

Аннотация рабочей программы дисциплины

## **Структуры и алгоритмы обработки данных**

Направление подготовки

*02.03.02 Фундаментальная информатика и информационные технологии*

Направленность (профиль) образовательной программы

*Сетевые технологии*

# 1 Цели и задачи освоения дисциплины

Цель дисциплины – приобретение студентами теоретических знаний и устойчивых практических навыков в области организации базовых структур данных и алгоритмов работы с этими структурами.

Задачами являются:

- системное освоение базовых структур данных;
- осмысленное использование алгоритмов их обработки;
- выработка умения выбора структуры данных наиболее эффективной для решаемой задачи на основе предъявляемых решению требований.

В результате освоения курса студент должен обладать навыками осмысленного выбора структуры данных при решении конкретной задачи, разработки и программной реализации алгоритмов на основе объектно-ориентированной технологии программирования.

№	Содержание раздела
Раздел 1.1	<b>Предмет и задачи дисциплины.</b> Понятие термина структура данных. Важность эффективного и обоснованного выбора структуры данных. Основные принципы выбора структуры данных при проектировании программного обеспечения. Содержание курса, его связь с другими дисциплинами учебного плана.
Раздел 1.2	<b>Статические и динамические структуры данных.</b> Указатель – базовый элемент программирования при реализации структур данных. Понятия статических и динамических структур данных. Достоинства и недостатки. Понятие указателя. Операции над указателями. Использование указателей как базового элемента при программной реализации различных структур данных
Раздел 1.3	<b>Простейшие структуры данных массив и запись.</b> Определения структур данных массив и запись. Прямой и последовательный методы доступа к данным. Достоинства и недостатки этих методов доступа. Обоснование возможности реализации прямого метода доступа в рассматриваемых структурах. Преимущества и недостатки этих структур
Раздел 1.4	<b>Структура данных стек.</b> Понятие стека. Базовые операции над стеками. Алгоритмы выполнения базовых операций. Программная реализация этих алгоритмов. Примеры областей использования стеков. Организация вызовов процедур и функций и возврата в вызвавший модуль. Обработка алгебраических выражений (обратная польская запись).
Раздел 1.5	<b>Структура данных список.</b> Линейные структуры данных. Понятие списка. Основные виды списков. Базовые операции над списками. Алгоритмы выполнения базовых операций над сцепленными списками. Программная реализация этих алгоритмов. Связанное представление элементов в памяти. Алгоритмы выполнения базовых операций над списками в векторной форме. Программная реализация этих алгоритмов.
Раздел 1.6	<b>Рекурсивные алгоритмы.</b> Рекурсия и ее применение при работе со структурами данных. Понятие рекурсии. Рекурсивные алгоритмы и структуры. Примеры рекурсивных алгоритмов. Области применения рекурсивных алгоритмов.
Раздел 1.7	<b>Древовидные структуры данных.</b> Иерархические структуры. Отображение этих структур в виде древовидных структур. Древовидные структуры данных. Отображение в виде графа. Основные понятия и определения. Понятия сбалансированное, идеально сбалансированное и упорядоченное дерево. Бинарное

	дерево. Базовые операции с бинарными деревьями. Алгоритмы выполнения базовых операций над древовидными структурами. Программная реализация этих алгоритмов.
Раздел 1.8	<b>Структура данных таблица.</b> Хеширование данных. Структура данных таблица. Особенности доступа к данным в таблицах. Понятие хеширования. Организация данных с помощью хеширования. Алгоритмы хеширования. Взаимосвязь размера таблицы и скорости поиска в ней конкретного элемента. Программная реализация алгоритмов хеширования
Раздел 1.9	<b>Сортировка массивов.</b> Понятие сортировки. Основные составные элементы сортировки. Критерии сравнения различных алгоритмов сортировки. Простейшие алгоритмы сортировки простыми включениями, простым выбором, пузырьком, с помощью дерева. Алгоритм сортировки Шелла. Алгоритм сортировки QuickSort. Сравнительный анализ алгоритмов
Раздел 1.10	<b>Итоговый обзор курса.</b> Важность обоснованного выбора структур данных и алгоритмов их обработки при проектировании и реализации программного обеспечения.
Раздел 2.1	<b>Структура программы на C++.</b> Шаги построения. Препроцессор. Функции и передача параметров. Указатели и массивы. Z-строки.
Раздел 2.2	<b>Классы в C++.</b> Виды конструкторов. Время жизни объектов в C++ и C# (.Net). Динамические объекты в C++, управление памятью. Классы с подсчетом ссылок. Наследование в C++. Шаблоны в C++.
Раздел 2.3	<b>Алгоритм Бойера-Мура поиска подстроки.</b> Реализация. Таблица суффиксов и префикс-функция. Сложность алгоритма.
Раздел 2.4	<b>Стандартная библиотека C++ (STL).</b> Основные последовательные и ассоциативные контейнеры и их свойства. Контейнер Pair. Умные указатели. Примеры применения контейнеров
Раздел 2.5	<b>В-Tree.</b> Кучи и их реализация на основе динамического массива vector. Очереди приоритетов. Применения. Контейнеры в .Net. $\lambda$ -выражения. LINQ.
Раздел 2.6	<b>Близость строк по Левенштейну.</b> Реализация ВагнераФишера. Сложность алгоритма. Применения. Разреженные векторы и матрицы. Реализация на основе STL. Применения.
Раздел 2.7	<b>Поиск кратчайшего пути.</b> Волновой алгоритм. Сложность и применения.
Раздел 2.8	<b>Кодирование информации.</b> Алгоритмы компрессии данных. Свойства. Другие применения кодирования.
Раздел 2.9	<b>Особенности множественного наследования в C++.</b> Динамическое приведение типов. Расширение классов. Регулярные выражения
Раздел 2.10	<b>Неявные типы.</b> Сериализация и десериализация. Иерархии классов и реляционные БД. XML-представление данных. JSON представление данных. Пулы объектов в .Net