

1 Binary Search

二分查找法 折半查找法

问题:

在有序序列 s 中查找元素 x 是否存在。

解法:

有序序列 s 可以是升序或降序的，即从小到大或从大到小。本问题中假设 s 是升序的。

在这样的序列 s 中想要找出某个元素 x 是否存在，首先检查序列 s 最小、最大和中间三个位置处的元素 low 、 $high$ 和 mid ，其中 $mid = \frac{high+low}{2}$ 。

若 x 与 mid 相等则 mid 即为所求，算法结束；若 x 小于 mid ，则 x 在 mid 左边的某一位置；若 x 大于 mid ，则 x 在 mid 右边的某一位置。然后将序列的检查范围缩小，继续递归的重复上面的过程，直到找到 x 等于 mid 。

对于下面这个长度为 10 的序列 s ，查找 $x = 48$ ，初始时设置 low 、 $high$ 和 mid 三个值，

分为作为序列 s 的最左、最右和中间的下标值，即 $low = 0$ ， $high = 9$ ， $mid = \frac{0+9}{2} = 4$ （

整数计算直接取整）：

0	1	2	3	4	5	6	7	8	9
0	3	7	8	12	15	30	36	48	50

(1) $mid = 4$ 时， $s[mid] < x$ ，因此 x 在 mid 与 $high$ 之间，并且不包括 mid ，设置 $low =$

$$mid + 1 = 5, \quad mid = \frac{5+9}{2} = 7;$$

(2) $mid = 7$ 时， $s[mid] < x$ ， x 仍然在 mid 与 $high$ 之间，设置 $low = mid + 1 = 8$ ， $mid =$

$$\frac{8+9}{2} = 8;$$

(3) $s[mid] = x$ ，返回 x 所在下标值 8，算法结束。

如果查找的 x 为 49，则在第(3)步中仍然有 $s[mid] < x$ ，则仍然设置 $low = mid + 1 = 9$ ，

$mid = \frac{9+9}{2} = 9$ ，有 $s[mid] > x$ ，因此设置 $high = mid - 1 = 8$ ，此时 $low \leq high$ 不再成

立，算法结束， x 没有找到。