Приложение № 4
к образовательной программе высшего образования – программе магистратуры
по направлению подготовки 09.04.01 Информатика и вычислительная техника
(направленность (профиль) «Разработка и управление
критической информационной инфраструктурой»)

**АВТОНОМНАЯ НЕКОММЕРЧЕСКАЯ ОБРАЗОВАТЕЛЬНАЯ ОРГАНИЗАЦИЯ ВЫСШЕГО ОБРАЗОВАНИЯ «НАУЧНО-ТЕХНОЛОГИЧЕСКИЙ УНИВЕРСИТЕТ «СИРИУС»
(АНОО ВО «УНИВЕРСИТЕТ «СИРИУС»)**

ДОКУМЕНТ ПОДПИСАН
ЭЛЕКТРОННОЙ ПОДПИСЬЮ
Сертификат: 7835bcdac1c89a3da835a18f6045494f3f404337
Владелец: Некрасова Людмила Андреевна
Действителен: с 08.07.2026 до 08.07.2027

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ (МОДУЛЯ)

«Алгоритмы и структуры данных»

1. Трудоемкость дисциплины (модуля): 3 з.е.

2. Место дисциплины (модуля) в учебном плане:

Обязательная часть «Профессиональной подготовки» в рамках Блока 1. Дисциплины (модули), реализуется в 1 семестре.

3. Цель дисциплины (модуля): обучении использованию существующих алгоритмов для эффективного решения различных задач, понимании основных структур данных и их ключевых операций, а также развитии навыков проектирования, анализа и тестирования алгоритмов. Изучение различных подходов к решению задач и выбору оптимальных алгоритмов и структур данных для конкретных сценариев. Получение навыков в создании эффективных алгоритмов и структур данных с целью оптимизации производительности и решения сложных вычислительных проблем.

4. Задачи дисциплины (модуля):

– сформировать представление о фундаментальных алгоритмах, методах оценки их сложности и подходах к решению NP-полных задач;

– обеспечить владение базовыми и продвинутыми структурами данных, а также навыками выбора эффективных алгоритмов для задач обработки информации;

– научить применять парадигмы жадных алгоритмов, динамического программирования и метода «разделяй и властвуй» при разработке программного обеспечения.

5. Перечень разделов (тем) дисциплины (модуля) и их краткое содержание:

Наименования разделов (тем) дисциплины (модуля)	Содержание разделов (тем) дисциплины (модуля)
Алгоритмы и структуры данных	Алгоритм. Память и время как ресурсы. Вычисление чисел Фибоначчи: экспоненциальный рекурсивный алгоритм, полиномиальный алгоритм, более детальный анализ. О-символика как инструмент оценки ресурсов, различные асимптотики (логарифм, полином, экспонента). Абстрактные типы данных, интерфейс и реализация. Стек, очередь, дек; моделирование на основе массива. Моделирование очереди с помощью двух стеков. Односвязный список, двусвязный список. Корневое дерево: бинарное дерево, дерево с произвольным ветвлением, представление «левый ребёнок—правый сосед». Массивы переменного размера: аддитивная и мультипликативная схемы реаллокации. Анализ учётных стоимостей операций: функция потенциала, истинные и учётные стоимости. Метод ростовщика.

	Метод «разделяй и властвуй». Умножение n-битовых чисел: простой рекурсивный алгоритм, улучшенный рекурсивный алгоритм. Рекуррентные соотношения: основная теорема. Двоичный поиск. Сортировка слиянием: с рекурсией и без.
Алгоритмы сортировки и порядковые статистики	Алгоритмы сортировки и порядковые статистики. Нижняя оценка Omega(n log n) для сортировки сравнениями. Квадратичные сортировки. Сортировка с помощью кучи: очередь с приоритетами, построение кучи за линейное время, частичная сортировка. Быстрая сортировка: анализ среднего времени работы, анализ глубины рекурсии, элиминация хвостовой рекурсии, IntroSort, массивы с малым количеством различных элементов, QuickSort3. Сортировка подсчётом, стабильность. Цифровая сортировка. Порядковые статистики, нахождение за линейное в среднем время. Медиана медиан.
Алгоритмы на графах	Графы и способы их представления, способы использования графов. Поиск в глубину в неориентированных графах, выделение компонент связности. Поиск в глубину в ориентированных графах: ориентированные ациклические графы, топологическая сортировка вершин, наличие стока и истока в ациклическом графе, выделение компонент сильной связности. Нахождение кратчайших путей из одной вершины в невзвешенных графах, поиск в ширину. Нахождение кратчайших путей из одной вершины в графах с положительными весами, алгоритм Дейкстры, оценка времени работы при различных реализациях очереди с приоритетами (массивом, двоичной кучей, d-чной кучей). Нахождение кратчайших путей из одной вершины в графах, в которых есть рёбра отрицательного веса, алгоритм Беллмана-Форда, проверка наличия цикла отрицательного веса.

Жадные алгоритмы и динамическое программирование	Задача о выборе заявок. Коды Хаффмена. Хорновские формулы. Задача о покрытии множествами. Минимальное покрывающее дерево: свойство разреза, жадная стратегия, алгоритм Прима, алгоритм Краскала. Система непересекающихся множеств: представление множеств с помощью деревьев, эвристика сжатия путей, верхняя оценка O(m log*n) на время работы m операций. Общие принципы динамического программирования, часто используемые подзадачи. Кратчайшие пути в ациклических ориентированных графах. Наибольшая возрастающая подпоследовательность: подзадачи, порядок на подзадачах, граф подзадач, сравнение с рекурсивным алгоритмом; нахождение не только длины, но и самой подпоследовательности. Редакционное расстояние: граф на подзадачах, нахождение кратчайшего пути в данном графе; нахождение оптимального выравнивания с использованием линейной памяти. Кратчайшие пути в ациклических ориентированных графах. Кратчайшие пути между всеми парами вершин: алгоритм Флойда-Уоршолла.
Хеширование	Прямая адресация. Хеширование. Коллизии. Разрешение коллизий методом цепочек, методом последовательных проб и методом двойного хеширования. Гипотеза равномерного хеширования. Оценка времени поиска в хеш-таблице при использовании метода цепочек для гипотезы равномерного хеширования. Алгоритм Карпа-Рабина. Универсальные семейства хеш-функций. Построение универсального семейства для целочисленных ключей. Оценка времени поиска в хештаблице при использовании метода цепочек для универсального семейства. Совершенные хеш-функции. Построение совершенной хеш-функции с

	помощью универсального семейства. Поиск подстроки в строке.
Деревья поиска	Дерево поиска: поиск, вставка, удаление, поиск следующего и предыдущего элемента за время, пропорциональное высоте. АВЛ-дерево: верхняя оценка O(log n) на высоту, сохранение свойства при помощи малых и больших вращений. Splayдерево: дерево, которое остаётся сбалансированным в среднем и при этом не хранит никакой дополнительной информации в вершинах; реализация основных операций через операцию splay; верхняя оценка O(log n) на среднюю стоимость операций. Декартово дерево: построение декартова дерева за линейное время (при предварительно отсортированных ключах), реализация операций вставки и удаления через split и merge. Использование неявного ключа, rope. Дерево отрезков, двоичный подъём. Динамическая задача RMQ/RSQ: оценка O(log n) для запросов суммы/минимума и изменения элемента. Статический вариант задачи RMQ: полное предвычисление (O(n^2) на предобработку, O(1) на запрос), метод разреженной таблицы (O(n log n) на предобработку, O(1) на запрос). Задача LCA: эйлеров обход дерева, сведение к задаче RMQ. Сведение задачи RMQ к задаче LCA через декартово дерево.
Числовые алгоритмы	Элементарная арифметика: сложение, умножение, деление. Арифметика сравнений: сложение, умножение, возведение в степень, алгоритм Евклида, расширенный алгоритм Евклида, деление. Проверка чисел на простоту. Генерация случайных простых чисел. Криптография: схемы с закрытым ключом, RSA, цифровая подпись.
Быстрое преобразование Фурье	Быстрое вычисление значений многочлена в точках: два способа задания многочленов — коэффициентами и значениями в точках; вычисление значений многочлена в точках

	методом «разделяй и властвуй»; дискретное преобразование Фурье; быстрое преобразование Фурье. Интерполяция: интерполяция в терминах матриц; матрица Вандермонда; интерполяция как домножение на обратную матрицу.
Строковые алгоритмы	Поиск подстроки в строке. Наивный алгоритм, алгоритм Карпа-Рабина. Z-функция. Алгоритм Кнута-Морриса-Пратта. Множественный поиск, алгоритм Ахо-Корасик. Поиск с ошибками: поиск образца с одной ошибкой, алгоритмы на основе Ахо-Корасик и БПФ. Суффиксное дерево, построение за линейное время. Суффиксный массив, построение за O(n log n)
NP-полные задачи	Задача пропозициональной выполнимости: задача поиска; алгоритм, задающий задачу поиска. Задача о коммивояжере: оптимизационные задачи и задачи поиска; сравнение с задачей о нахождении покрывающего дерева. Сложные и простые задачи: сравнение нескольких пар задач, которые формулируются похоже, но имеют разную сложностью. Подходы к решению NP-трудных задач: локальный поиск, эвристики, приближённые алгоритмы.

6. Образовательные результаты освоения дисциплины (модуля):

Код и наименование компетенции	**Код и наименование индикатора достижения компетенции**
ПК-2. Способен разрабатывать и внедрять программные продукты для объектов критической информационной инфраструктуры	ИПК-2.1. Знает основы синтаксиса высокоуровневых и низкоуровневых языков программирования, а также знаком с паттернами разработки программного обеспечения как для микроэлектроники, так и для микросервисных приложений
	ИПК-2.2. Умеет разрабатывать программные продукты с использованием различных языков программирования и внедрять их в объекты критической инфраструктуры на аппаратном и программном уровнях

Код и наименование компетенции	Код и наименование индикатора достижения компетенции
	ИПК-2.3. Владеет методами создания, внедрения, тестирования и обеспечения безопасности программного обеспечения на всех этапах жизненного цикла продукта, в том числе с применением технологий искусственного интеллекта, а также способен разработать программное обеспечение как для микроэлектроники, так и для микросервисных и монолитных систем
ПК-6. Способен проектировать и реализовывать объекты критической информационной инфраструктуры для различных операционных систем и облачных платформ	ИПК-6.1. Знает принципы проектирования информационных систем и объектов критической инфраструктуры, особенности различных операционных систем и облачных платформ
	ИПК-6.2. Умеет реализовывать объекты критической инфраструктуры для различных операционных систем и облачных платформ, учитывая их специфику
	ИПК-6.3. Владеет методами проектирования и реализации объектов критической инфраструктуры, включая установку, настройку и обеспечение их работоспособности с учетом специфики работы в различных операционных системах и облачных платформах
ПК-7. Способен разрабатывать и применять решения на основе математических моделей для обеспечения информационной безопасности	ИПК-7.1. Знает принципы построения математических моделей для обеспечения информационной безопасности
	ИПК-7.2. Умеет разрабатывать и внедрять математические модели для обеспечения информационной безопасности
	ИПК-7.3. Владеет методами сбора, анализа и использования данных для математического моделирования в составе решений, обеспечивающих информационную безопасность в информационных системах и объектах критической информационной инфраструктуры