

贝琪的学习任务【1.21】

任务简介：

这是 20-22 号三天任务的后半部分

同样，考虑到渣澜的经验，布置的任务量会比渣澜的多一些

那今天的任务就是 python 的注释、运算符、字符串操作和列表操作，以及一节算法+5 道题目

Python 的注释：

单行注释：

zhushi

(相当于 c++的 //zhushi

多行注释：

""" zhushi """
''' zhushi '''
或

(相当于 c++的 /* zhushi */

Python 的运算符：

算术运算符：

+, -, *, 这些和 c++一样

%, 取余数符号，需要注意的是 python 的余数 ≥ 0

/, 在 python 里 '/' 代表带小数除法，也就是说， $21 / 10 = 2.1$

//, 这是取整除符号, 与 c++ 的 '/' 一样, $21 // 10 = 2$

******, 乘方符号, $2**3 = 2$ 的 3 次方 = 8

比较运算符:

=、**!=**、**>**、**<**、**>=**、**<=**, 和 c++ 一样

赋值运算符:

=、**+=**、**-=**、***=**、**/=**、**%=**、****=**、**//=**, 效果和 c++ 差不多

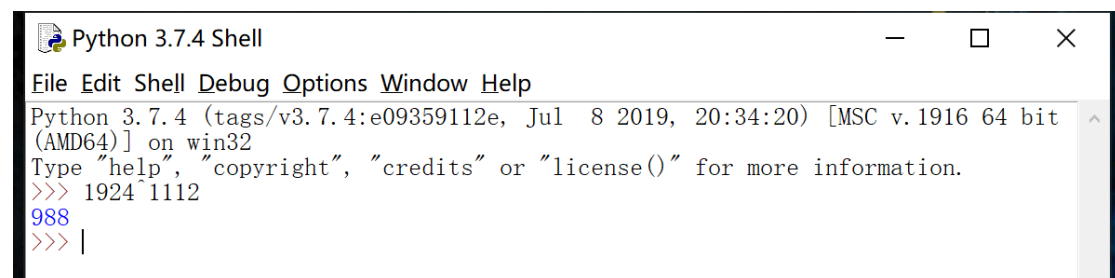
(例如 $x ** = y$ 等价于 $x = x ** y$)

位运算符:

&、**|**、**^**、**~**、**<<**、**>>**, 与、或、异或、取反、左右移位

一个有趣的东西:

Python 交互式编程的一个好玩的地方, 就是它直接可以当高级计算器来用, 比如说直接输入 1924^{1112} :



```
Python 3.7.4 Shell
File Edit Shell Debug Options Window Help
Python 3.7.4 (tags/v3.7.4:e09359112e, Jul 8 2019, 20:34:20) [MSC v.1916 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>> 1924^1112
988
>>> |
```

(对这三个数字还有印象吗)

Python 字符串声明:

之前提过, python 会自动识别变量类型

所以想声明一个字符串的话, 只需要写下面两者之一:

```
a = 'zhalan'
a = "zhalan"
```

在 python 中，单引号和双引号都可以声明字符串，但是不能一边单引号一边双引号

一些常见的字符串操作：

截取操作：

a[0]: 返回 a 的下标为 0 的字符'z'（和 c++一样

a[0:3]: 返回 a 的[0,3)的子字符串'zha'（和 a.substr(0,3)类似，但是 python 是返回[0,3)左闭右开区间的子字符串，c++是返回从 0 开始的长度为 3 的子字符串

a[0:5:2]: 先得到 a 的[0,5)子字符串，然后每隔 2-1=1 个截取 1 个字符，拼接起来得到最终的字符串'zaa'（也就是从 a[0,5)中截取 a[0]+a[2]+a[4]形成'zaa'

总结：[]括号内可以有 1、2、3 个参数，参数之间用 : 隔开

一个参数的话 [i]，返回下标为 i 的字符；

两个参数的话 [i:j]，截取[i,j)区间，第一个参数可以不填，默认值为 0，第二个参数可以不填，默认为字符串的末尾；如 a[:3]代表[0,3)，a[3:]代表从 3 开始到最后的子串，a[:]代表整个 a

三个参数的话 [i:j:k]，先截取[i,j)区间，然后每隔 k-1 个截取 1 个字符拼接形成最后的字符串。3 个参数都可以不填，前两个同上，第三个默认值为 1

运算符操作:

```
Python 3.7.4 (tags/v3.7.4:e09359112e, Jul 8 2019, 20:34:20) [MSC v.1916 64 bit  
(AMD64)] on win32  
Type "help", "copyright", "credits" or "license()" for more information.  
>>> a = 'zha'  
>>> b = 'lan'  
>>> a+b  
'zhalan'  
>>> a*3  
'zhazhazha'  
>>> a+b*2  
'zhalanlan'  
>>> (a+b)*3  
'zhalanzhalanzhalan'
```

a+b: 把 b 拼接在 a 后面

a*3: 等价于 a+a+a

运算符的优先级和普通计算一样, 可以用小括号提升优先级

还有一些不太常见的操作, 可以看一下后面的菜鸟教程

Python 列表声明:

列表的声明, 可以用一对中括号搞定:

```
a = [1, 2.2, 'zha']
```

列表类似于 vector, 但是里面的数据可以是不同类型的

一些常见的列表操作:

截取操作: 与字符串相同, 如 a[1]返回 2.2, a[0:2]返回[1,2.2],

a[:2]返回[1,'zha'], 等等

运算符操作: +和*也和字符串相同

```
>>> a = [1, 2.2, 'zha']  
>>> b = [True, 'lan']  
>>> a+b  
[1, 2.2, 'zha', True, 'lan']  
>>> a*2  
[1, 2.2, 'zha', 1, 2.2, 'zha']
```

插入和删除：

插入有 `append()`和 `insert()`函数，如 `a.append('z')`把字符'z'插入列表 `a` 的末端，`a.insert(3,b)`把列表 `b` 插入到 `a[3]`前面

```
>>> a.append('z')
>>> a
[1, 2.2, 'zha', 'z']
>>> a.insert(3,b)
>>> a
[1, 2.2, 'zha', [True, 'lan'], 'z']
```

删除有 `pop()`和 `remove()`函数，哎打字好累直接上图了：

```
list.pop([index=-1])
移除列表中的一个元素（默认最后一个元素），并且返回该元素的值

list.remove(obj)
移除列表中某个值的第一个匹配项
```

另外，`del a[2]`也可以删除 `a[2]`位置的值，后面的数据前移一位

算法：

今天主要学排序

我们先来看一道简单的排序题：

题目描述

[展开](#)

利用快速排序算法将读入的 N 个数从小到大排序后输出。

快速排序是信息学竞赛的必备算法之一。对于快速排序不是很了解的同学可以自行上网查询相关资料，掌握后独立完成。（`C++` 选手请不要试图使用 `STL`，虽然你可以使用 `sort` 一遍过，但是你并没有掌握快速排序算法的精髓。）

输入格式

第 1 行为一个正整数 N ，第 2 行包含 N 个空格隔开的正整数 a_i ，为你需要进行排序的数，数据保证了 A_i 不超过 10^9 。

输出格式

将给定的 N 个数从小到大输出，数之间空格隔开，行末换行且无空格。

请不要试图使用 sort（我偏不
所以

```
#include <iostream>
#include <algorithm>
using namespace std;
int main()
{
    int n;
    cin>>n;
    int a[n];
    for(int i=0;i<n;i++) cin>>a[i];
    sort(a,a+n);
    for(int i=0;i<n;i++) cout<<a[i]<<" ";
}
```

就可以了（两下子也接受不了吧

学习网站：

Python: <https://www.runoob.com/python3/python3-intro.html>



RUNOOB.COM 搜索.....

首页 HTML CSS JAVASCRIPT JQUERY BOOTSTRAP PYTHON3 PYTHON2 JAVA C C++ C# SQL MYSQL PHP 本地书签

Python 3 教程 Python3 简介 Python3 环境搭建 Python3 基础语法 Python3 基本数据类型 Python3 解释器 Python3 注释 Python3 运算符 Python3 数字(Number) Python3 字符串 Python3 列表

Python3 简介 →

Python 3 教程

 Python 的 3.0 版本，常被称为 Python 3000，或简称 Py3k。相对于 Python 的早期版本，这是一个较大的升级。为了不带来过多的累赘，Python 3.0 在设计的时候没有考虑向下兼容。

 Python 介绍及安装教程我们在 [Python 2.X 版本的教程](#) 中已有介绍，这里就不再赘述。

[你也可以点击 Python2.x 与 3.x 版本区别](#) 来查看两者的不同。

本教程主要针对 Python 3.x 版本的学习，如果你使用的是 Python 2.x 版本请移步至 [Python 2.X 版本的教程](#)。

官方宣布，2020 年 1 月 1 日，停止 Python 2 的更新。

查看 Python 版本

我们可以在命令窗口(Windows 使用 win+R 调出 cmd 运行框)使用以下命令查看我们使用的 Python 版本：

看一下左边的 python3 注释、运算符、字符串、列表吧。

任务：

Python：

1.在脚本式编程下编写代码，实现如下操作：

输入：两个字符串；输出：两个字符串的拼接

2.打开文件 1.21.py 并完成其中的要求

（打开方式：右键该文件，点击 Edit with IDLE

提交：把 2 道题的代码/修改后的文件截图发给姐就好啦

算法：完成洛谷题目 1059、1068、1540、1603、1012

1059 明明的随机数：入门级难度，要求用 `sort()` 排序；

1068 分数线划定：也是排序思路，可以学一下 `sort` 排序结构体

1781 宇宙总统：高精度整数，或者不用高精度？

1603 斯诺登的密码：模拟+排序，ok

1012 拼数：更加有趣的比较方式（其实不太好想也不太好理解

提交：把 5 道题的代码和 AC 截图发给姐就好啦

ddl：22 号晚 11.