных списков, включая линейные, двунаправленные и кольцевые структуры.
- Изучение алгоритмов работы с двоичными деревьями (построение, обход, балансировка).
- Знакомство с методами представления графов и базовыми алгоритмами их обработки (поиск в глубину, в ширину, кратчайшие пути).
- Практическое применение структур данных при работе с файлами (чтение, запись, сериализация) и контейнерами STL (vector, list, map, set и др.).
- Развитие навыков оптимизации кода, отладки программ и выбора оптимальных структур данных для решения прикладных задач.

Место дисциплины в структуре образовательной программы

Дисциплина «Методы программирования» относится к обязательной части Блока 1 «Дисциплины (модули)» учебного плана.

Для успешного освоения дисциплины необходимы знания, умения и навыки, полученные при изучении предшествующих дисциплин, таких как «Основы программирования».

Знания и навыки, полученные в ходе изучения дисциплины, являются основой для последующего изучения таких дисциплин, как «Объектно-ориентированное программирование», «Анализ, проектирование и разработка БД», «Программная инженерия», «Технологии тестирования программного обеспечения», а также для выполнения выпускной квалификационной работы.

Требования к уровню освоения дисциплины

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

Код и наименование индикатора*	Результаты обучения по дисциплине (<i>знает, умеет, владеет (навыки и/или опыт деятельности)</i>)
УК-1. Способен осуществлять поиск, критический анализ и синтез информации, применять системный подход для решения поставленных задач.	
ИД-1.УК-1. Осуществляет поиск необходимой информации, опираясь на результаты анализа поставленной задачи.	Знать: современные методы программирования на C++ (ООП, STL); принципы работы и реализации линейных динамических структур, деревьев, графов, контейнеров STL; методы работы с файлами.

Код и наименование индикатора*	Результаты обучения по дисциплине (<i>знает, умеет, владеет (навыки и/или опыт деятельности)</i>)
	Уметь: анализировать требования к структурам данных и выбирать оптимальную реализацию; проверять корректность реализации алгоритмов (верификация кода); применять методы анализа эффективности алгоритмов (О-нотация). Владеть: навыками разработки и оптимизации структур данных на C++; сравнительного анализа различных реализаций; обработки данных из файлов и их структурированного хранения; применения STL-контейнеров и алгоритмов.
ИД-2.УК-1. Выбирает оптимальный вариант решения задачи, аргументируя свой выбор.	Знать: преимущества и недостатки различных структур данных; особенности работы с памятью в C++; критерии выбора алгоритмов.
	Уметь: сравнивать различные подходы к реализации; обосновывать выбор контейнера STL; анализировать производительность кода.
	Владеть: навыками оптимизации структур данных под конкретные задачи; реализации высокоэффективных алгоритмов на графах и деревьях; использования продвинутых возможностей STL.
УК-2. Способен определять круг задач в рамках поставленной цели и выбирать оптимальные способы их решения, исходя из действующих правовых норм, имеющихся ресурсов и ограничений.	
ИД-3.УК-2. Использует принципы проектной методологии для решения профессиональных задач.	Знать: алгоритмы и структуры данных; особенности реализации в C++; критерии выбора структур (время доступа, сложность операций).
	Уметь: выбирать между разными структурами (например, массив и список); обосновывать выбор алгоритма; анализировать производительность кода.
	Владеть: навыками реализации динамических структур (списки, деревья) на C++; применения STL (std::list, std::map, std::unordered_set); оптимизации работы с памятью.
ИД-4.УК-2. Выбирает оптимальный способ решения задач, имеющихся ресурсов и ограничений, оценки рисков на основе проектного инструментария.	Знать: методы работы с файлами в C++; алгоритмы обработки графов (Дейкстра, Флойд); особенности контейнеров STL.
	Уметь: читать/записывать структурированные данные в файлы; реализовывать алгоритмы на графах; использовать итераторы и лямбды для обработки данных.
	Владеть: навыками обработки больших данных с помощью файлового ввода-вывода; применения алгоритмов STL (sort, find,

Код и наименование индикатора*	Результаты обучения по дисциплине (знает, умеет, владеет (навыки и/или опыт деятельности))
	accumulate); реализации графовых алгоритмов на C++.
ОПК-2. Способен применять современный математический аппарат, связанный с проектированием, разработкой, реализацией и оценкой качества программных продуктов и программных комплексов в различных областях человеческой деятельности.	
ИД-1.ОПК-2. Способен применять системный подход к анализу предметной (проблемной) области, выявлению требований к ИС.	Знать: математические основы алгоритмов (теория графов, асимптотический анализ); принципы работы с памятью в C++.
	Уметь: анализировать требования и выбирать подходящую структуру; использовать шаблоны проектирования (итератор, адаптер).
	Владеть: навыками реализации структур данных "с нуля"; сравнения STL и рукописных решений.
ИД-3.ОПК-2. Аргументировано применяет методы проектирования, разработки и реализации программных продуктов и программных комплексов в различных областях человеческой деятельности.	Знать: расширенные возможности C++ (шаблоны, move-семантика, многопоточность); алгоритмы сортировки и поиска для деревьев и графов.
	Уметь: оптимизировать код (кеширование, эффективные обходы); тестировать и отлаживать сложные структуры.
	Владеть: навыками реализации деревьев (бинарные, AVL) и графовых алгоритмов; использования STL для высокопроизводительных вычислений.

Содержание дисциплины:

Распределение видов учебной работы и их трудоемкости по разделам дисциплины.

№	Наименование разделов (тем)	Количество часов				
		Всего	Аудиторная работа			Внеаудиторная работа
			Л	ПЗ	ЛР	
1	2	3	4	5	6	7
1.	Линейные динамические информационные структуры	18	8		8	2
2.	Линейные двунаправленные связные списки	26	10		14	2
3.	Кольцевые списки	10	4		4	2
4.	Двоичные деревья	20	6		12	2
5.	Файлы	14	4		6	4
6.	Контейнеры	26	10		12	4
7.	Обработка графов	24	8		12	4
ИТОГО по разделам дисциплины		138	50		68	20
Контроль самостоятельной работы (КСР)		6				

№	Наименование разделов (тем)	Количество часов				
		Всего	Аудиторная работа			Внеаудиторная работа
			Л	ПЗ	ЛР	
1	2	3	4	5	6	7
	Промежуточная аттестация (ИКР)	0,5				
	Подготовка к текущему контролю	35,5				
	Общая трудоемкость по дисциплине	180				

Курсовые работы: *не предусмотрены*

Форма проведения аттестации по дисциплине: *зачет, экзамен*

Автор: Харченко А.В., доцент, канд. пед наук