Assignment Two

Name:	陈禾宇
Student Number:	1950698

Direction:

Please answer all the questions below and hand in your answers before the due day. All work, must be handed in **on time**.

Due day:

May. 10, 2021

Questions:

1. Given an array $A = \{13, 15, 124, 28, 44, 28, 27, 5, 71\}$. Please solve following problems:
   - Arrange A in descending order by **insertion sort**.
   - Arrange A in descending order by **quick sort**.
   - Describe the basic idea of binary search for decrement arrays and give a non-recursive algorithm and also the recursive version.
   - Use above algorithms to find the elements (i.e. 13, 124) and provide necessary details of the searching process.

Insertion sort:

① 15 13 124 28 44 28 27 5 71

② 124 15 13 28 44 28 27 5 71

③ 124 28 15 13 44 28 27 5 71

④ 124 44 28 15 13 28 27 5 71

⑤ 124 44 28 28 15 13 27 5 71

⑥

124	44	28	28	27	15	13	5	71

⑦

124	44	28	28	27	15	13	5	71

⑧

124	71	44	28	28	27	15	13	5

Quick sort

①

71	15	124	28	44	28	27	13	5

[0, 8]

②

124	71	15	28	44	28	27	13	5

[0, 6] [8, 8]

③

124	71	27	28	44	28	15	13	5

[0, 0] [2, 6]

④

124	71	28	28	44	27	15	13	5

[2, 5]

⑤

124	71	44	28	28	27	15	13	5

[2, 4]

后面没变化了...

二分搜索法

基本思想：n个元素分两半 取x与$a[\frac{n}{2}]$比较

如果 $x = a[\frac{n}{2}]$ 算法终止 找到了

如果 $x < a[\frac{n}{2}]$ 去右边找

如果 $x > a[\frac{n}{2}]$ 去左边找

$n = 0$ 算法结束 没找到 (左边界大于右边界)

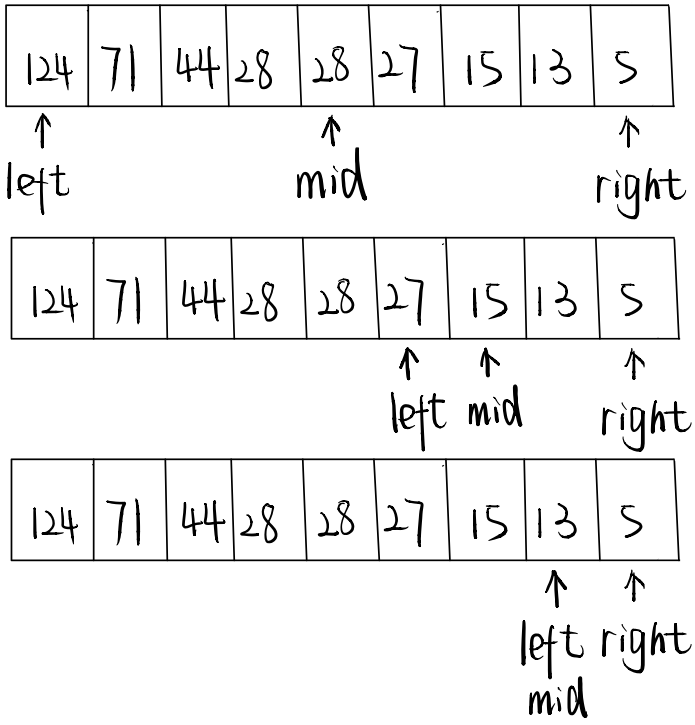
非递归版本

```
while (left <= right)
{  int mid = (left+right)/2;
   if (a[mid] == x)
        return mid;
   else if (a[mid] > x)
        mid = right-1;
   else if (a[mid] < x)
        mid = left+1;
}
return NOFOUND;
```

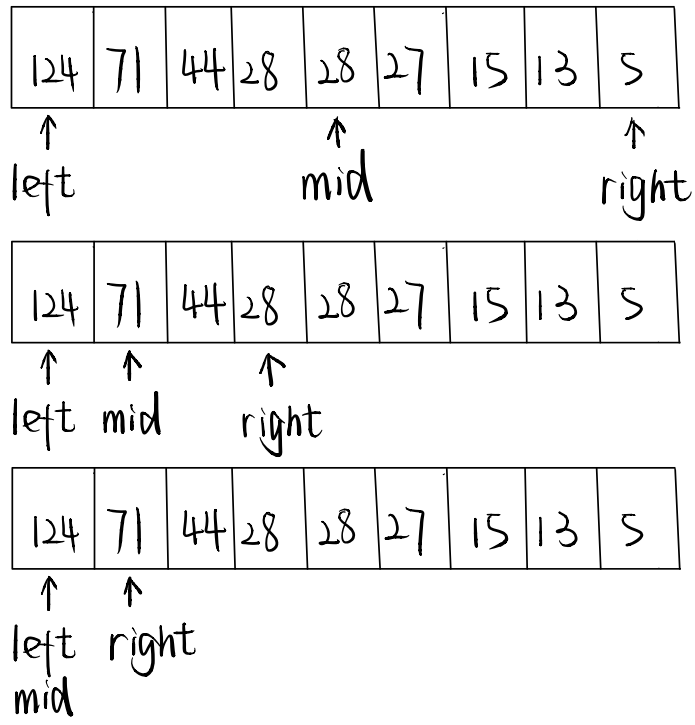
递归版本：

```
int Search (int left, int right, int x)
{ if (left > right)
     return NOFOUND;
  int mid = (left+right)/2;
  if (a[mid] == x)
    return mid;
  else if (a[mid] > x)
    return Search (left, mid-1, x);
  else
     return Search (mid+1, right, x);
}
```

找 13

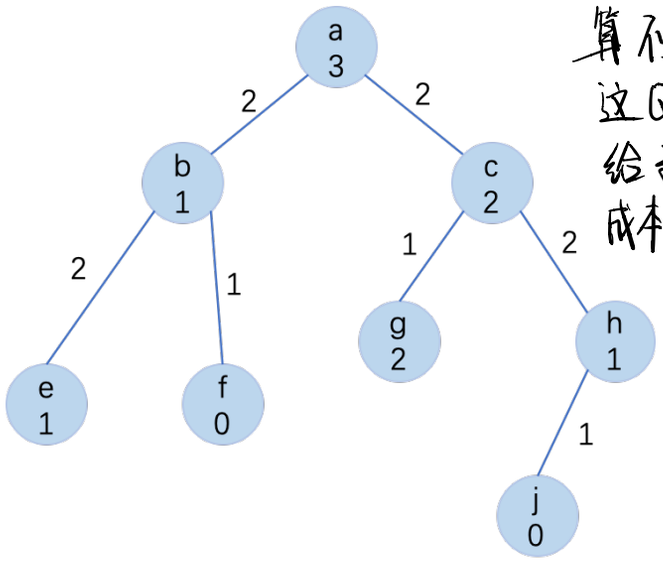

124 71 44 28 28 27 15 13 5
left
mid
right
a[mid] > 13
124 71 44 28 28 27 15 13 5
left mid
right
a[mid] > 13
124 71 44 28 28 27 15 13 5
left right
mid
a[mid] == 13
return mid.

找124


124 71 44 28 28 27 15 13 5
left
mid
right
a[mid] < 124
124 71 44 28 28 27 15 13 5
left mid
right
a[mid] < 124
124 71 44 28 28 27 15 13 5
left right
mid
a[mid] == 124
return mid;

2. Consider the minimal cost search problem represented in the figure, where a is the start node and there are goal nodes at f and j. For each node, the heuristic cost is indicated on the node, and for each arc, the arc cost is indicated along the arc. What is the upper bound when only the start node has been explored? Which goal node is found first by Branch&Bound? What is the upper bound immediately after the first goal node is found? Is the second goal found by Branch&Bound?

这个启发式成本到底
算不算到 cost 里面呢?
这QQ群里说优先级
给我整迷茫了,那这启发式
成本是算优先级还是cost呢



上界是 12 当只有 start node 被访问.
f 先被发现,上界改为 7
第二个 goal 不会被访问,因为 C 点那已经等于 7 了
不同不走了。