

**Краснодонский факультет инженерии и менеджмента (филиал)
Кафедра информационных технологий и транспорта**



(подпись)

«14» марта 2025 года

(наименование учебной дисциплины, практики)

(код и наименование направления подготовки (специальности))

наименование профиля подготовки (специальности, магистерской программы); при отсутствии ставится прочерк)

(ПОДПИСЬ)

(подпись)

Заведующий кафедрой
информационных
технологий и транспорта

(подпись)

Верительник Е.А

Краснодон 2025

**Комплект оценочных материалов по дисциплине
«Структуры и алгоритмы компьютерной обработки данных»**

Задания закрытого типа

Задания закрытого типа на выбор правильного ответа

Выберите один правильный ответ.

1. Какой алгоритм сортировки имеет гарантированную временную сложность $O(n \log n)$ в худшем случае?

- А) Сортировка пузырьком.
- Б) Сортировка вставками.
- В) Быстрая сортировка.
- Г) Сортировка слиянием.

Правильный ответ: Г

Компетенции (индикаторы): ОПК-8 (ОПК-8.1)

2. Когда бинарный поиск является наиболее эффективным?

- А) Когда данные не отсортированы.
- Б) Когда данные отсортированы.
- В) Когда нужно найти первый элемент в списке.
- Г) Когда нужно найти последний элемент в списке.

Правильный ответ: Б

Компетенции (индикаторы): ОПК-8 (ОПК-8.1)

3. Какой тип данных в Python лучше всего подходит для реализации стека?

- А) Словарь (Dictionary).
- Б) Множество (Set).
- В) Список (List).
- Г) Кортеж (Tuple).

Правильный ответ: В

Компетенции (индикаторы): ОПК-8 (ОПК-8.1)

4. Какая структура данных является иерархической и состоит из узлов, связанных между собой?

- А) Связный список (Linked List).
- Б) Дерево (Tree).
- В) Очередь (Queue).
- Г) Хеш-таблица (Hash Table).

Правильный ответ: Б

Компетенции (индикаторы): ОПК-8 (ОПК-8.1)

Задания закрытого типа на установление соответствия

Установите правильное соответствие.

Каждому элементу левого столбца соответствует только один элемент правого столбца.

1. Алгоритмы поиска и их характеристика. Установите соответствие типу алгоритма поиска и его характеристике. Каждому элементу левого столбца соответствует только один элемент правого столбца:

Алгоритмы поиска		Характеристика	
1)	Линейный поиск	А)	Работает только с отсортированными данными.
2)	Бинарный поиск	Б)	Проверяет каждый элемент последовательно.
3)	Хеш-поиск (в словаре)	В)	Обеспечивает в среднем $O(1)$ время поиска.

Правильный ответ: 1Б, 2А, 3В

Компетенции (индикаторы): ОПК-8 (ОПК-8.1)

2. Алгоритмы сортировки и их характеристика. Установите соответствие типу алгоритма сортировки и его характеристике. Каждому элементу левого столбца соответствует только один элемент правого столбца:

Алгоритм сортировки		Пример	
1)	Сортировка пузырьком	А)	Эффективна для почти отсортированных данных.
2)	Сортировка вставками	Б)	Разделяет массив на подмассивы, сортирует их и объединяет.
3)	Сортировка слиянием	В)	Сравнивает и меняет местами соседние элементы.

Правильный ответ: 1В, 2А, 3Б

Компетенции (индикаторы): ОПК-8 (ОПК-8.1)

3. Абстрактные типы данных их описание. Установите соответствие абстрактному типу данных и его содержанию. Каждому элементу левого столбца соответствует только один элемент правого столбца:

Тип		Описание	
1)	Стек	А)	Элементы добавляются и удаляются с обоих концов.
2)	Очередь	Б)	Элементы добавляются в конец и удаляются с начала (FIFO).

3. Сортировка массива с помощью алгоритма сортировки вставками. Запишите правильную последовательность букв слева направо:

А) Выбор следующего элемента для вставки.

Б) Сравнение выбранного элемента с элементами в отсортированной части массива.

В) Сдвиг элементов отсортированной части массива, чтобы освободить место для вставляемого элемента.

Г) Вставка элемента в правильное место в отсортированной части массива.

Правильный ответ: А, Б, В, Г

Компетенции (индикаторы): ОПК-8 (ОПК-8.1)

4. Обход дерева в глубину. Запишите правильную последовательность букв слева направо:

А) Посетить текущий узел.

Б) Выбрать один из непосещенных дочерних узлов.

В) Рекурсивно выполнить обход в глубину для выбранного дочернего узла.

Г) Если нет непосещенных дочерних узлов, вернуться к предыдущему узлу.

Правильный ответ: А, Б, В, Г

Компетенции (индикаторы): ОПК-8 (ОПК-8.1)

Задания открытого типа

Задания открытого типа на дополнение

Напишите пропущенное слово (словосочетание).

1. Списки в Python являются _____, что означает возможность изменения их содержимого после создания.

Правильный ответ: Изменяемыми.

Компетенции (индикаторы): ОПК-8 (ОПК-8.1)

2. Бинарный поиск требует, чтобы входной список был предварительно _____, чтобы корректно работать.

Правильный ответ: Отсортирован.

Компетенции (индикаторы): ОПК-8 (ОПК-8.1)

3. При возникновении _____ в хеш-таблице, несколько ключей отображаются в один и тот же индекс.

Правильный ответ: Коллизий.

Компетенции (индикаторы): ОПК-8 (ОПК-8.1)

4. Алгоритм _____ сортировки делит список на подсписки, рекурсивно сортирует их, а затем объединяет отсортированные подсписки.

Правильный ответ: Слиянием.

Компетенции (индикаторы): ОПК-8 (ОПК-8.1)

Задания открытого типа с кратким свободным ответом

Дайте ответ на вопрос.

1. Как называется алгоритм поиска, который последовательно проверяет каждый элемент в массиве до тех пор, пока не найдет нужный?

Правильный ответ: Линейный поиск / Последовательный поиск / Простой поиск.

Компетенции (индикаторы): ОПК-8 (ОПК-8.1)

2. Как называется алгоритм сортировки, который сравнивает соседние элементы и меняет их местами, если они находятся в неправильном порядке?

Правильный ответ: Сортировка пузырьком / Пузырьковая сортировка / Обменная сортировка.

Компетенции (индикаторы): ОПК-8 (ОПК-8.1)

3. Как называется операция добавления элемента в бинарное дерево поиска?

Правильный ответ: Вставка / Добавление / Добавить.

Компетенции (индикаторы): ОПК-8 (ОПК-8.1)

4. Как называется способ разрешения коллизий в хеш-таблице, при котором при возникновении коллизии ключ помещается в следующую свободную ячейку?

Правильный ответ: Открытая адресация / Открытое хеширование.

Компетенции (индикаторы): ОПК-8 (ОПК-8.1)

Задания открытого типа с развернутым ответом

Дайте развернутый ответ на вопрос.

1. Объясните, что такое временная сложность алгоритма и как она оценивается с использованием О-нотации. Приведите примеры алгоритмов с различной временной сложностью ($O(1)$, $O(\log n)$, $O(n)$, $O(n \log n)$, $O(n^2)$)?

Время выполнения – 20 мин.

Ожидаемый результат:

Временная сложность алгоритма описывает, как время выполнения алгоритма растет с увеличением размера входных данных. Она оценивается с использованием О-нотации, которая выражает верхнюю границу роста времени выполнения.

1. $O(1)$ (Константное время). Время выполнения не зависит от размера входных данных (например, доступ к элементу по индексу в списке).

2. $O(\log n)$ (Логарифмическое время). Время выполнения растет логарифмически с размером входных данных (например, бинарный поиск).
3. $O(n)$ (Линейное время): Время выполнения растет линейно с размером входных данных (например, линейный поиск).
4. $O(n \log n)$ (Линейно-логарифмическое время). Время выполнения растет как n , умноженное на логарифм n (например, сортировка слиянием, быстрая сортировка в среднем случае).
5. $O(n^2)$ (Квадратичное время). Время выполнения растет как квадрат размера входных данных (например, сортировка пузырьком, сортировка вставками).

Критерии оценивания: Наличие в ответе четырех типов сложности.

Компетенции (индикаторы): ОПК-8 (ОПК-8.1)

2. Объясните, что такое хеш-таблица и как она работает. Опишите основные методы разрешения коллизий?

Время выполнения – 10 мин.

Ожидаемый результат:

Хеш-таблица (hash table) – это структура данных, которая реализует ассоциативный массив (отображение ключ-значение). Она использует хеш-функцию для преобразования ключа в индекс (хеш-код), который указывает на место хранения соответствующего значения в массиве (таблице).

Коллизии возникают, когда два или более ключа имеют одинаковый хеш-код.

Методы разрешения коллизий.

1. Метод цепочек. Каждый элемент массива содержит связный список элементов с одинаковым хеш-кодом.

2. Открытая адресация. При возникновении коллизии ищется другая свободная ячейка в массиве (например, линейное пробирование, квадратичное пробирование, двойное хеширование).

Критерии оценивания: Наличие в ответе определения хэш-таблицы и методов разрешения коллизий, описанных минимум одним предложением.

Компетенции (индикаторы): ОПК-8 (ОПК-8.1)

3. Опишите, что такое дерево поиска и как оно используется для эффективного поиска данных. Объясните, как работает алгоритм вставки элемента в бинарное дерево поиска?

Время выполнения – 15 мин.

Ожидаемый результат:

Дерево поиска – это древовидная структура данных, предназначенная для быстрого поиска, вставки и удаления элементов. В бинарном дереве поиска каждый узел имеет не более двух потомков (левый и правый), и для каждого узла выполняется следующее условие: значения всех узлов в левом поддереве меньше значения текущего узла, а значения всех узлов в правом поддереве больше значения текущего узла.

Вставка элемента.

1. Начать с корневого узла.
2. Сравнить значение вставляемого элемента со значением текущего узла.
3. Если значение меньше, перейти в левое поддерево.
4. Если значение больше, перейти в правое поддерево.
5. Повторять шаги 2-4, пока не будет найден пустой узел, в который можно вставить новый элемент.

Критерии оценивания: Ответ должен содержать краткое определение дерева поиска и описание вставки элемента.

Компетенции (индикаторы): ОПК-8 (ОПК-8.1)

4. Опишите, что такое рекурсия и объясните, какие существуют ограничения на использование рекурсии.

Время выполнения – 10 мин.

Ожидаемый результат:

Рекурсия – это способ определения функции через саму себя. Рекурсивная функция вызывает саму себя для решения подзадач, пока не будет достигнут базовый случай, который можно решить напрямую.

Глубина рекурсии. Python имеет ограничение на глубину рекурсивных вызовов (обычно 1000). Превышение лимита вызывает RecursionError.

Переполнение стека. Каждый рекурсивный вызов занимает место в стеке вызовов, который может переполниться. Для больших задач часто предпочтительнее использовать итеративные решения.

Критерии оценивания: Наличие в ответе определения рекурсии, глубины и переполнения рекурсии.

Компетенции (индикаторы): ОПК-8 (ОПК-8.1)

Экспертное заключение

Представленный фонд оценочных средств (далее – ФОС) по дисциплине «Структуры и алгоритмы компьютерной обработки данных» соответствует требованиям ФГОС ВО.

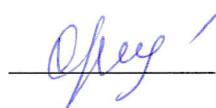
Предлагаемые формы и средства текущего и промежуточного контроля адекватны целям и задачам реализации основной профессиональной образовательной программы по направлению подготовки 09.03.01 Информатика и вычислительная техника.

Оценочные средства для текущего контроля успеваемости, промежуточной аттестации по итогам освоения дисциплины представлены в полном объеме.

Виды оценочных средств, включенные в представленный фонд, отвечают основным принципам формирования ФОС.

Разработанный и представленный для экспертизы фонд оценочных средств рекомендуется к использованию в процессе подготовки обучающихся по указанному направлению 09.03.01 Информатика и вычислительная техника.

Председатель учебно-методической
комиссии Краснодонского факультета
инженерии и менеджмента (филиала)

 Родионова О.Ю.

Лист изменений и дополнений

№ п/п	Виды дополнений и изменений	Дата и номер протокола заседания кафедры (кафедр), на котором были рассмотрены и одобрены изменения и дополнения	Подпись (с расшифровкой) заведующего кафедрой (заведующих кафедрами)