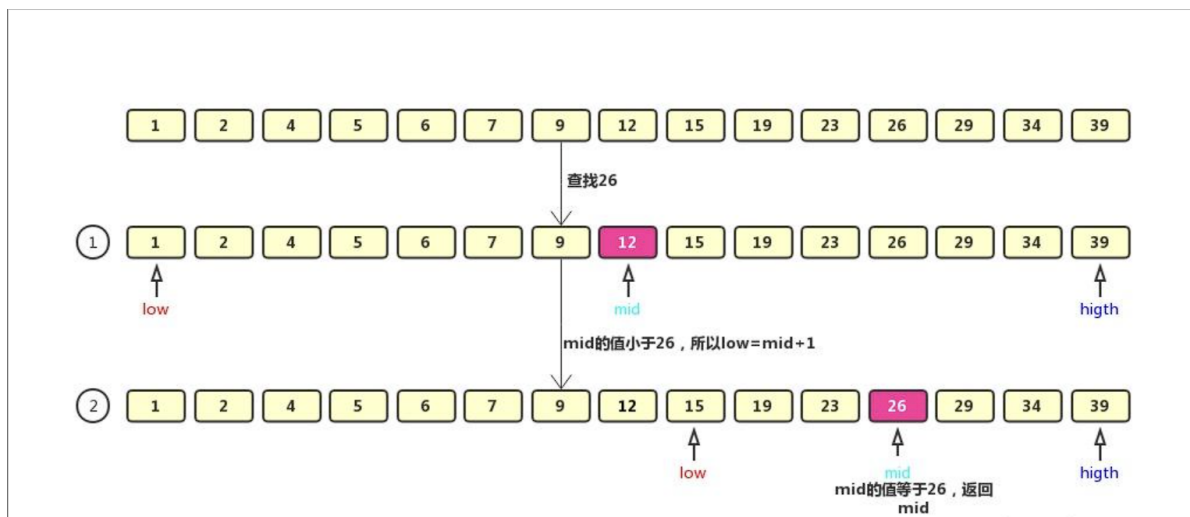


二分查找

二分查找（Binary Search）算法，也叫折半查找算法。二分查找的思想非常简单，有点类似分治的思想。二分查找针对的是一个有序的数据集合，每次都通过跟区间的中间元素对比，将待查找的区间缩小为之前的一半，直到找到要查找的元素，或者区间被缩小为 0。



随机排名

各位同学，通过数组元素反转的案例，我们学会了如何对两个数据进行交换。接下来，我们再学习随机排名案例，将数据交换的思路再巩固一下。

先来看一下需求

1. 需求：某公司开发部5名开发人员，要进行项目进展汇报演讲，现在采取随机排名后进行汇报。请先依次录入5名员工的工号，然后展示出一组随机的排名顺序。

分析一下随机排名的思路

1. 在程序中录入5名员工的工号存储起来 ---> 使用动态初始化数组的方式。
2. 依次遍历数组中的每个数据。
3. 每遍历到一个数据，都随机一个索引值出来，让当前数据与该索引位置处的数据进行交换。

如下图所示，每次遍历到一个元素，随机将当前位置元素和随机索引元素换位置。

10	20	30	40	50
0	1	2	3	4

核心思路：遍历过程中，每获取一个元素，同时随机产生一个索引，让当前索引位置的元素和随机索引位置的元素互换

当前索引：i = 0;
随机索引：index = 3;
arr[i]和arr[index]换位置

当前索引：i = 1;
随机索引：index = 2
arr[i]和arr[index]换位置

代码如下

```

1  public class Test3 {
2      public static void main(String[] args) {
3          // 目标：完成随机排名
4          // 1、定义一个动态初始化的数组用于存储5名员工的工号
5          int[] codes = new int[5];
6
7          // 2、提示用户录入5名员工的工号。
8          Scanner sc = new Scanner(System.in);
9          for (int i = 0; i < codes.length; i++) {
10             // i = 0 1 2 3 4
11             System.out.println("请您输入第" + (i + 1) + "个员工的工
号: ");
12             int code = sc.nextInt();
13             codes[i] = code;
14         }
15
16         // 3、打乱数组中的元素顺序。
17         // [12, 33, 54, 26, 8]
18         // i      index
19         Random r = new Random();
20         for (int i = 0; i < codes.length; i++) {
21             // codes[i]
22             // 每遍历到一个数据，都随机一个数组索引范围内的值。
23             // 然后让当前遍历的数据与该索引位置处的值交换。
24             int index = r.nextInt(codes.length); // 0 - 4
25             // 定义一个临时变量记住index位置处的值
26             int temp = codes[index];
27             // 把i位置处的值赋值给index位置处
28             codes[index] = codes[i];
29             // 把index位置原来的值赋值给i位置处
30             codes[i] = temp;
31         }

```

```

32
33         // 4、遍历数组中的工号输出即可
34         for (int i = 0; i < codes.length; i++) {
35             System.out.print(codes[i] + " ");
36         }
37     }
38 }

```

✧ 多维数组

在 Java 中，可以使用多维数组来存储多个数据值，以便更好地组织和访问这些数据。Java 中的多维数组是一种数组的数组，即一个数组的元素也是一个数组。Java 中的多维数组可以包含任意数量的维度。在处理多维数组时，需要注意数组下标的范围，以避免数组越界异常。同时，还可以使用循环嵌套来遍历多维数组中的所有元素。

二维数组

Java 中定义和操作多维数组的语法与一维数组类似。在实际应用中，三维及其以上的数组使用很少，主要使用二维数组。使用二维数组同一维数组的步骤，（1）定义数组、（2）为数组元素分配内存、（3）数组元素初始化、（4）使用数组。下面主要以二维数组为例进行讲解。

H5 定义二维数组

定义二维数组的语法规则如下：

数据类型[][] 数组名；

或者

数据类型 数组名[][]；

语法解析：

[][] 表示二维数组，前面的[]表示第一维，后面的[]表示第二维。

[][] 放在数组名的前面或后面都是正确的。

H5 分配内存

```
1  int[][] arr = new int[3][4];
2
```

H5 数组元素初始化

```
1  // 二维数组初始化
2  int[][] arr = new int[3][4]; // 动态初始化
3  arr[0][0] = 1;
4  int[][] arr1 = new int[][]{ // 静态初始化
5      {1, 2, 3},
6      {2, 3},
7      {3, 4, 5, 4}
8  };
```

H5 二维数组的迭代

```
1  for (int i = 0; i < arr1.length; i++) {
2      System.out.println(arr1[i]);
3      for (int i1 = 0; i1 < arr1[i].length; i1++) {
4          System.out.println(arr1[i][i1]);
5      }
6  }
7
8  for (int[] t : arr1) {
9      for (int a : t) {
10         System.out.println(a);
11     }
12 }
```