第二讲：排序与二分查找

王争

前 Google 工程师

排序算法回顾

	时间复杂度	是稳定排序?	是原地排序?
冒泡排序	$O(n^2)$	√	√
插入排序	$O(n^2)$	√	√
选择排序	$O(n^2)$	×	√
快速排序	$O(n\log n)$	×	√
归并排序	$O(n\log n)$	√	×
计数排序	$O(n+K)$ K是数据范围	√	×
桶排序	$O(n)$	√	×
基数排序	$O(dn)$ d是维度	√	×

排序算法回顾

```
public class ArraySort implements Runnable {

    private int number;

    public ArraySort(int number) {
        this.number = number;
    }

    public static void main(String[] args) {
        int[] numbers = new int[]{102, 338, 62, 9132, 580, 666};
        for (int number : numbers) {
            new Thread(new ArraySort(number)).start();
        }
    }

    @Override
    public void run() {
        try {
            Thread.sleep(this.number);
            System.out.println(this.number);
        } catch (InterruptedException e) {
            e.printStackTrace();
        }
    }
}
```

排序算法回顾

睡眠排序算法（Sleep Sort）

猴子排序算法（Bogo Sort）

慢速排序算法（Slow Sort）

侏儒排序算法（Stupid Sort）

臭皮匠排序算法（Stooge Sort）

目录

稳定性&原地性

什么是稳定排序算法?

我们现在要给电商交易系统中的“订单”排序。订单有两个属性，一个是下单时间，另一个是订单金额。

如果我们现在有 10 万条订单数据，我们希望按照金额从小到大对订单数据排序。对于金额相同的订单，我们希望按照下单时间从早到晚排序。

对于这样一个排序需求，我们怎么来做呢?

什么是稳定排序算法?

逻辑处理流程：

STEP 1: 根据金额排序

STEP 2: 统计金额相同的区间

STEP 3: 针对每个区间，按照下单时间排序

什么是稳定排序算法?

对一组数据进行排序，相同大小的数据，在排序前后，先后顺序不变。

5 7 3 2 1 3 9 3 -》1 2 3 3 3 5 7 9

什么是稳定排序算法?

需求：按金额从小到大对订单数据排序，对金额相同的，按下单时间从早到晚排序

逻辑处理流程：

1. 按照下单时间排序
2. 按照金额排序

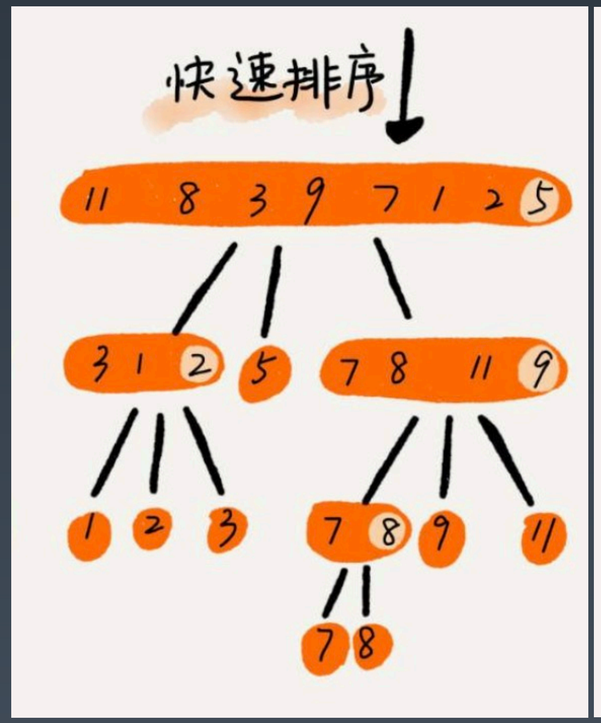
什么是原地排序算法?

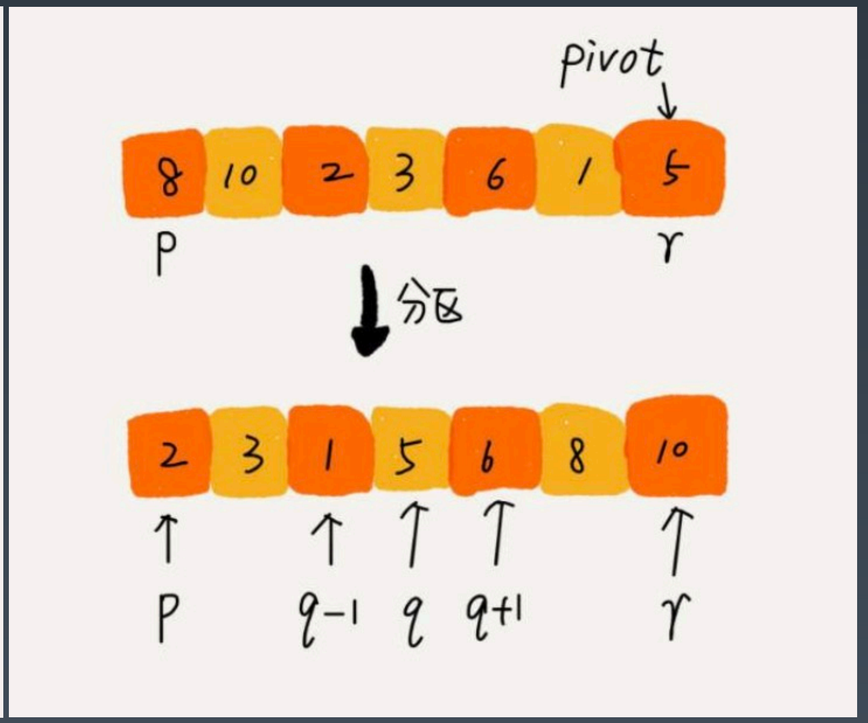
原地排序 ≠ 空间复杂度 O(1)

特殊情况：递归调用，栈空间的消耗

原地：除了存储数据本身的内存空间之外，不借助非常量级的额外空间

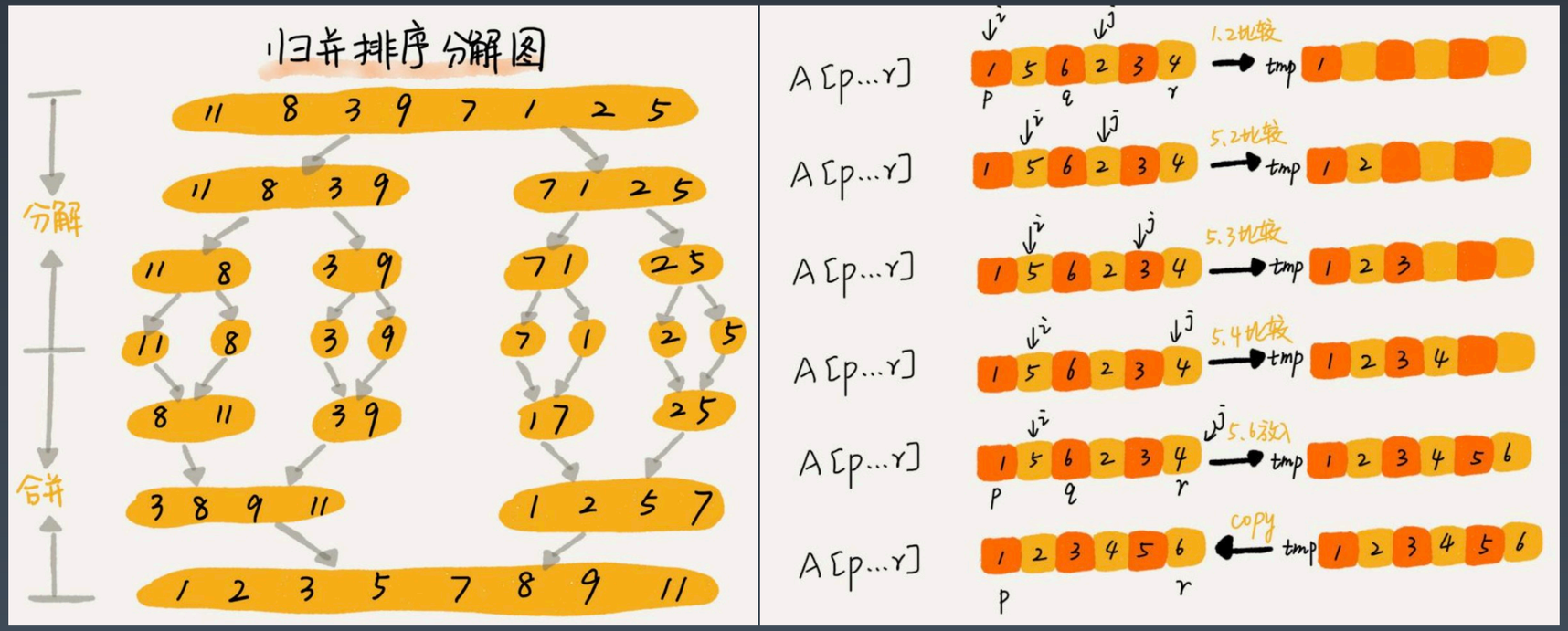
原地排序举例





快速排序：原地排序算法, 空间复杂度 O(logn)

非原地排序举例



归并排序：非原地排序算法

排序问题实战

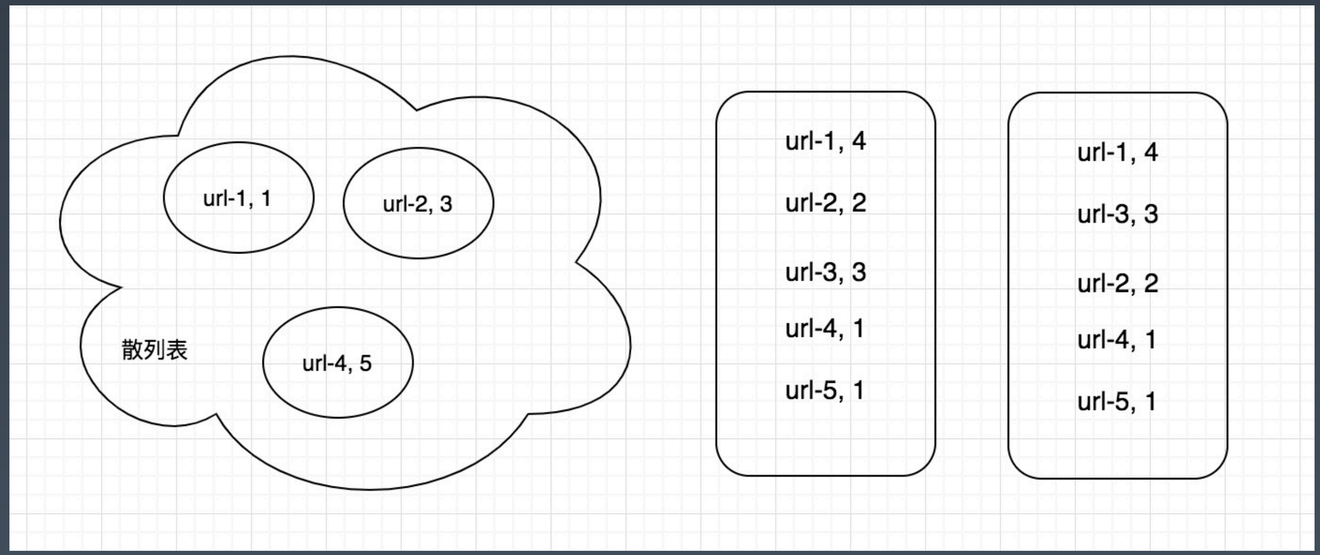
排序问题（一）

假设我们有 10 万条 URL 访问日志，如何按照访问次数给 URL 排序?

算法逻辑：

1. 统计每个 URL 的访问次数

2. 根据访问次数进行排序



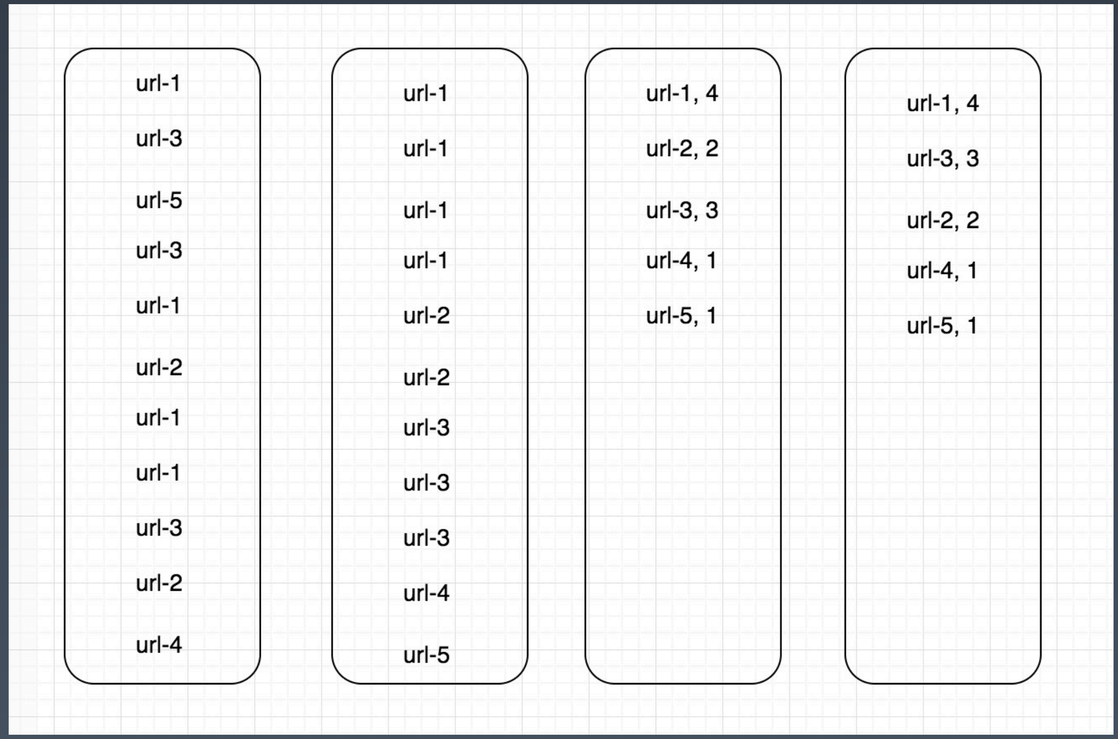
排序问题（一）

假设我们有 10 万条 URL 访问日志，如何按照访问次数给 URL 排序?

算法逻辑：

1. 统计每个 URL 的访问次数

2. 根据访问次数进行排序



排序问题（一）

如果访问日志的从 10 万 URL，变成了 1 亿 URL，该如何根据访问次数排序？

估算：

假设 URL 平均长度 128B

10 万 URL 占用内存：100,000*128B=10MB

1 亿 URL 占用的内存：100,000,000*128B = 10GB

排序问题（一）

如果访问日志的从 10 万 URL，变成了 1 亿 URL，该如何根据访问次数排序？

估算：1 亿 URL 占用的内存：100,000,000*128B = 10GB

算法思路：

分治思想、桶排序、归并排序、哈希算法、多机并行处理、MapReduce

排序问题（一）

如果访问日志的从 10 万 URL，变成了 1 亿 URL，该如何根据访问次数排序?

估算：1 亿 URL 占用的内存：100,000,000*128B = 10GB

算法一：借助归并排序的思想

1. 顺序读取日志文件，每 512MB 数据放入内存中排序，排好序之后的数据，写入新的文件中（log_1.txt, log_2.txt…）

2. 借助归并排序中合并两个有序数组为一个有序数组的算法，将已经有序的小日志文件，合并为一个有序的大日志文件

3. 顺序读取有序的大日志文件，统计得到每个 URL 对应的访问次数: (url1,100), (url2,23)……

4. 根据访问次数，再次执行 1，2 两步

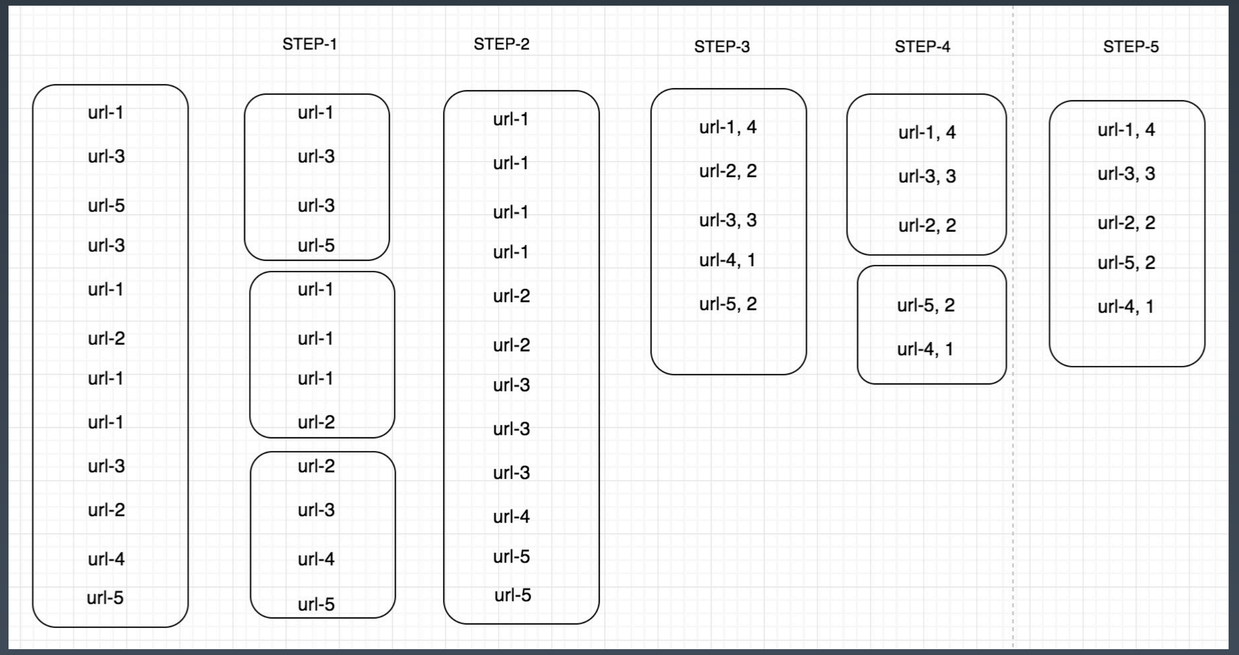
排序问题（一）

1. 顺序读取日志文件，每 512MB 数据放入内存中排序，排好序之后的数据，写入新的文件中（log_1.txt, log_2.txt…）

2. 借助归并排序中合并两个有序数组为一个有序数组的算法，将已经有序的小日志文件，合并为一个有序的大日志文件

3. 顺序读取有序的大日志文件，统计得到每个 URL 对应的访问次数: url1,100), (url2,23)……

4. 根据访问次数，再次执行 1，2 两步



排序问题（一）

如果访问日志的从 10 万 URL，变成了 1 亿 URL，该如何根据访问次数排序?

估算：1 亿 URL 占用的内存：100,000,000*128B = 10GB

算法一：借助哈希算法+桶排序的思路

1. 利用哈希算法（比如：md5(url)%20），将 URL 分成 20 个小文件

2. 每个小文件单独排序，并且统计 URL 出现的次数: (url1,100), (url2,23)……

3. 每个小文件根据 URL 访问次数，再行排序

4. 借助归并排序中，合并两个有序数组为一个有序数组的算法，将已经有序的小日志文件，合并为一个有序的大日志文件

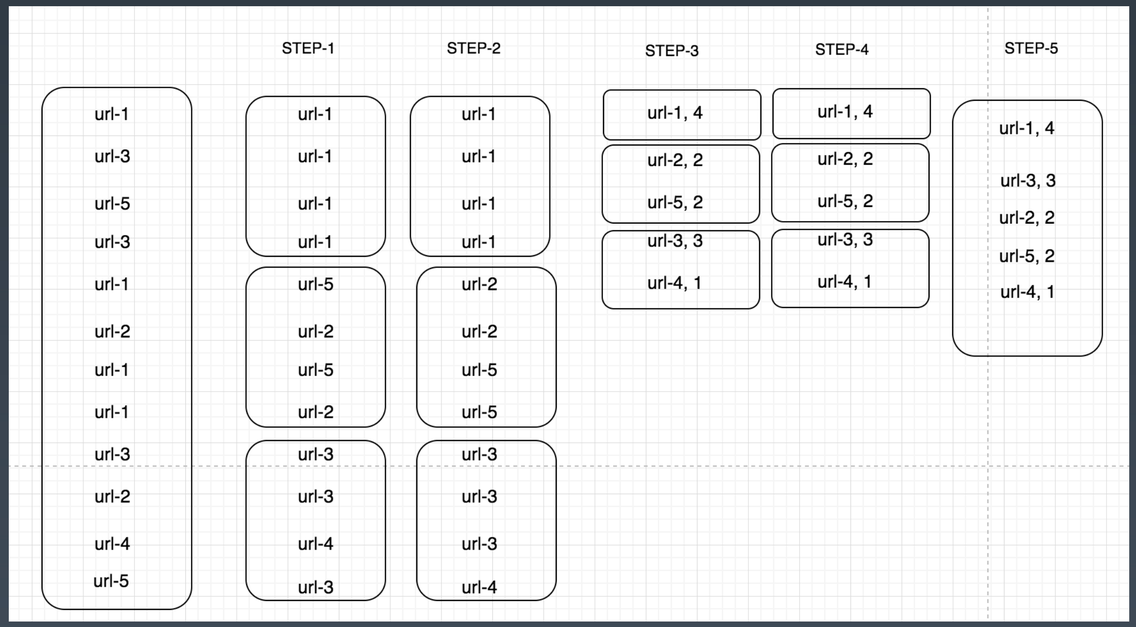
排序问题（一）

1. 利用哈希算法（比如：md5(url)%20），将URL分成 20 个小文件

2. 每个小文件单独排序，并且统计 URL 出现的次数: (url1,100), (url2,23)……

3. 每个小文件根据 URL 访问次数，再行排序

4. 借助归并排序中，合并两个有序数组为一个有序数组的算法，将已经有序的小日志文件，合并为一个有序的大日志文件



排序问题（二）

1. 有两个字符串数组，每个数组有 10 万条字符串，如何快速找出两个数组中相同的字符串?

2. 有两个包含 1 亿行字符串的文件，如何快速找出相同的字符串?

排序问题（三）

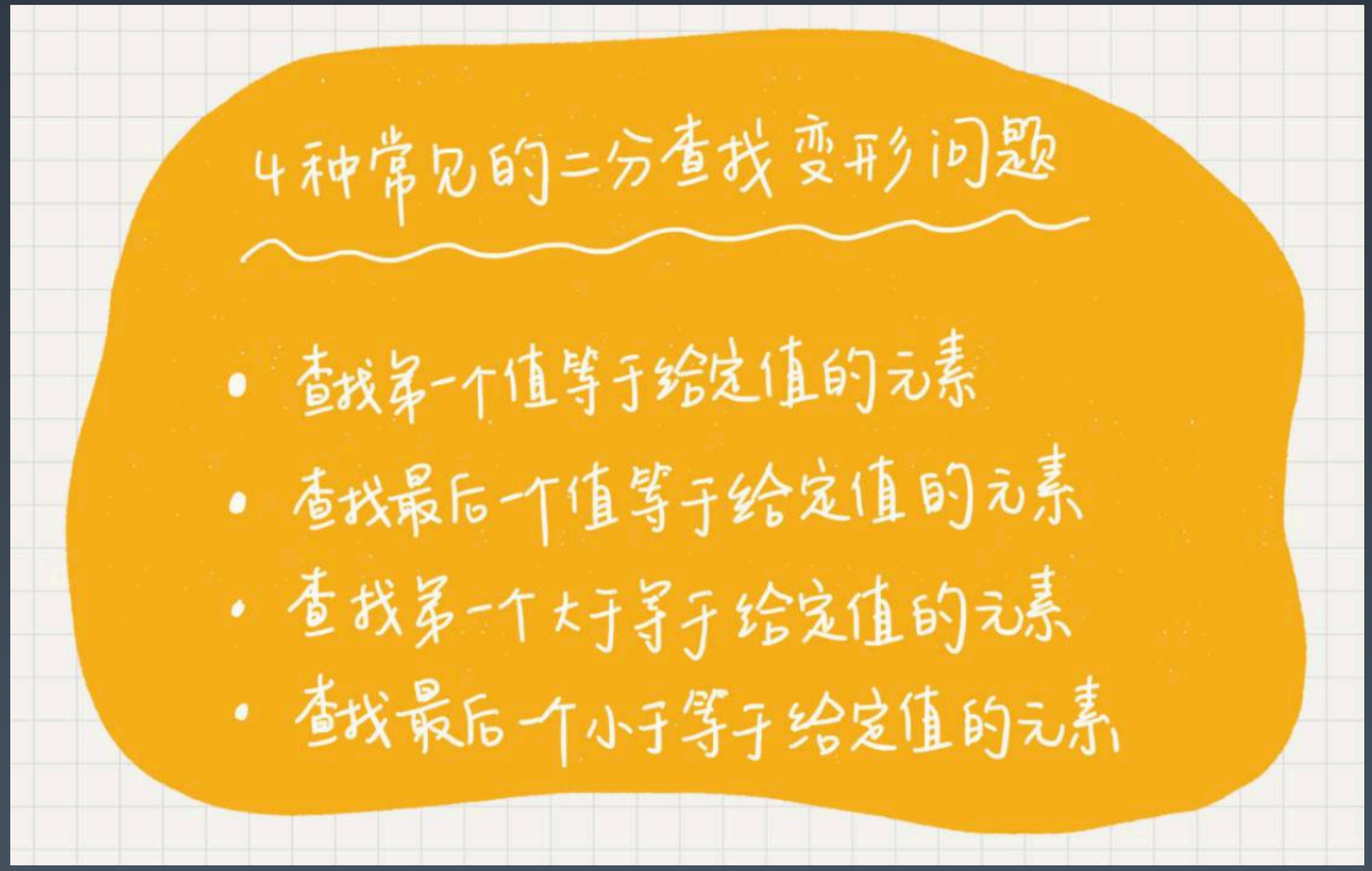
假设我们现在需要对 D，a，F，B，c，A，z 这个字符串进行排序，要求将其中所有小写字母都排在大写字母的前面，但小写字母内部和大写字母内部不要求有序。比如经过排序之后为 a，c，z，D，F，B，A，这个如何来实现呢?

排序问题（三）

如果字符串中存储的不仅有大小写字母，还有数字。要将小写字母的放到前面，大写字母放在最后，数字放在中间，不用排序算法，又该怎么解决呢?

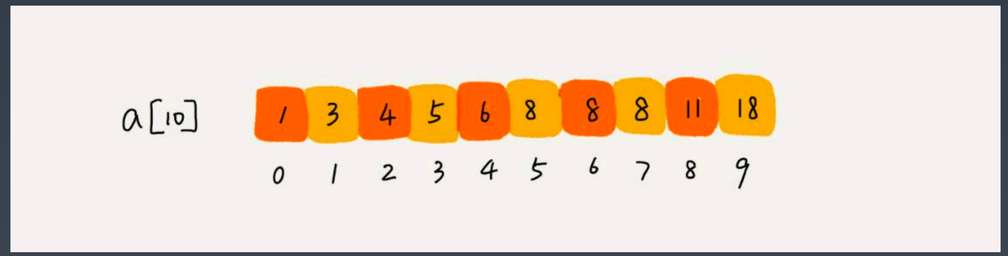
二分查找

各种二分查找的变体


4种常见的二分查找变形问题
查找第一个值等于给定值的元素
查找最后一个值等于给定值的元素
查找第一个大于等于给定值的元素
查找最后一个小于等于给定值的元素

查找第一个值等于给定值的元素

```
public int bsearch(int[] a, int n, int value) {
  int low = 0;
  int high = n - 1;
  while (low <= high) {
    int mid =  low + ((high - low) >> 1);
    if (a[mid] > value) {
      high = mid - 1;
    } else if (a[mid] < value) {
      low = mid + 1;
    } else {
      if ((mid == 0) || (a[mid - 1] != value)) return
mid;
      else high = mid - 1;
    }
  }
  return -1;
}
```



THANKS!