# 依次接收用户输入的3个数，排序后打印

1. 转换int后，判断大小排序
2. 使用max函数
3. 使用列表的sort方法
4. 冒泡法

```
nums = []

for i in range(3):
    nums.append(int(input('{}: '.format(i))))

if nums[0] > nums[1]:
    if nums[0] > nums[2]:
        i3 = nums[0]
        if nums[1] > nums[2]:
            i2 = nums[1]
            i1 = nums[2]
        else:
            i2 = nums[2]
            i1 = nums[1]
    else:
        i2 = nums[0]
        i3 = nums[2]
        i1 = nums[1]
else: # 0<1
    if nums[0] > nums[2]:
        i3 = nums[1]
        i2 = nums[0]
        i1 = nums[2]
    else: # 0<2
        if nums[1] < nums[2]: # 1<2
            i1 = nums[0]
            i2 = nums[1]
            i3 = nums[2]
        else: # 1 > 2
            i1 = nums[0]
            i2 = nums[2]
            i3 = nums[1]
```

```
print(i1,i2,i3)
```

改进

```
nums = []
out = None
for i in range(3):
    nums.append(int(input('{}: '.format(i))))

if nums[0] > nums[1]:
    if nums[0] > nums[2]:
        if nums[1] > nums[2]:
            out = [2,1,0]
        else:
            out = [1,2,0]
    else:
        out = [1,0,2]
else: # 0<1
    if nums[0] > nums[2]:
        out = [2,0,1]
    else: # 0<2
        if nums[1] < nums[2]: # 1<2
            out = [0,1,2]
        else: # 1 > 2
            out = [0,2,1]
out.reverse()
for i in out:
    print(nums[i],end=', ')
```

```
# max min的实现
nums = []
out = None
for i in range(3):
    nums.append(int(input('{}: '.format(i))))

# 此处不能使用for循环，不能一边迭代该列表，同时删除或者增加该列表
while True:
    cur = min(nums)
    print(cur)
    nums.remove(cur)
```

```
    if len(nums) == 1:
        print(nums[0])
        break
```

```
# 列表sort实现
nums = []

for i in range(3):
    nums.append(int(input('{}: '.format(i))))

nums.sort()
print(nums)
```

## 冒泡法

```
# 简单实现
numlist = [
    [1, 9, 8, 5, 6, 7, 4, 3, 2],
    [1, 2, 3, 4, 5, 6, 7, 8, 9]
]
nums = numlist[0]
print(nums)
length = len(nums)
count_swap = 0
count = 0
# bubble_sort
for i in range(length):
    for j in range(length-i-1):
        count += 1
        if nums[j] > nums[j+1]:
            tmp = nums[j]
            nums[j] = nums[j+1]
            nums[j+1] = tmp
            count_swap += 1
print(nums, count_swap, count)
```

```
# 优化
num_list = [
    [1, 9, 8, 5, 6, 7, 4, 3, 2],
```

```
    [1, 2, 3, 4, 5, 6, 7, 8, 9],
    [1, 2, 3, 4, 5, 6, 7, 9, 8]
]
nums = num_list[2]
print(nums)
length = len(nums)
count_swap = 0
count = 0
# bubble_sort
for i in range(length):
    flag = False
    for j in range(length-i-1):
        count += 1
        if nums[j] > nums[j+1]:
            tmp = nums[j]
            nums[j] = nums[j+1]
            nums[j+1] = tmp
            flag = True # swapped
            count_swap += 1
    if not flag:
        break
print(nums, count_swap, count)
```



# 字符串习题

## 用户输入一个数字

1. 判断是几位数
2. 打印每一位数字及其重复的次数
3. 依次打印每一位数字，顺序个、十、百、千、万…位

```
num = ''
# 数字输入的简单判断
while True:
    num = input('Input a positive number >>> ').strip().lstrip('0')
    if num.isdigit():
        break
```

```
print("The length of {} is {}.".format(num, len(num)))

# 倒序打印1
for i in range(len(num),0,-1):
    print(num[i-1], end=' ')
print()

# 倒序打印2
for i in reversed(num):
    print(i, end=' ')
print()

# 负索引方式打印
for i in range(len(num)):
    print(num[-i-1], end=' ')
print()

# 判断0-9的数字在字符串中出现的次数，每一次迭代都是用count，都是O(n)问题
counter = [0]*10
for i in range(10): # 10*n
    counter[i] = num.count(str(i))
    if counter[i]:
        print("The count of {} is {}".format(i, counter[i]))

print('~'*20)
# 迭代字符串本身的字符
counter = [0]*10
for x in num: # unique(n) * n ,unique(n)取值[1,10]
    i = int(x)
    if counter[i] == 0:
        counter[i] = num.count(x)
        print("The count of {} is {}".format(x, counter[i]))

print('~'*20)
# 迭代字符串本身的字符
counter = [0]*10
for x in num: # n
    i = int(x)
    counter[i] += 1
```

```
for i in range(len(counter)):
    if counter[i]:
        print("The count of {} is {}".format(i, counter[i]))
```

# 输入5个数字，打印每个数字的位数，将这些数字排序打印，要求升序打印

```
nums = []
while len(nums) < 5:
    num = input("Please input a number:").strip().lstrip('0')
    if not num.isdigit():
        continue
    print('The length of {} is {}'.format(num, len(num)))
    nums.append(int(num))
print(nums)

# sort方法排序
lst = nums.copy()
lst.sort() # 就地修改
print(lst)

# 冒泡法
for i in range(len(nums)):
    flag = False
    for j in range(len(nums)-i-1):
        if nums[j] > nums[j+1]:
            tmp = nums[j]
            nums[j] = nums[j+1]
            nums[j+1] = tmp
            flag = True
    if not flag:
        break
print(nums)
```