

第一题

A

```
1 s: good boy #直接输出s的元素
2 s[-1]: y #输出字符串中最后一个元素
3 s[5:8]: boy #从第5个元素开始切片输出，第8个元素结束
4 s[::2]: go o #以步长为2，从第一个元素开始输出
5 s[-2::-1]: ob doog #从倒数第二个元素开始，反向切片
6 s + t: good boybeautiful girl #字符串s和字符串t拼接
7 s * 2: good boygood boy #字符串输出两次
```

B

```
1 s.upper(): GOOD BOY #把字符串的内容全改为大写
2 s.lower(): good boy #把字符串的内容全改为小写
3 s.find('o'): 1 #输出第一次找到相同字符的元素的位置（下标从0开始）
4 s.replace('boy', 'children'): good children #用后者替代前者
5 s.split(' '): ['good', 'boy'] #根据空格，将字符串分割为若干份，放入列表中存储
```

第二题

源代码：

```
1 a = eval(input('输入三角形的第1条边: '))
2 b = eval(input('输入三角形的第2条边: '))
3 c = eval(input('输入三角形的第3条边: '))
4
5 if a + b > c and a + c > b and b + c > a:
6     C = a + b + c
7     p = (a + b + c) / 2
8     S = (p * (p - a) * (p - b) * (p - c)) ** 0.5
9     print('三角形的周长为: ', C)
10    print('三角形的面积为: ', S)
11 else:
12    print('无法构成三角形!')
```

输出结果：

```
1 输入三角形的第1条边: 1
2 输入三角形的第2条边: 2
3 输入三角形的第3条边: 3
4 无法构成三角形!
```

```
1 输入三角形的第1条边: 3
2 输入三角形的第2条边: 4
3 输入三角形的第3条边: 5
4 三角形的周长为: 12
5 三角形的面积为: 6.0
```

第三题

源代码：

```
1 w = eval(input('输入第1个数: '))
2 x = eval(input('输入第2个数: '))
3 y = eval(input('输入第3个数: '))
4 z = eval(input('输入第4个数: '))
5
6 i = 0
7 j = 0
8 arr = [w, x, y, z]
9
10 #冒泡排序
11 while i < 4:
12     j = 0
13     while j < 4-i-1:
14         if arr[j] < arr[j+1]:
15             arr[j], arr[j+1] = arr[j+1], arr[j]
16         j = j+1
17     i = i+1
18
19 #输出
20 for k in range(4):
21     print(arr[k], end=' ')
```

输出结果：

```
1 输入第1个数: 1
2 输入第2个数: 2
3 输入第3个数: 3
4 输入第4个数: 4
5 4 3 2 1
```