

*1、输出 Fibonacci 数列：1, 1, 2, 3, 5, 8, 的前 20 个数。

*2、求出一组整数中的最大值。

**3、请选择一种排序方法，将一组数按从小到大排序。请尽量优化程序。

1) **选择法**：首先在未排序序列中找到最小元素，存放到排序序列的起始位置，然后，再从剩余未排序元素中继续寻找最小元素，然后放到已排序序列的末尾。以此类推，直到所有元素均排序完毕。

2) **冒泡排序法**：它重复地访问要排序的数列，一次比较两个元素，如果他们的顺序错误就把他们交换过来。访问数列的工作重复地进行直到没有再需要交换，也就是说该数列已经排序完成。这个算法的名字由来是因为越小的元素会经由交换慢慢“浮”到数列的顶端。

算法描述：

①比较相邻的元素。如果第一个比第二个大，就交换他们两个。

②对每一对相邻元素作同样的工作，从开始第一对到结尾的最后一对。在这一点，最后的元素应该会是最大的数。

③针对所有的元素重复以上的步骤，除了最后一个。

④持续每次对越来越少的元素重复上面的步骤，直到没有任何一对数字需要比较。

3) **插入排序法**：对于未排序数据，在已排序序列中从后向前扫描，找到相应位置并插入。

算法描述：

①从第一个元素开始，该元素可以认为已经被排序

②取出下一个元素，在已经排序的元素序列中从后向前扫描

③如果该元素（已排序）大于新元素，将该元素移到下一位置重复步骤③，直到找到已排序的元素小于或者等于新元素的位置。

④将新元素插入到该位置后。

⑤重复步骤②~⑤

**4、查找：在一组数中查找某给定的数 x ，如果找到则输出该数所在下标位置，如果找不到则输出“not found”。请选择一种查找方法。

1) 顺序查找

将查找值与数组中的元素逐个进行比较，相等即为查找成功，否则查找失败。

2) 二分查找

在**有序**数组（已排好序的数组）中查找某一特定元素。查找过程从数组的中间元素开始，如果中间元素正好是要查找的元素，则查找过程结束；如果某一特定元素大于或者小于中间元素，则在数组大于或小于中间元素的那一半中查找，而且跟开始一样从中间元素开始比较。如果在某一步骤数组为空，则代表找不到。这种查找算法每一次比较都使查找范围缩小一半。

**5、输出杨辉三角形

```
1
1 1
1 2 1
1 3 3 1
1 4 6 4 1
.....
```

**6、如果一个班级有 40 位同学，共有 4 门考试课程，从键盘输入各位学生的考试成绩，1) 计算每位学生的平均分。2) 计算每门课程的平均分。

***7、给定一个 $n \times n$ 矩阵 A 。矩阵 A 的鞍点是一个位置 (i, j) ，在该位置上的元素是第 i 行上的最大数，第 j 列上的最小数。一个矩阵 A 也可能没有鞍点。请找出 A 的鞍点。