

Вопрос 1

Отметить вопрос

Пока нет ответа

Балл: 1,00

Какие утверждения верны для бинарной пирамиды (max-heap)?

Выберите один или несколько ответов:

- ☒ a. Каждый узел имеет значение не меньше значений его потомков (для max-heap)
- ☒ b. Может быть представлена в виде массива
- ☐ c. Гарантирует полную упорядоченность всех элементов
- ☒ d. Является полным бинарным деревом
- ☒ e. Позволяет эффективно находить максимальный (минимальный) элемент

Вопрос 2

Отметить вопрос

Пока нет ответа

Балл: 1,00

Какие особенности имеет алгоритм пирамидальной сортировки?

Выберите один или несколько ответов:

- ☒ a. Не требует дополнительной памяти (сортировка на месте)
- ☒ b. Имеет гарантированную сложность $O(n \log n)$
- ☒ c. Не является устойчивой сортировкой
- ☒ d. Требуется случайного доступа к данным по индексу
- ☒ e. Менее эффективна на частично отсортированных данных чем быстрая сортировка
- ☒ f. Плохо использует кэш-память процессора из-за прыжков по индексам

Вопрос 3

[Отметить вопрос](#)

Пока нет ответа

Балл: 1,00

Где применяется структура данных "пирамида"?

Выберите один или несколько ответов:

- ☒ a. Реализация очереди с приоритетом
- ☒ b. Алгоритм Дейкстры для поиска кратчайших путей
- ☒ c. Алгоритм Прима для построения минимального остовного дерева
- ☐ d. Хеш-таблицы для разрешения коллизий
- ☒ e. Алгоритм поиска k-го порядкового элемента
- ☒ f. Планировщики задач в операционных системах

Вопрос 4

[Отметить вопрос](#)

Пока нет ответа

Балл: 1,00

Какие утверждения верны для минимальной и максимальной пирамид?

Выберите один или несколько ответов:

- ☒ a. В max-heap корень содержит максимальный элемент
- ☒ b. В min-heap корень содержит минимальный элемент
- ☒ c. Алгоритмы операций для min-heap и max-heap симметричны
- ☐ d. Min-heap всегда строится быстрее чем max-heap
- ☒ e. Для сортировки по возрастанию можно использовать max-heap

[Предыдущая страница](#)[Следующая страница](#)

Вопрос 5

[Отметить вопрос](#)

Пока нет ответа

Балл: 1,00

Как пирамида используется для реализации приоритетной очереди?

Выберите один или несколько ответов:

- ☒ а. Операция вставки соответствует добавлению элемента в пирамиду
- ☒ б. Извлечение элемента с наивысшим приоритетом соответствует извлечению корня
- ☒ в. Просмотр элемента с наивысшим приоритетом соответствует обращению к корню
- ☐ г. Требуется дополнительная память $O(n \log n)$
- ☒ д. Обеспечивает логарифмическое время для основных операций

[Предыдущая страница](#)[Следующая страница](#)

Вопрос 6

[Отметить вопрос](#)

Пока нет ответа

Балл: 1,00

В каких случаях выбирать нисходящее или восходящее построение пирамиды?

Выберите один или несколько ответов:

- ☒ а. Нисходящее: когда данные поступают потоком или по одному
- ☒ б. Восходящее: когда весь массив данных доступен сразу
- ☒ в. Восходящее: когда важна эффективность начального построения
- ☐ г. Нисходящее всегда предпочтительнее восходящего
- ☒ д. Нисходящее: для инкрементального обновления пирамиды
- ☒ е. Восходящее: для реализации пирамидальной сортировки

[Предыдущая страница](#)[Следующая страница](#)

Вопрос 7

[Отметить вопрос](#)

Пока нет ответа

Балл: 1,00

Какие этапы включает алгоритм пирамидальной сортировки?

Выберите один или несколько ответов:

- ☒ a. Построение пирамиды из исходного массива
- ☒ b. Последовательное извлечение элементов из пирамиды
- ☒ c. Перемещение корня в конец массива и уменьшение размера пирамиды
- ☐ d. Предварительная сортировка части массива
- ☒ e. Восстановление свойства пирамиды после каждого извлечения

[Предыдущая страница](#)[Следующая страница](#)

Вопрос 8

[Отметить вопрос](#)

Пока нет ответа

Балл: 1,00

Что важно учитывать при реализации операций с пирамидой?

Выберите один или несколько ответов:

- ☒ a. Корректное вычисление индексов родителя и потомков
- ☒ b. Обработку граничных случаев (пустая пирамида, один элемент)
- ☒ c. Поддержку свойства пирамиды после каждой модификации
- ☐ d. Использование рекурсии для всех операций
- ☒ e. Эффективное восстановление пирамиды после извлечения корня
- ☒ f. Учет типа пирамиды (min-heap или max-heap) в сравнениях

[Предыдущая страница](#)[Следующая страница](#)

Вопрос 9

[Отметить вопрос](#)

Пока нет ответа

Балл: 1,00

Какие теоретические основы лежат в основе структуры данных "пирамида"?

Выберите один или несколько ответов:

- ☒ a. Свойство полного бинарного дерева
- ☒ b. Свойство упорядоченности между родителями и потомками
- ☒ c. Представление дерева в виде массива
- ☐ d. Теория графов для представления связей
- ☒ e. Логарифмическая высота полного бинарного дерева с n узлами

[Предыдущая страница](#)[Следующая страница](#)

Вопрос 10

Пока нет ответа

Балл: 1,00

Как пирамида сравнивается с другими структурами данных?

Выберите один или несколько ответов:

- ☒ a. Быстрее чем сбалансированные деревья для операций вставки/извлечения корня
- ☒ b. Использует меньше памяти чем деревья с указателями
- ☒ c. Менее гибкая чем сбалансированные деревья поиска
- ☐ d. Всегда эффективнее сортировочных деревьев
- ☒ e. Не поддерживает эффективный поиск произвольного элемента
- ☒ f. Идеальна для реализации очереди с приоритетом

[Предыдущая страница](#)[Следующая страница](#)

Вопрос 11

[Отметить вопрос](#)

Пока нет ответа

Балл: 1,00

Какие существуют модификации и расширения бинарной пирамиды?

Выберите один или несколько ответов:

- ☒ a. d-ичные пирамиды (d-ary heaps)
- ☒ b. Биномиальные пирамиды
- ☒ c. Фибоначчиевы пирамиды
- ☐ d. Все пирамиды являются бинарными
- ☒ e. Пирамиды с поддержкой слияния (mergeable heaps)

[Предыдущая страница](#)[Следующая страница](#)

Вопрос 12

[Отметить вопрос](#)

Пока нет ответа

Балл: 1,00

Какие утверждения верно описывают нисходящее построение пирамиды?

Выберите один или несколько ответов:

- ☒ a. Элементы добавляются по одному в конец массива
- ☒ b. После добавления элемента выполняется процедура "просеивания вверх" (sift-up)
- ☒ c. Имеет временную сложность $O(n \log n)$ для построения из n элементов
- ☐ d. Требуется предварительной сортировки элементов
- ☒ e. Начинается с пустой пирамиды
- ☒ f. Подходит для инкрементального построения при потоковом поступлении данных

[Предыдущая страница](#)[Следующая страница](#)

Вопрос 13

[Отметить вопрос](#)

Пока нет ответа

Балл: 1,00

Какие практические рекомендации можно дать по применению пирамиды?

Выберите один или несколько ответов:

- ☒ a. Использовать для реализации очередей с приоритетом
- ☒ b. Применять пирамидальную сортировку когда нужна гарантированная $O(n \log n)$
- ☒ c. Выбирать восходящее построение для начальной инициализации
- ☐ d. Всегда использовать рекурсивные реализации
- ☒ e. Учитывать плохую локальность ссылок при работе с большими пирамидами

[Предыдущая страница](#)[Следующая страница](#)

Вопрос 14

[Отметить вопрос](#)

Пока нет ответа

Балл: 1,00

Какие утверждения верно описывают восходящее построение пирамиды?

Выберите один или несколько ответов:

- ☒ a. Начинается с произвольного массива элементов
- ☒ b. Выполняется процедура "просеивания вниз" (sift-down) для всех нелистовых узлов
- ☒ c. Имеет линейную временную сложность $O(n)$ для построения из n элементов
- ☐ d. Требуем дополнительной памяти $O(n \log n)$
- ☒ e. Обрабатывает узлы в порядке от последнего родителя к корню
- ☒ f. Более эффективно для построения пирамиды из всего массива сразу

Вопрос 15

[Отметить вопрос](#)

Пока нет ответа

Балл: 1,00

Какие утверждения верно описывают алгоритм пирамидальной сортировки?

Выберите один или несколько ответов:

- ☒ a. Имеет временную сложность $O(n \log n)$ в худшем, среднем и лучшем случаях
- ☒ b. Является алгоритмом сортировки на месте (in-place)
- ☐ c. Является устойчивой (stable) сортировкой
- ☒ d. Состоит из двух основных этапов: построение пирамиды и извлечение элементов
- ☒ e. Использует структуру данных "пирамида" для управления элементами

[Предыдущая страница](#)[Следующая страница](#)

Вопрос 16

[Отметить вопрос](#)

Пока нет ответа

Балл: 1,00

В чем состоят основные различия между нисходящим и восходящим построением пирамиды?

Выберите один или несколько ответов:

- ☒ a. Нисходящее имеет сложность $O(n \log n)$, восходящее — $O(n)$
- ☒ b. Нисходящее подходит для потоковых данных, восходящее — для полного массива
- ☒ c. Нисходящее использует sift-up, восходящее — sift-down
- ☐ d. Только один метод гарантирует корректность построения пирамиды
- ☒ e. Нисходящее начинается с пустой пирамиды, восходящее — с произвольного массива
- ☒ f. Восходящее обычно более эффективно для начального построения

Вопрос 17

[Отметить вопрос](#)[Пока нет ответа](#)

Балл: 1,00

Какие основные операции поддерживает структура данных "пирамида"?

Выберите один или несколько ответов:

- ☒ a. Вставка нового элемента
- ☒ b. Извлечение максимального (минимального) элемента
- ☒ c. Построение пирамиды из произвольного массива
- ☐ d. Поиск произвольного элемента за $O(1)$
- ☒ e. Изменение приоритета элемента
- ☒ f. Удаление произвольного элемента

[Предыдущая страница](#)[Следующая страница](#)

Вопрос 18

[Отметить вопрос](#)[Пока нет ответа](#)

Балл: 1,00

Какие утверждения о временной сложности операций с пирамидой верны?

Выберите один или несколько ответов:

- ☒ a. Вставка элемента имеет сложность $O(\log n)$
- ☒ b. Извлечение максимума имеет сложность $O(\log n)$
- ☒ c. Получение максимума без извлечения имеет сложность $O(1)$
- ☐ d. Поиск произвольного элемента имеет сложность $O(1)$
- ☒ e. Восходящее построение из n элементов имеет сложность $O(n)$

[Предыдущая страница](#)[Следующая страница](#)

Вопрос 19

[Отметить вопрос](#)[Пока нет ответа](#)[Балл: 1,00](#)

Какие утверждения о представлении пирамиды в памяти верны?

Выберите один или несколько ответов:

- ☒ a. Пирамида может быть представлена массивом
- ☒ b. Для элемента с индексом i потомки имеют индексы $2i+1$ и $2i+2$
- ☒ c. Для элемента с индексом i родитель имеет индекс $\lfloor (i-1)/2 \rfloor$
- ☐ d. Требуются дополнительные указатели для связи между узлами
- ☒ e. Представление в виде массива эффективно использует память
- ☒ f. Массив представляет полное бинарное дерево

[Предыдущая страница](#)[Следующая страница](#)

Вопрос 20

[Пока нет ответа](#)[Балл: 1,00](#)

Какие утверждения верно описывают операции просеивания?

Выберите один или несколько ответов:

- ☒ a. Sift-up восстанавливает свойство пирамиды при вставке элемента
- ☒ b. Sift-down восстанавливает свойство пирамиды при извлечении корня
- ☒ c. Sift-up перемещает элемент вверх по дереву, пока не выполнится свойство пирамиды
- ☐ d. Sift-up и sift-down имеют одинаковую вычислительную сложность $O(1)$
- ☒ e. Sift-down сравнивает элемент с его потомками и меняет с большим (меньшим)
- ☒ f. Обе операции имеют сложность $O(\log n)$ в худшем случае

[Предыдущая страница](#)[Закончить попытку...](#)