

АиСД. Зачет. Попытка #1. Вариант 1.

1. Приведите пример двух функций, для которых будет выполняться отношение $O('O'$ большое), но не будет выполняться отношение $o('o'$ маленькое).
2. Чему равны размерность задачи, трудоемкость и емкостная сложность для алгоритма **шейкерной сортировки в лучшем случае**. Дать ответ, используя аннотацию $O('O'$ большое).
3. Составьте рекуррентное уравнение для оценки **трудоемкости** алгоритма:

```
function  algoritm(massive, left, right){
    middle = 2(left + right)/3;
    if(left == right)
        return  massive[left];
    else{
        v = algoritm(massive, middle + 1, right);
        return  max(u, v);
    }
}
```

4. Расположите следующие функции по порядку в соответствии с нотацией $O('O'$ большое), объясните результат:

$$4^{\lg n}, \quad \lg^4 n, \quad n \lg n.$$

5. Определите нижнюю асимптотическую границу функции $T(n)$ для рекуррентного уравнения:

$$T(n) = 2T(n/8) + n^3/2, T(1) = c > 0.$$

6. Приведите пример алгоритма сортировки, который имеет сложность в лучшем случае $O(n^2)$.
7. Решить рекуррентное уравнение: $T(n) = T(n/4) + T(3n/4) + n^2$.
8. Пошагово продемонстрируйте работу алгоритма **быстрой сортировки по убыванию** для следующего массива чисел:

4 15 1 5 10 7 4

9. Проведите начальное распределение последовательности элементов по файлам для **сортировки каскадным слиянием при N=6** (серии состоят из 1 эл-та):

28 42 33 20 12 14 29 19 21 1 49 14 16 37 4 13 25

10. Опишите словесно алгоритм **сортировки слиянием**.

Задания 12-17 выполняются с деревом из задания 11.

11. Пошагово (с учетом поворотов) сформировать **АВЛ-дерево** из следующих чисел:

16 13 12 7 6 1 18 11 4 5

12. Выполнить **обратный обход** полученного дерева.

13. Рассчитать глубину для вершин с четным (по значению) ключом.

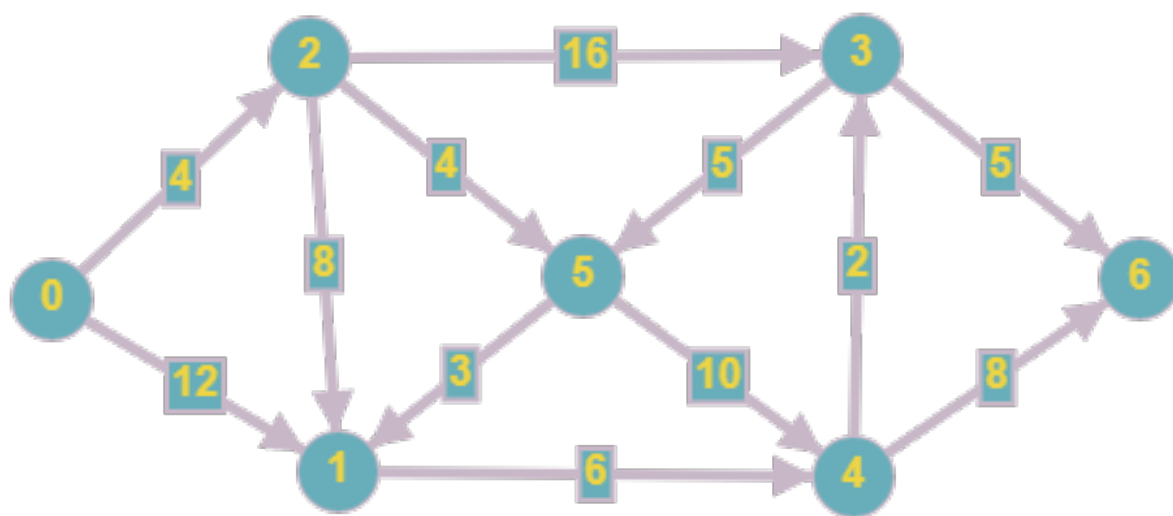
14. Удалить **корень** дерева.

15. Выполнить **симметричный обход** полученного дерева.

16. Из полученного списка чисел сформировать пошагово **кучу Фибоначчи**.

17. Выполнить 3 операции **удаления элемента** с максимальным приоритетом.

Задан ориентированный граф:



18. Составить матрицу смежности.

19-20. Построить минимальное остовое дерево. Пошагово продемонстрировать алгоритм.