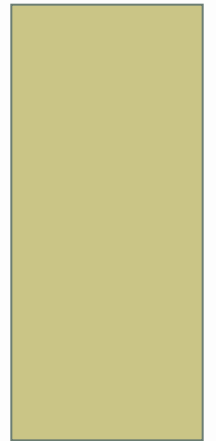
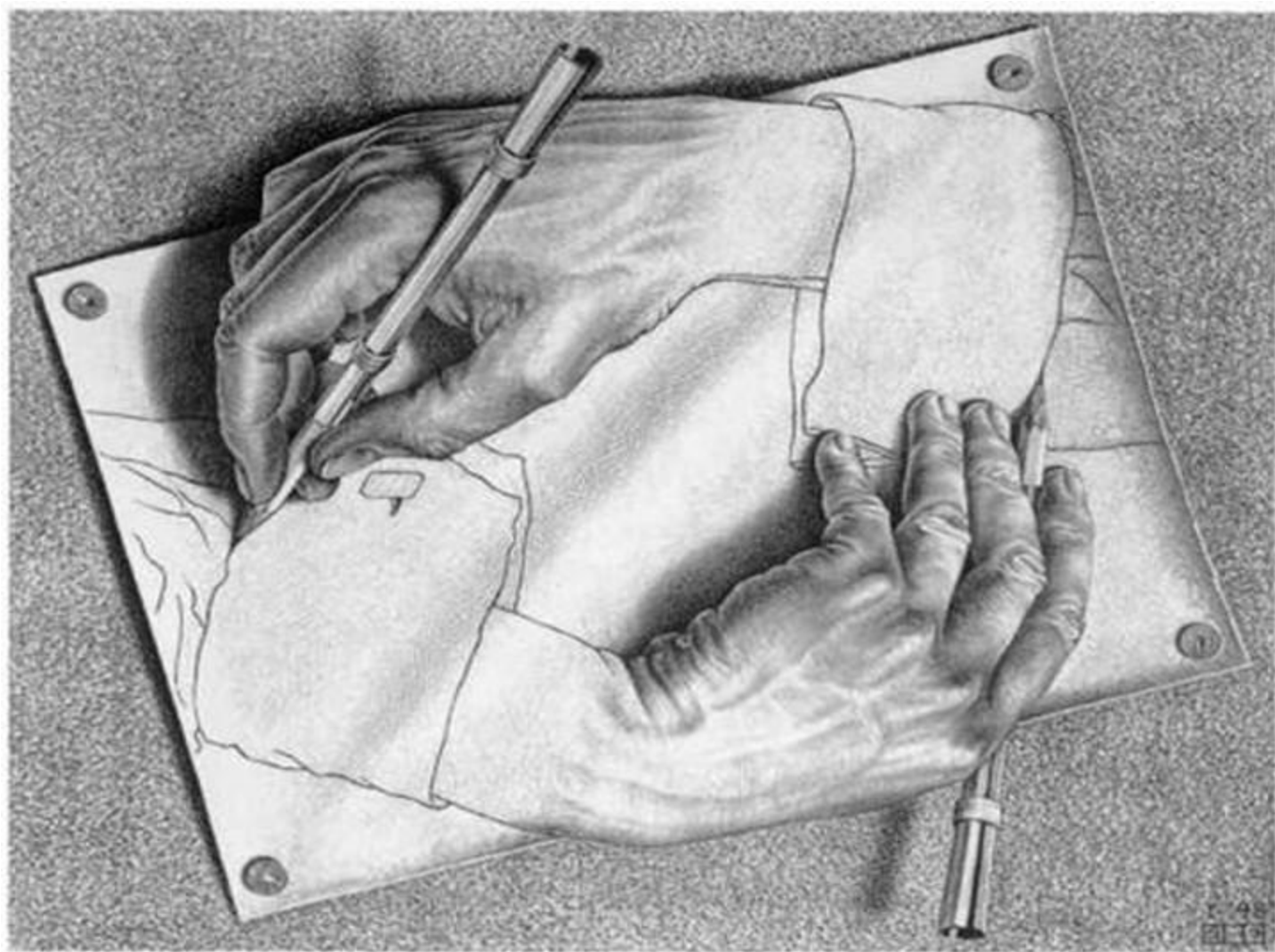


ЛЕКЦІЯ 8

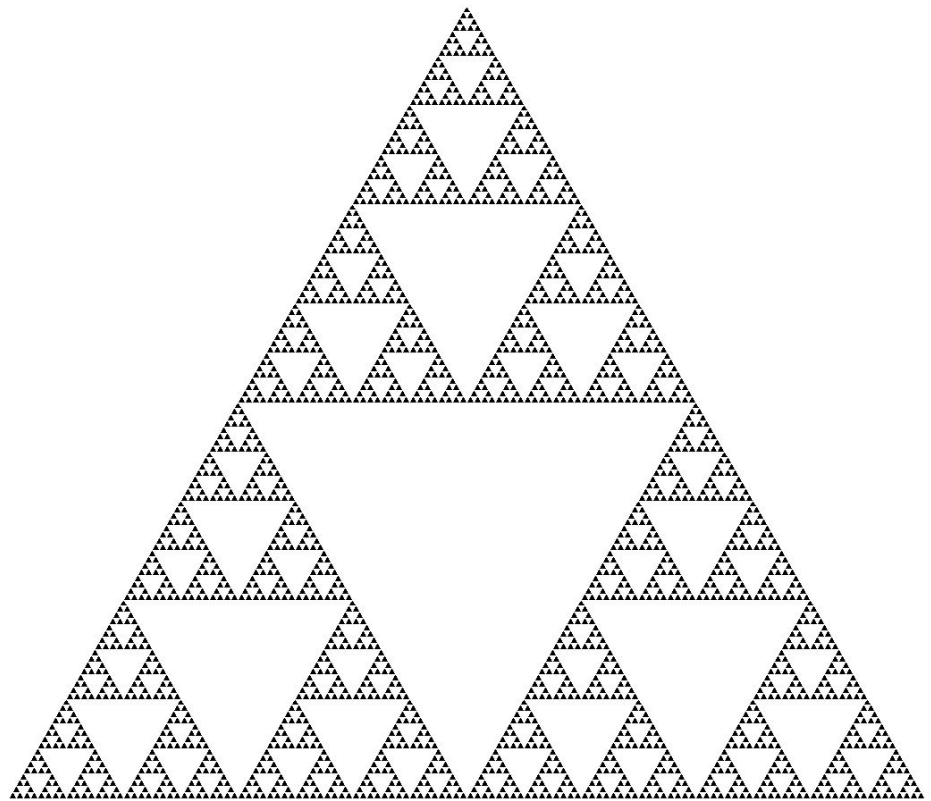
ОСНОВЫ ПРОГРАММИРОВАНИЯ





РЕКУРСИВНЫМ

называется способ построения объекта, в котором определение объекта включает аналогичные объекты в виде составных частей.



РЕКУРСИЯ

- **Рекурсия** – метод определения функции через её предыдущие и ранее определенные значения, а так же способ организации вычислений, при котором функция вызывает сама себя с другим аргументом.
- **Простая рекурсия** – вызов функции из неё же самой непосредственно.
- **Сложная, или косвенная рекурсия** – вызов через другие функции, например, функция а вызывает функцию б, а функция б функцию а.

РЕКУРСИЯ

- **Глубина рекурсии** – количество вложенных вызовов функции.

recursionLevel = 1



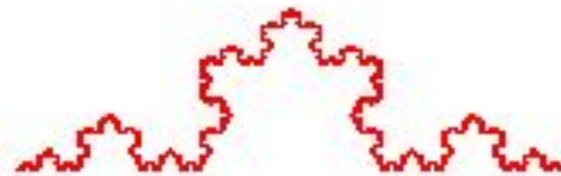
recursionLevel = 2



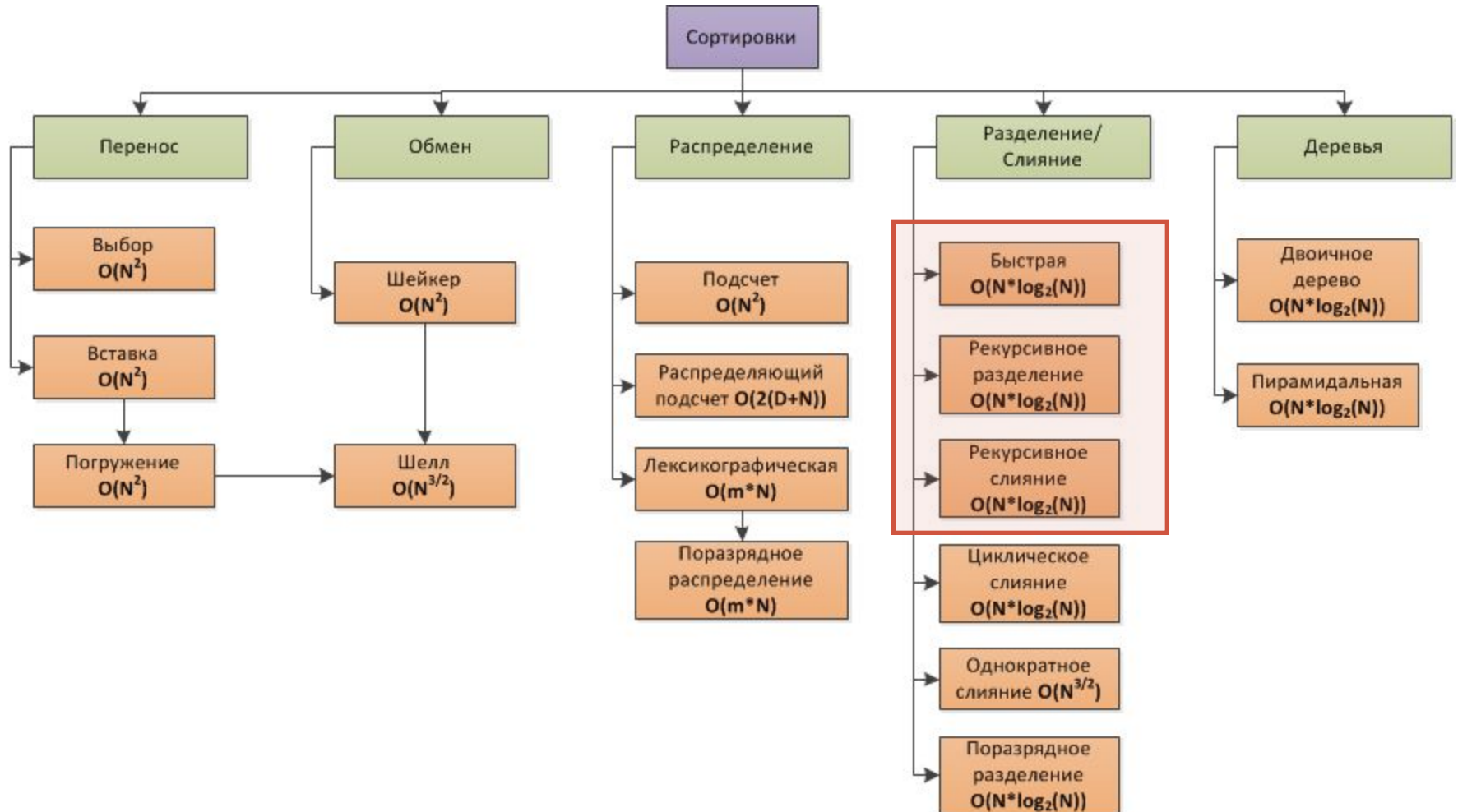
recursionLevel = 3



recursionLevel = 8



АЛГОРИТМЫ СОРТИРОВКИ



РЕКУРСИВНЫЕ АЛГОРИТМЫ

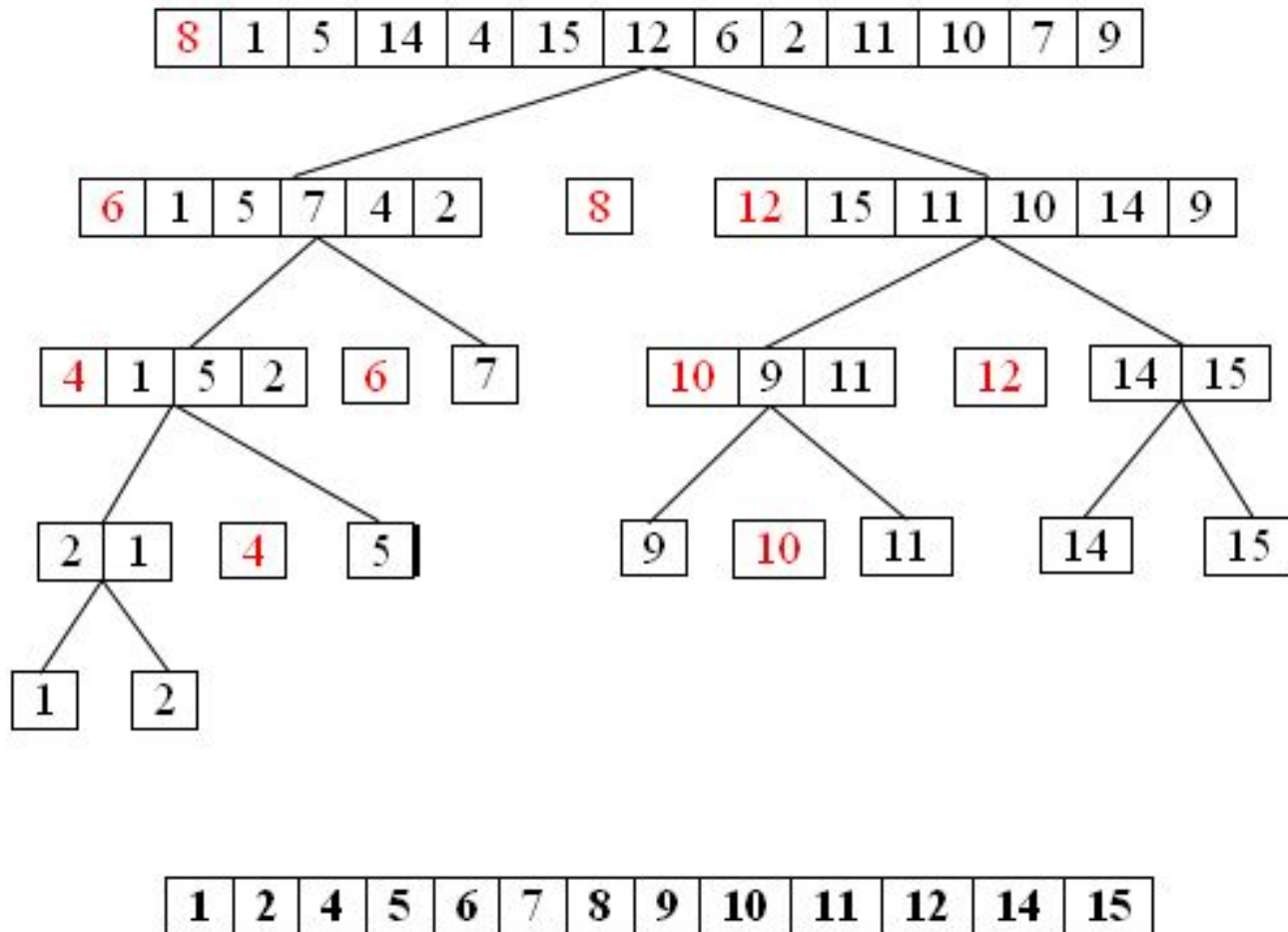
АЛГОРИТМЫ СОРТИРОВКИ



БЫСТРАЯ СОРТИРОВКА

- Стратегия – «**разделяй и властвуй**».
- Описание алгоритма:
- Выбираем в массиве некоторый элемент, который будем называть **опорным элементом**.
- **Операция разделения** массива: реорганизуем массив таким образом, чтобы все элементы, меньшие или равные опорному элементу, оказались слева от него, а все элементы, большие опорного — справа от него.
- Рекурсивно упорядочиваем подмассивы.

БЫСТРАЯ СОРТИРОВКА



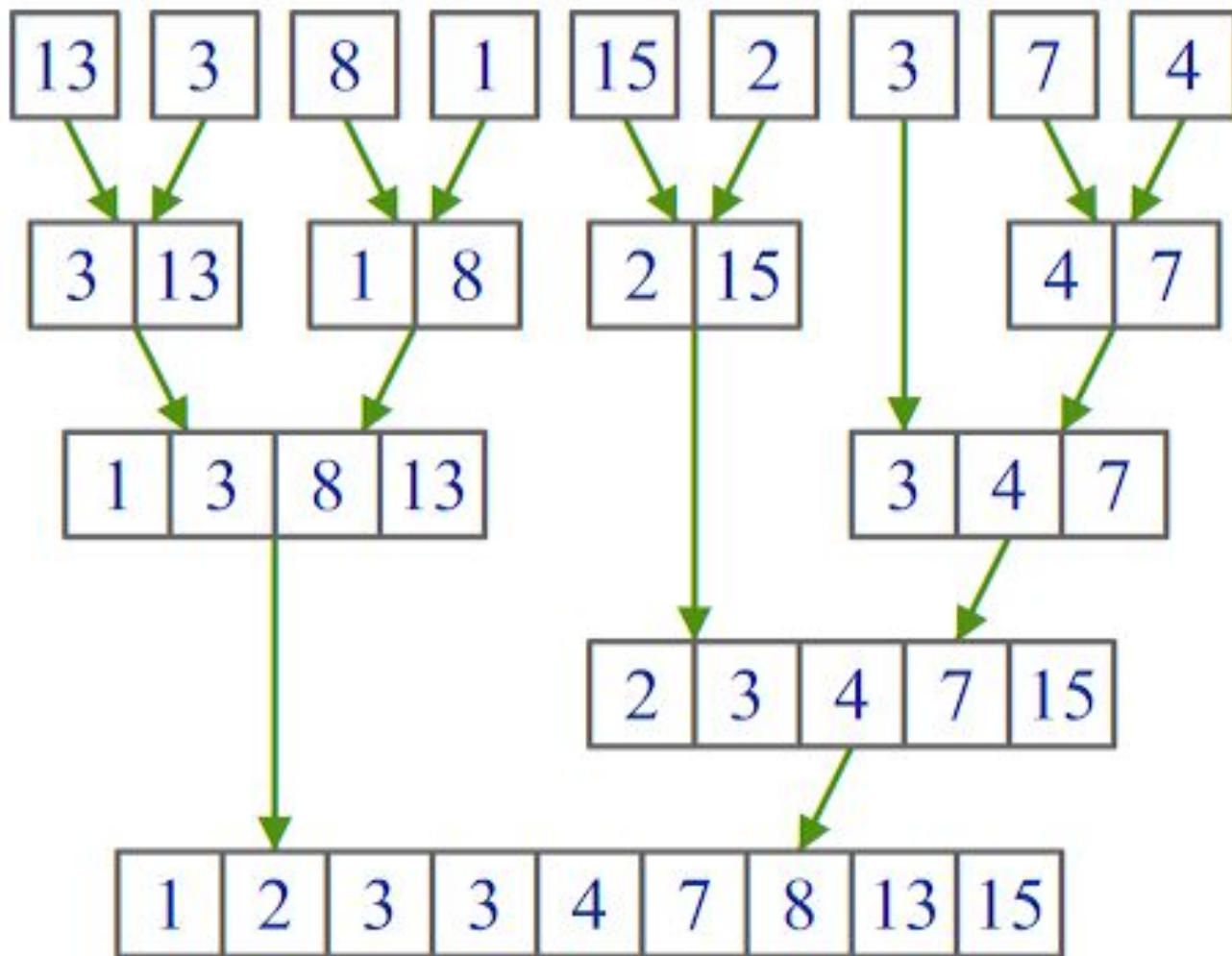
БЫСТРАЯ СОРТИРОВКА

- `void sort(int in[], int a, int b){`
- `int i,j,mode;`
- `if (a>=b) return;`
- `for (i=a, j=b, mode=1; i < j; mode >0 ? j-- : i++)`
- `if (in[i] > in[j]) {`
- `int c = in[i]; in[i] = in[j]; in[j]=c;`
- `mode = -mode;`
- `}`
- `sort(in,a,i-1);`
- `sort(in,i+1,b);`
- `}`

СОРТИРОВКА СЛИЯНИЕМ

- Стратегия – «**разделяй и властвуй**».
- Описание алгоритма:
- Сортируемый массив разбивается на две части примерно одинакового размера;
- Каждая из получившихся частей сортируется отдельно тем же самым алгоритмом;
- Два упорядоченных массива половинного размера соединяются в один.

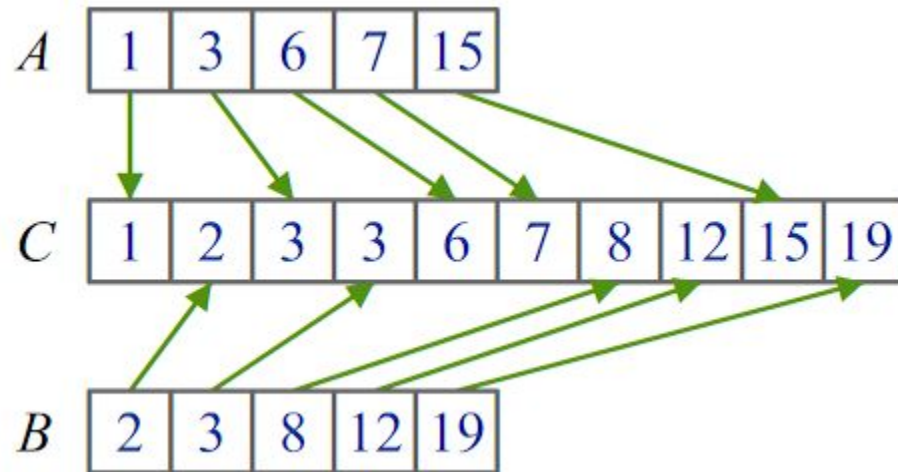
СОРТИРОВКА СЛИЯНИЕМ



ОПЕРАЦИЯ СЛИЯНИЯ

- Отделяем элемент, наименьший из двух имеющихся в началах исходных массивов, и присоединяем его к концу результирующего массива.
- Элементы переносим до тех пор, пока один из исходных массивов не закончится.
-
- После этого оставшийся «хвост» одного из входных массивов дописывается в конец результирующего массива.

ОПЕРАЦІЯ СЛИЯННЯ



ЗАДАЧИ

- 1. Реализовать алгоритм сортировки рекурсивным разделением списка.
- 2. Реализовать алгоритм сортировки рекурсивным слиянием списка.

