

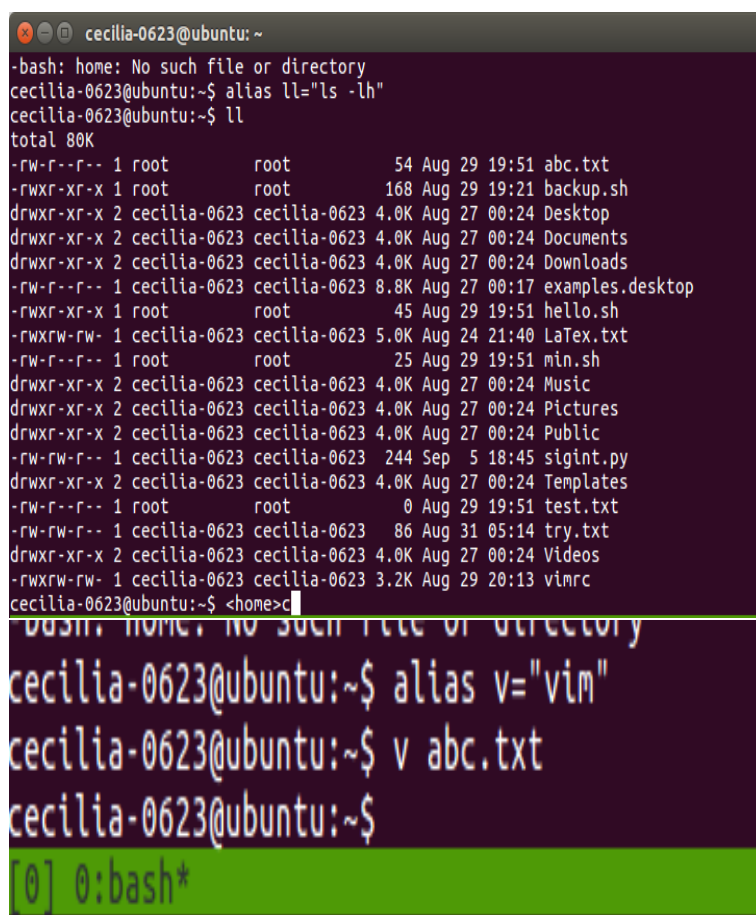
第三周实验报告

张津侨

2024 年 9 月 7 日

1 练习内容及结果

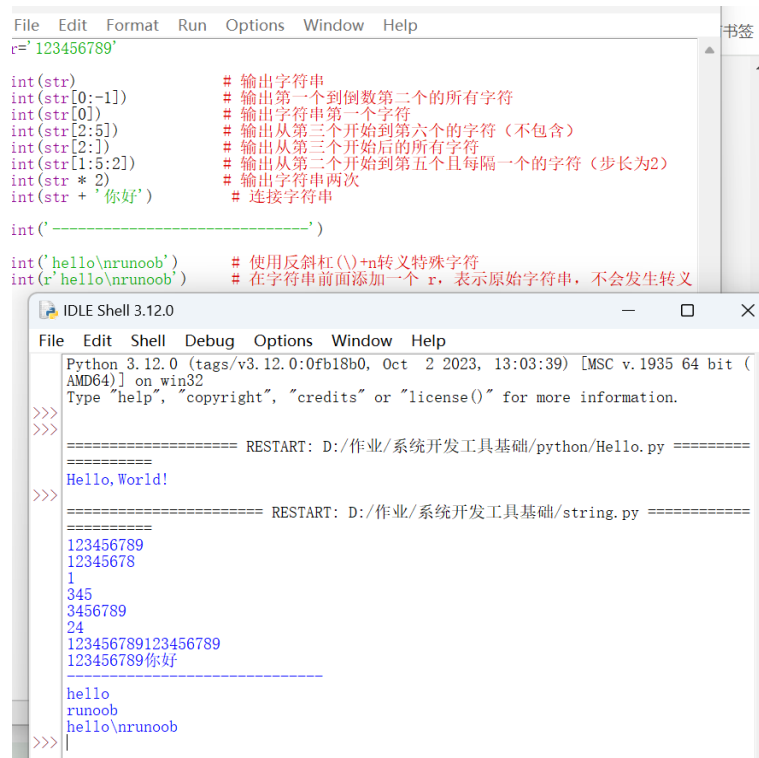
1.1 命令行环境-别名



```
cecilia-0623@ubuntu: ~  
-bash: home: No such file or directory  
cecilia-0623@ubuntu:~$ alias ll="ls -lh"  
cecilia-0623@ubuntu:~$ ll  
total 80K  
-rw-r--r-- 1 root      root          54 Aug 29 19:51 abc.txt  
-rwxr-xr-x 1 root      root         168 Aug 29 19:21 backup.sh  
drwxr-xr-x 2 cecilia-0623 cecilia-0623 4.0K Aug 27 00:24 Desktop  
drwxr-xr-x 2 cecilia-0623 cecilia-0623 4.0K Aug 27 00:24 Documents  
drwxr-xr-x 2 cecilia-0623 cecilia-0623 4.0K Aug 27 00:24 Downloads  
-rw-r--r-- 1 cecilia-0623 cecilia-0623 8.8K Aug 27 00:17 examples.desktop  
-rwxr-xr-x 1 root      root          45 Aug 29 19:51 hello.sh  
-rwxrwxrwx 1 cecilia-0623 cecilia-0623 5.0K Aug 24 21:40 LaTeX.txt  
-rw-r--r-- 1 root      root          25 Aug 29 19:51 min.sh  
drwxr-xr-x 2 cecilia-0623 cecilia-0623 4.0K Aug 27 00:24 Music  
drwxr-xr-x 2 cecilia-0623 cecilia-0623 4.0K Aug 27 00:24 Pictures  
drwxr-xr-x 2 cecilia-0623 cecilia-0623 4.0K Aug 27 00:24 Public  
-rw-rw-r-- 1 cecilia-0623 cecilia-0623 244 Sep  5 18:45 sigint.py  
drwxr-xr-x 2 cecilia-0623 cecilia-0623 4.0K Aug 27 00:24 Templates  
-rw-r--r-- 1 root      root           0 Aug 29 19:51 test.txt  
-rw-rw-r-- 1 cecilia-0623 cecilia-0623  86 Aug 31 05:14 try.txt  
drwxr-xr-x 2 cecilia-0623 cecilia-0623 4.0K Aug 27 00:24 Videos  
-rwxrwxrwx 1 cecilia-0623 cecilia-0623 3.2K Aug 29 20:13 vimrc  
cecilia-0623@ubuntu:~$ <home>c  
-bash: home: No such file or directory  
cecilia-0623@ubuntu:~$ alias v="vim"  
cecilia-0623@ubuntu:~$ v abc.txt  
cecilia-0623@ubuntu:~$  
[0] 0:bash*
```

alias + 别名 = “命令”

1.2 python 中字符串的运用



```
File Edit Format Run Options Window Help
r='123456789'

int(str)           # 输出字符串
int(str[0:-1])     # 输出第一个到倒数第二个的所有字符
int(str[0])         # 输出字符串第一个字符
int(str[2:5])       # 输出从第三个开始到第六个的字符（不包含）
int(str[2:])        # 输出从第三个开始后的所有字符
int(str[1:5:2])     # 输出从第二个开始到第五个且每隔一个的字符（步长为2）
int(str * 2)        # 输出字符串两次
int(str + '你好')   # 连接字符串

int('-----')

int('hello\nrunoob') # 使用反斜杠(\)+n转义特殊字符
int(r'hello\nrunoob') # 在字符串前面添加一个 r，表示原始字符串，不会发生转义

IDLE Shell 3.12.0
File Edit Shell Debug Options Window Help
Python 3.12.0 (tags/v3.12.0:0fb18b0, Oct 2 2023, 13:03:39) [MSC v.1935 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
>>>
===== RESTART: D:/作业/系统开发工具基础/python/Hello.py =====
Hello,World!
>>>
===== RESTART: D:/作业/系统开发工具基础/string.py =====
123456789
12345678
1
345
3456789
24
123456789123456789
123456789你好
-----
hello
runoob
hello\nrunoob
>>>
```

读取字符串，设置起始位置、终止位置的后一位以及步长，
连接字符串，防止转义的发生

1.3 引入 sys 模块，设置命令行参数

```
import sys
print('====Python import mode====')
print('命令行参数为:')
for i in sys.argv:
    print(i)
print('\n python 路径为', sys.path)
```

Ln: 7 Col: 0

```
====Python import mode=====
命令行参数为:
D:/作业/系统开发工具基础/sys.py

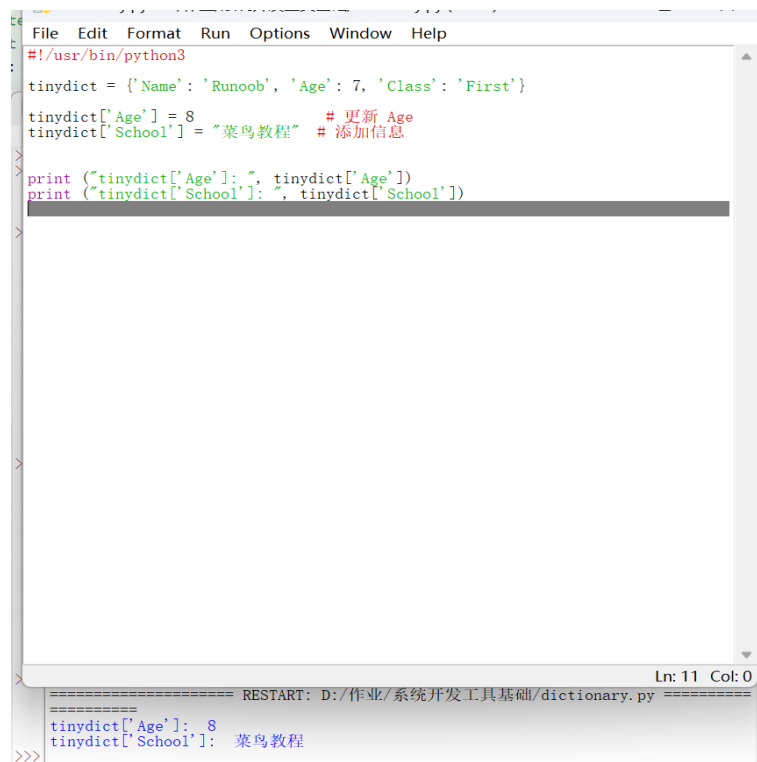
python 路径为 ['D:/作业/系统开发工具基础', 'C:\\Users\\ZJQ\\AppData\\Local\\Programs\\Python\\Python312\\Lib\\idlelib', 'C:\\Users\\ZJQ\\AppData\\Local\\Programs\\Python\\Python312\\python312.zip', 'C:\\Users\\ZJQ\\AppData\\Local\\Programs\\Python\\Python312\\DLLs', 'C:\\Users\\ZJQ\\AppData\\Local\\Programs\\Python\\Python312\\Lib', 'C:\\Users\\ZJQ\\AppData\\Local\\Programs\\Python\\Python312\\site-packages']
>
```

```
python 路径为 ['D:/作业/系统开发工具基础/python', 'C:\\Users\\ZJQ\\AppData\\Local\\Programs\\Python\\Python312\\python312.zip', 'C:\\Users\\ZJQ\\AppData\\Local\\Programs\\Python\\Python312\\DLLs', 'C:\\Users\\ZJQ\\AppData\\Local\\Programs\\Python\\Python312\\Lib', 'C:\\Users\\ZJQ\\AppData\\Local\\Programs\\Python\\Python312\\site-packages']
D:/作业/系统开发工具基础/python>python sys.py 111
====Python import mode=====
sys.py
111

python 路径为 ['D:/作业/系统开发工具基础/python', 'C:\\Users\\ZJQ\\AppData\\Local\\Programs\\Python\\Python312\\python312.zip', 'C:\\Users\\ZJQ\\AppData\\Local\\Programs\\Python\\Python312\\DLLs', 'C:\\Users\\ZJQ\\AppData\\Local\\Programs\\Python\\Python312\\Lib', 'C:\\Users\\ZJQ\\AppData\\Local\\Programs\\Python\\Python312\\site-packages']
D:/作业/系统开发工具基础/python>python sys.py 111 87998
====Python import mode=====
sys.py
111
87998

python 路径为 ['D:/作业/系统开发工具基础/python', 'C:\\Users\\ZJQ\\AppData\\Local\\Programs\\Python\\Python312\\python312.zip', 'C:\\Users\\ZJQ\\AppData\\Local\\Programs\\Python\\Python312\\DLLs', 'C:\\Users\\ZJQ\\AppData\\Local\\Programs\\Python\\Python312\\Lib', 'C:\\Users\\ZJQ\\AppData\\Local\\Programs\\Python\\Python312\\site-packages']
D:/作业/系统开发工具基础/python>
```

1.4 python 中字典的运用

A screenshot of a Python IDE window. The menu bar includes File, Edit, Format, Run, Options, Window, and Help. The code editor contains the following Python code:

```
#!/usr/bin/python3

tinydict = {'Name': 'Runoob', 'Age': 7, 'Class': 'First'}

tinydict['Age'] = 8 # 更新 Age
tinydict['School'] = "菜鸟教程" # 添加信息

print ("tinydict['Age']: ", tinydict['Age'])
print ("tinydict['School']: ", tinydict['School'])
```

The status bar at the bottom right shows "Ln: 11 Col: 0". Below the code editor, a console window displays the output of the code:

```
===== RESTART: D:/作业/系统开发工具基础/dictionary.py =====
tinydict['Age']: 8
tinydict['School']: 菜鸟教程
>>>
```

创建字典，修改字典中内容，打印字典内容

1.6 用 python 进行冒泡排序

```
>>>
===== RESTART: D:/作业/系统开发工具基础/python/rank.py =====
排序后的数组: [11, 11, 12, 22, 25, 30, 34, 64, 90]
>>>
===== RESTART: D:/作业/系统开发工具基础/python/fblq.py =====

rank.py - D:\作业\系统开发工具基础\python\rank.py (3.12.0)
File Edit Format Run Options Window Help
def bubble_sort(arr):
    n = len(arr)
    for i in range(n):
        for j in range(0, n-i-1):
            if arr[j] > arr[j+1]:
                arr[j], arr[j+1] = arr[j+1], arr[j]
    return arr

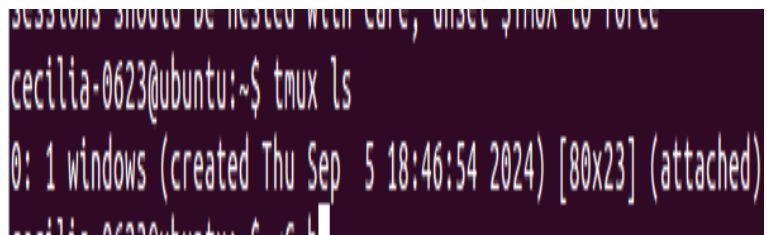
numbers = [64, 34, 25, 12, 22, 11, 90, 30, 11]
sorted_numbers = bubble_sort(numbers)
print("排序后的数组:", sorted_numbers)
```

1.7 终端多路复用：用 tmux 开始新的会话

```
cecilia-0623@ubuntu: ~
cecilia-0623@ubuntu:~$ tmux new -s work
[exited]
cecilia-0623@ubuntu:~$
```

tmux new -s NAME 以指定名称开始一个新的会话

1.8 列出当前所有会话

A terminal window with a dark background and light blue text. The prompt is 'cecilia-0623@ubuntu:~\$'. The command 'tmux ls' has been entered. The output is '0: 1 windows (created Thu Sep 5 18:46:54 2024) [80x23] (attached)'. The cursor is at the end of the output line.

```
cecilia-0623@ubuntu:~$ tmux ls
0: 1 windows (created Thu Sep 5 18:46:54 2024) [80x23] (attached)
```

tmux ls 列出当前所有会话

1.9 远端设备

```
ZJQ@LAPTOP-BSMDJD8U MINGW64 ~/Desktop (master)
$ ssh-keygen
Generating public/private ed25519 key pair.
Enter file in which to save the key (/c/Users/ZJQ/.ssh/id_ed25519):
Created directory '/c/Users/ZJQ/.ssh'.
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /c/Users/ZJQ/.ssh/id_ed25519
Your public key has been saved in /c/Users/ZJQ/.ssh/id_ed25519.pub
The key fingerprint is:
SHA256:jihoBbnzD1U//FyIT8qS1vHoc6wi9ZDb3xzj3TFDIY8 ZJQ@LAPTOP-BSMDJD8U
The key's randomart image is:
--[ED25519 256]--+
.
o . . . .
o . o . . + .
o o *So . E o
.= .o@ . .
.o...B.= o +
. o.o.B. .+ + .+
o...=o.+ . .
-----[SHA256]-----
```

```
ZJQ@LAPTOP-BSMDJD8U MINGW64 ~/Desktop (master)
$ ssh -T git@github.com
Hi Cecilia-hub! You've successfully authenticated,
hell access.
```

```
ZJQ@LAPTOP-BSMDJD8U MINGW64 ~/Desktop/git.test01 (master)
$ git remote add origin git@github.com:Cecilia-hub/class.git

ZJQ@LAPTOP-BSMDJD8U MINGW64 ~/Desktop/git.test01 (master)
$ git remote
origin
```

秘钥连接：使用 `ssh-keygen` 命令可以生成一对密钥：

1.10 配置文件

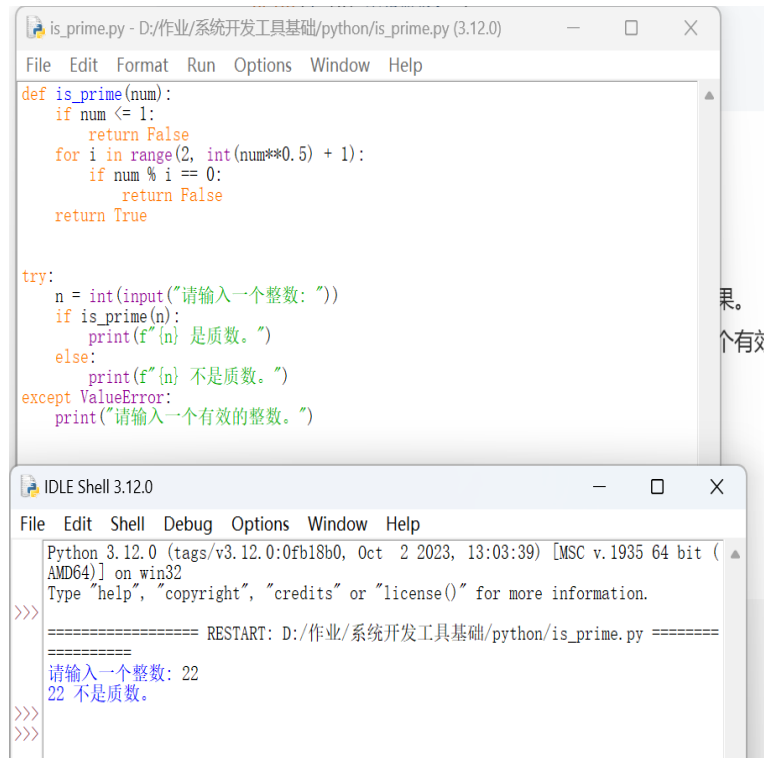
```
cecilia-0623@ubuntu:~$ mkdir ~/gits/dotfiles
cecilia-0623@ubuntu:~$ git init ~/gits/dotfiles
cecilia-0623@ubuntu:~$ ls -a ~/gits/dotfiles
.  .. .git
cecilia-0623@ubuntu:~$ .  .. .bashrc .git .profile .tmux.conf .vimrc .zshrc
```

gits 目录是创建用来存放所有 git 及 github 仓库的目录
将本机的配置文件，如.vimrc/.bashrc/.tmux.conf 等复制进该目录
其中，”.” 分别表示本目录及上级目录，”.git” 为 git 仓库的配置文件，其
他文件为存放在仓库中的系统配置文件

1.11 用 python 读取文件

```
File Edit Format Run Options Window Help
def read_file(file_path):
    with open(file_path, 'r') as file:
        content = file.read()
    return content
```

1.12 用 python 判断质数



The image shows a Python IDE window titled 'is_prime.py - D:/作业/系统开发工具基础/python/is_prime.py (3.12.0)'. The code defines a function `is_prime(num)` that checks if a number is prime. It returns `False` for numbers less than or equal to 1, and for numbers greater than 1, it checks divisibility by integers from 2 to $\sqrt{\text{num}}$. If any divisor is found, it returns `False`; otherwise, it returns `True`. The script then prompts the user to enter an integer, checks if it's prime, and prints the result. It also includes an exception handler for `ValueError` to prompt the user for a valid integer.

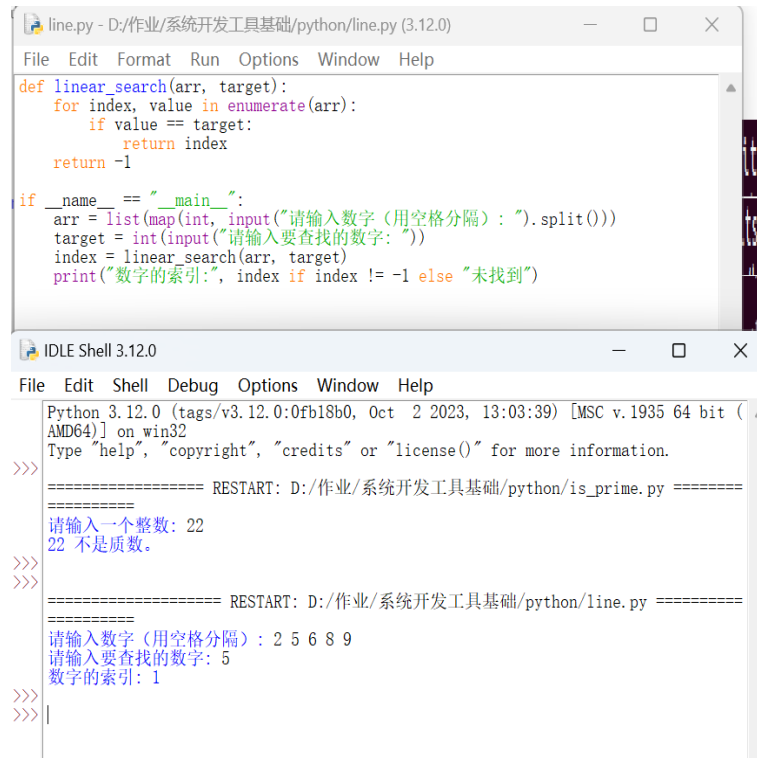
```
def is_prime(num):
    if num <= 1:
        return False
    for i in range(2, int(num**0.5) + 1):
        if num % i == 0:
            return False
    return True

try:
    n = int(input("请输入一个整数: "))
    if is_prime(n):
        print(f"{n} 是质数。")
    else:
        print(f"{n} 不是质数。")
except ValueError:
    print("请输入一个有效的整数。")
```

The IDE Shell window shows the execution output. It displays the Python version and architecture, followed by a restart message. The user enters '22', and the program outputs '22 不是质数。'.

```
Python 3.12.0 (tags/v3.12.0:0fb18b0, Oct 2 2023, 13:03:39) [MSC v.1935 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: D:/作业/系统开发工具基础/python/is_prime.py =====
请输入一个整数: 22
22 不是质数。
>>>
>>>
```

1.13 用 python 进行线性查找



The image shows a screenshot of a Python IDE with two windows. The top window, titled 'line.py - D:/作业/系统开发工具基础/python/line.py (3.12.0)', contains the following Python code:

```
def linear_search(arr, target):
    for index, value in enumerate(arr):
        if value == target:
            return index
    return -1

if __name__ == "__main__":
    arr = list(map(int, input("请输入数字（用空格分隔）: ").split()))
    target = int(input("请输入要查找的数字: "))
    index = linear_search(arr, target)
    print("数字的索引:", index if index != -1 else "未找到")
```

The bottom window, titled 'IDLE Shell 3.12.0', shows the execution of the program. It displays the Python version and architecture, followed by a restart message for 'is_prime.py'. The user enters '22' and is informed that it is not a prime number. After another restart, the user enters the array '2 5 6 8 9' and the target '5', and the program correctly outputs the index '1'.

```
Python 3.12.0 (tags/v3.12.0:0fb18b0, Oct 2 2023, 13:03:39) [MSC v.1935 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: D:/作业/系统开发工具基础/python/is_prime.py =====
>>>
请输入一个整数: 22
22 不是质数。
>>>
===== RESTART: D:/作业/系统开发工具基础/python/line.py =====
>>>
请输入数字（用空格分隔）: 2 5 6 8 9
请输入要查找的数字: 5
数字的索引: 1
>>>
>>> |
```

1.14 用 python 统计字符出现次数

```
=====
请输入字符串: hcuaiebcanaaaaaajik
字符出现次数: {'h': 1, 'c': 2, 'u': 1, 'i': 2, 'a': 7, 'e': 1, 'b': 1, 'n': 1, 'j': 1, 'k': 1}
>>>

count.py - D:/作业/系统开发工具基础/python/count.py (3.12.0)
File Edit Format Run Options Window Help

def char_count(s):
    count = {}
    for char in s:
        count[char] = count.get(char, 0) + 1
    return count

if __name__ == "__main__":
    s = input("请输入字符串: ")
    counts = char_count(s)
    print("字符出现次数:", counts)
```

1.15 用 python 进行二分查找

```
def binary_search(arr, target):
    left, right = 0, len(arr) - 1
    while left <= right:
        mid = left + (right - left) // 2
        if arr[mid] == target:
            return mid
        elif arr[mid] < target:
            left = mid + 1
        else:
            right = mid - 1
    return -1

if __name__ == "__main__":
    arr = list(map(int, input("请输入已排序的数字（用空格分隔）：").split()))
    target = int(input("请输入要查找的数字："))
    index = binary_search(arr, target)
    print("数字的索引：", index if index != -1 else "未找到")
```

Ln: 18 Col: 6

===== RESTART: D:/作业/系统开发工具基础/python/two.py =====

请输入已排序的数字（用空格分隔）： 4 6 7 9 11 35 67
请输入要查找的数字： 7
数字的索引： 2

>>>

1.16 用 python 计算平均值

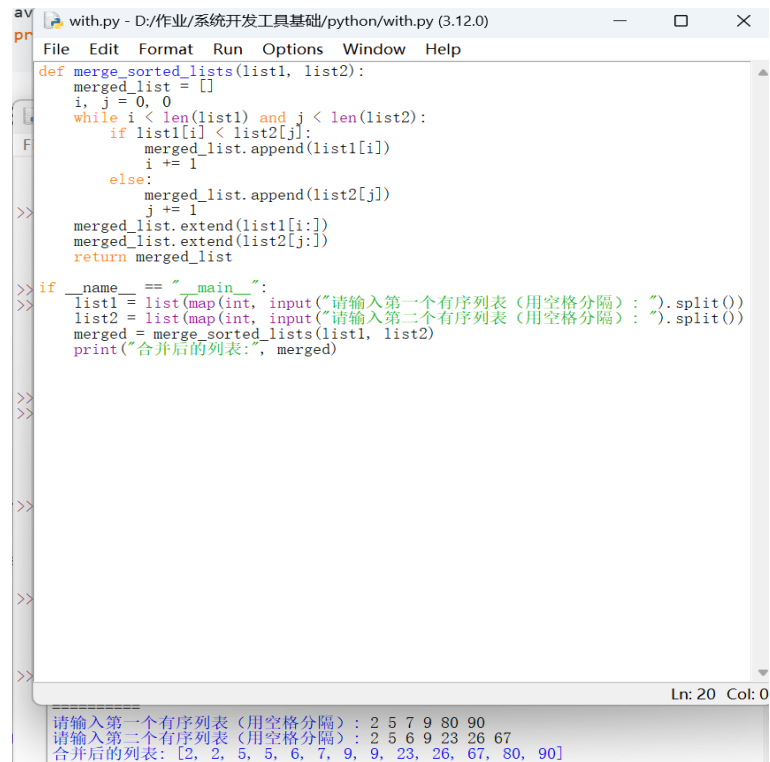
```
-----
请输入数字（用空格分隔）： 67 89 90 76
平均值： 80.5
>>>

average.py - D:/作业/系统开发工具基础/python/average.py (3.12.0)
File Edit Format Run Options Window Help

def calculate_average(arr):
    if len(arr) == 0:
        return 0
    return sum(arr) / len(arr)

if __name__ == "__main__":
    arr = list(map(float, input("请输入数字（用空格分隔）：").split()))
    average = calculate_average(arr)
    print("平均值:", average)
```

1.17 用 python 合并两个有序列表



```
with.py - D:/作业/系统开发工具基础/python/with.py (3.12.0)
File Edit Format Run Options Window Help

def merge_sorted_lists(list1, list2):
    merged_list = []
    i, j = 0, 0
    while i < len(list1) and j < len(list2):
        if list1[i] < list2[j]:
            merged_list.append(list1[i])
            i += 1
        else:
            merged_list.append(list2[j])
            j += 1
    merged_list.extend(list1[i:])
    merged_list.extend(list2[j:])
    return merged_list

if __name__ == "__main__":
    list1 = list(map(int, input("请输入第一个有序列表 (用空格分隔): ").split()))
    list2 = list(map(int, input("请输入第二个有序列表 (用空格分隔): ").split()))
    merged = merge_sorted_lists(list1, list2)
    print("合并后的列表:", merged)
```

Ln: 20 Col: 0

请输入第一个有序列表 (用空格分隔): 2 5 7 9 80 90
请输入第二个有序列表 (用空格分隔): 2 5 6 9 23 26 67
合并后的列表: [2, 2, 5, 5, 6, 7, 9, 9, 23, 26, 67, 80, 90]

1.18 用 python 解决八皇后问题

```

queens.py - D:/作业/系统开发工具基础/python/queens.py (3.12.0)
File Edit Format Run Options Window Help

def is_safe(board, row, col):
    # 检查当前列
    for i in range(row):
        if board[i][col] == 'Q':
            return False

    # 检查左上对角线
    for i, j in zip(range(row, -1, -1), range(col, -1, -1)):
        if j < 0:
            break
        if board[i][j] == 'Q':
            return False

    # 检查右上对角线
    for i, j in zip(range(row, -1, -1), range(col, len(board))):
        if j == len(board):
            break
        if board[i][j] == 'Q':
            return False

    return True

def solve_n_queens_util(board, row):
    if row >= len(board):
        print_board(board)
        return

    for col in range(len(board)):
        if is_safe(board, row, col):
            board[row][col] = 'Q' # 放置皇后
            solve_n_queens_util(board, row + 1) # 递归放置下一个皇后
            board[row][col] = '.' # 回溯，移除皇后

def solve_n_queens(n):
    board = [['.' for _ in range(n)] for _ in range(n)]
    solve_n_queens_util(board, 0)

if __name__ == '__main__':
    n = 8 # 八皇后
    solve_n_queens(n)

```

Ln: 45 Co

1.19 下载 pillow 模块

```
C:\Windows\system32\cmd.exe X + v
Microsoft Windows [版本 10.0.22621.2428]
(c) Microsoft Corporation. 保留所有权利。

C:\Users\ZJQ>pip install Pillow
Collecting Pillow
  Obtaining dependency information for Pillow from https://files.pythonhosted.org/packages/74/0a/d4ce3c44bca8635bd29a2ea
b5aa181b654a734a29b263ca8efe013bee98/pillow-10.4.0-cp312-cp312-win_amd64.whl.metadata
  Downloading pillow-10.4.0-cp312-cp312-win_amd64.whl.metadata (9.3 kB)
  Downloading pillow-10.4.0-cp312-cp312-win_amd64.whl (2.6 MB)
    2.6/2.6 MB 5.4 MB/s eta 0:00:00
Installing collected packages: Pillow
Successfully installed Pillow-10.4.0

[notice] A new release of pip is available: 23.2.1 -> 24.2
[notice] To update, run: python.exe -m pip install --upgrade pip

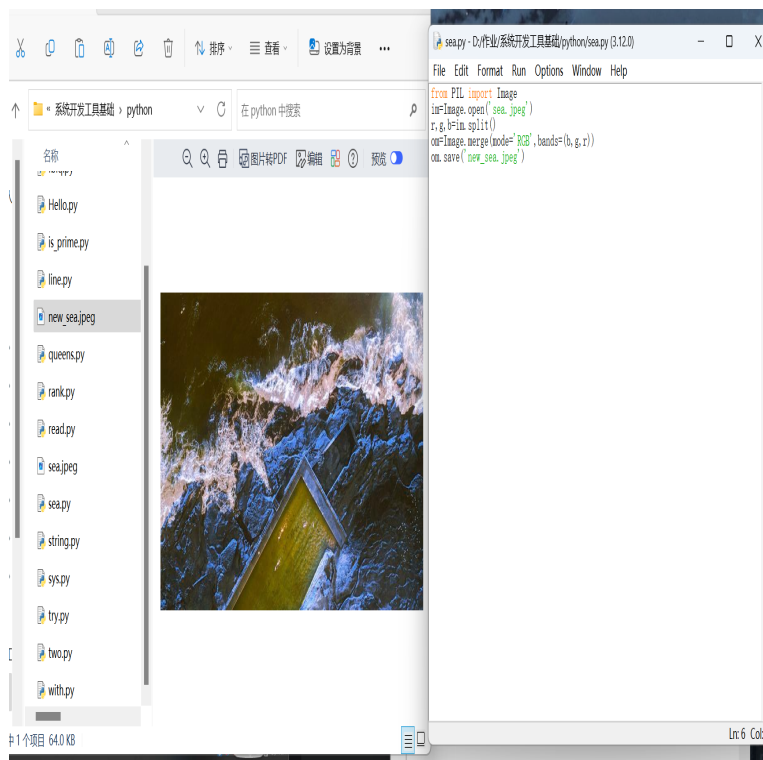
C:\Users\ZJQ>pip list
Package Version
-----
pillow  10.4.0
pip     23.2.1

[notice] A new release of pip is available: 23.2.1 -> 24.2
[notice] To update, run: python.exe -m pip install --upgrade pip

C:\Users\ZJQ>
```

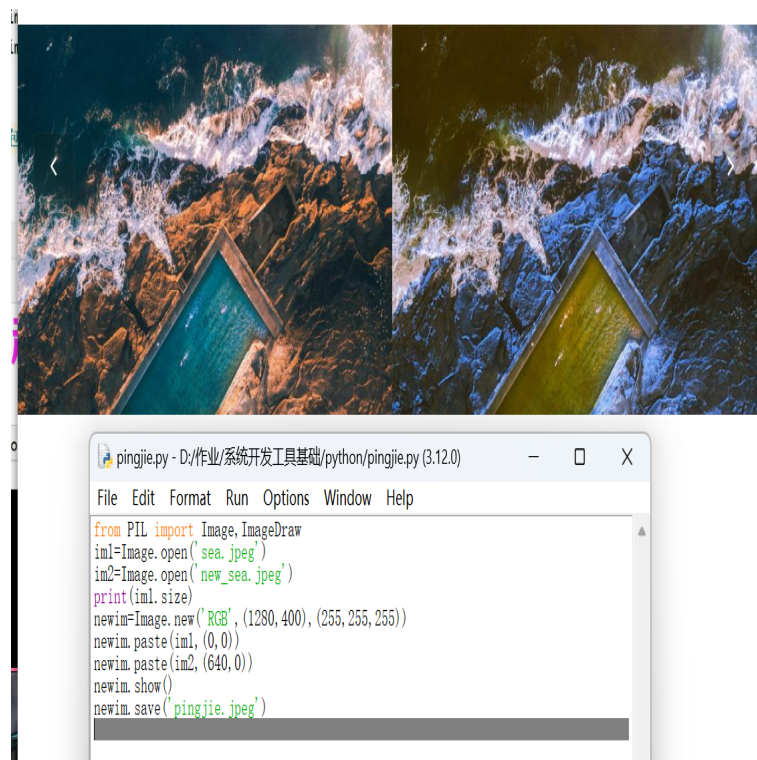
pip install Pillow

1.20 PIL 库-图像的颜色交换



加载图片，提取 RGB 图像的颜色通道
合并通道，其中 mode 表示色彩，bands 表示的是新的色彩通道
保存新的文件至磁盘

1.21 PIL 库-图片拼接



打印图片尺寸，新建空白图片，
调整颜色参数，拼接并保存图片

2 解题感悟

夏季学期的第三周,我学习了命令行环境,python 入门基础以及 python 视觉运用

命令行环境的学习与之前学习 git, shell 有类似的地方,有一种熟练感
python 与其他编程语言相比其实只是语法上的不同,python 中的列表字典等我也有了初步的了解与应用

python 视觉应用是通过下载 Pillow 得以实现的,给我一种新奇的感觉
虽然对于这几个工具我只是进行了初步学习,但是我的视野得到了拓展,此后有对于相关工具的使用需要时也能更快地上手操作。

3 gitbit 账号链接:<https://github.com/Cecilia-hub>