

Вопросы на 4 к экзамену по алгоритмам SPb HSE, 1-й курс ПМИ, октябрь 2024

База, асимптотики

- (b) 1. База. Алгоритм Карацубы для умножения многочленов: идея, время работы.
- (b) 2. База. Доказательство мастер-теоремы.
- (b) 3. База. Доказательства по индукции. Пример $T(n) = \max_{x=1..n-1} (T(x) + T(n-x) + x(n-x))$.
- (b) 4. База. Доказательства по индукции. *Пример из дз:* $T(n) = n + T(\frac{1}{2}n) + T(\frac{1}{3}n)$.
- (b) 5. База. Оценка суммы гармонического ряда через куски длины 2^k .
- (b) 6. База. Список делителей для всех чисел от 1 до n за $\mathcal{O}(n \log n)$.
- (b) 7. База. Неасимптотические оптимизации. Примеры 6 проблем.

Структуры данных

- (b) 8. DS. Список односвязный, реализация на массиве.
- (b) 9. DS. Вектор. Доказательство времени работы.
- (b) 10. DS. Стек, очередь, дек. Интерфейс. *Средства языка C++ и их сравнение.*
- (b) 11. DS. Очередь на односвязном списке.
- (b) 12. DS. Очередь и дек на циклическом массиве.
- (b) 13. DS. Стек с минимумом. Очередь с `min` через два стека.
- (b) 14. База. Связь амортизированного времени и среднего. Произвольный случай с $\Delta\varphi = \varphi_{end} - \varphi_0$.
- (b) 15. База. Примеры на потенциалы: `push/pop(k)`, хороший и плохой потенциал для $x^2 + y^2 = n$
- (b) 16. DS. Разбор арифметического выражения со стеком за линейное время.
- (b) 17. DS. Бинпоиск вещественный: для монотонной функции, для корня многочлена 3-й степени.
- (b) 18. DS. Бинпоиск: определение границ L, R бинпоиска для корня многочлена 3-й степени.
- (b) 19. DS. *Два указателя. Максимальный отрезок, без повторяющихся чисел.*
- (b) 20. DS. Хеш-таблица. Версия с открытой адресацией.
- (b) 21. DS. *События. Покраска отрезков массива за $\mathcal{O}(n + q \log q)$.*
- (b) 22. DS. Вектор: избавление от амортизации. Любой способ.
- (b) 23. Heap. Задача `a[i]=x, getMin`.
- (b) 24. Heap. `decreaseKey, delete`.
- (b) 25. DS. Аллокация памяти. *Переопределение `new, delete` на стековой.*
- (b) 26. DS. Аллокация памяти. Список.
- (b) 27. DS. `ExtractMin` \rightarrow `Del` для кучи.
- (b) 28. DS. `Add` \rightarrow `Merge`. Оценка времени работы
- (b) 29. DS. Пополняемый массив. `Build` \rightarrow `Add`. Корневая.
- (b) 30. DS. `Build, Add` \rightarrow `Del`.

Сортировки

- (b) 31. Sort. `Insertion(+BS)`, время работы, число сравнений.
- (b) 32. Sort. Подробный анализ/сравнение сортировок за квадрат: число сравнений, присваиваний, стабильность, дошпамять.
- (b) 33. Sort. `MergeSort`: рекурсивная версия.

- (b) 34. Sort. *MergeSort*: подсчёт числа инверсий.
- (b) 35. Sort. *QuickSort*. Доказательство через дерево рекурсии и вероятность «сравнить 2 элемента».
- (b) 36. Sort. Порядковая статистика за $\mathcal{O}(n)$, детерминированный алгоритм. Алгоритм.
- (b) 37. Sort. *CountSort* для пары (ключ сортировки, полезные данные).
- (b) 38. Sort. *Bucket Sort*. Две версии алгоритма. Время работы (формулировка).

Кучи

- (b) 39. Heap. Van Emde Boas trees. `add` за $\mathcal{O}(\log \log C)$.
- (b) 40. Heap. Van Emde Boas trees. `extractMin` за $\mathcal{O}(\log \log C)$.
- (b) 41. Heap. MinMax heap. Алгоритм `siftUp`, `siftDown`.
- (b) 42. Heap. Leftist heap. Обоснование времени работы.
- (b) 43. Heap. Skew Heap: операции `add`, `extractMin`.
- (b) 44. Heap. Лёгкие и тяжёлые рёбра, доказательство времени работы Skew Heap.
- (b) 45. Heap. *d-куча*.
- (b) 46. Heap. Турнирные деревья. Операции `change(i, x)` и `extractMin` за $\mathcal{O}(\log n)$.
- (b) 47. Heap. Список турнирный деревьев: `add`, `min`, `merge` за $\mathcal{O}(1)$, `decKey` и `extractMin` за $\mathcal{O}(\log n)$.
- (b) 48. Inplace. *Rotate* за $\mathcal{O}(n)$. Средства языка C++.
- (b) 49. D&C. Поиск двух ближайших точек.