南京大学本科生实验报告

课程名称：计算机网络 **任课教师：李文中** **助教：**

学院	计算机科学与技术	专业（方向）	计算机科学与技术
学号	191220008	姓名	陈南曈
Email	924690736@qq.com	开始/完成日期	6.13~6.16

一、实验名称

Lab 7: Content Delivery Network

二、实验目的

在本次实验中，我将创建一个简版的内容分发网络（CDN），CDN是代理服务器及其数据中心的地理分布网络。它的功能是通过相对于终端用户在空间上分布服务来提供高可用性和性能。

为了尽量减少访问者和网站服务器之间的距离，CDN在多个地理位置（又称存在点或POP）存储其内容的缓存版本。每个PoP包含多个缓存服务器，负责将内容交付给邻近的访问者。

此外，我们这次还在真实的网络平台OpenNetLab上对自己实现的功能进行了测试。

具体完成的功能有：

- 缓存服务器的逻辑：缓存内容
- DNS服务器的逻辑：寻找最近的缓存服务器

三、实验内容

Task 1: Preparation

Initiate your project with our template.

Task 2: DNS server

Implement the DNS server.

Task 3: Caching server

Implement the caching server.

Task 4: Deployment

Deploy the code to OpenNetLab.

四、实验结果

Task 2: DNS server

1、思路分析

首先，从txt文件中加载每个表项。

DNS server会对接收到的域名在dns表中查找匹配。

- 如果未匹配上，则返回None；
- 如果匹配上了，则反应对应的类型和IP地址。

其中，如果有多个地址，则依次判断与client的距离，去最小距离返回。但如果client无法获得距离，则随机返回一个。

2、核心代码

加载表项：

```
with open(dns_file) as info_file:
            for line in info_file:
                info = line.rsplit() # 切割字符串(从行尾开始切割，以去掉行尾回车）
                if info is None: # 确认不是空行
                    break
                self._dns_table.append(Entry(info[0], info[1], info[2:]))
        for entry in self._dns_table:
            print(entry.domain_name, entry.record_type, entry.record_values)
```

表项Entry类：

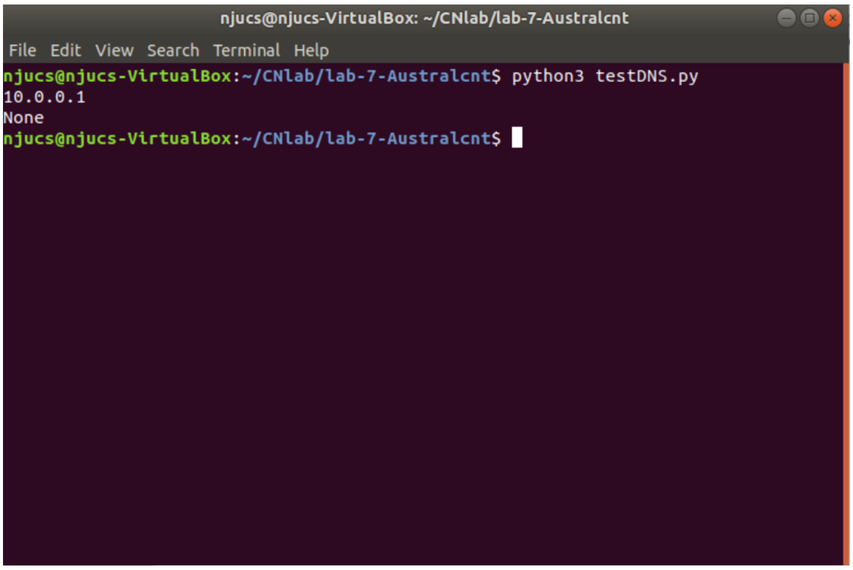
```
class Entry:
    def __init__(self, domain_name, record_type, record_values):
        self.domain_name = domain_name
        self.record_type = record_type
        self.record_values = record_values
```

回应

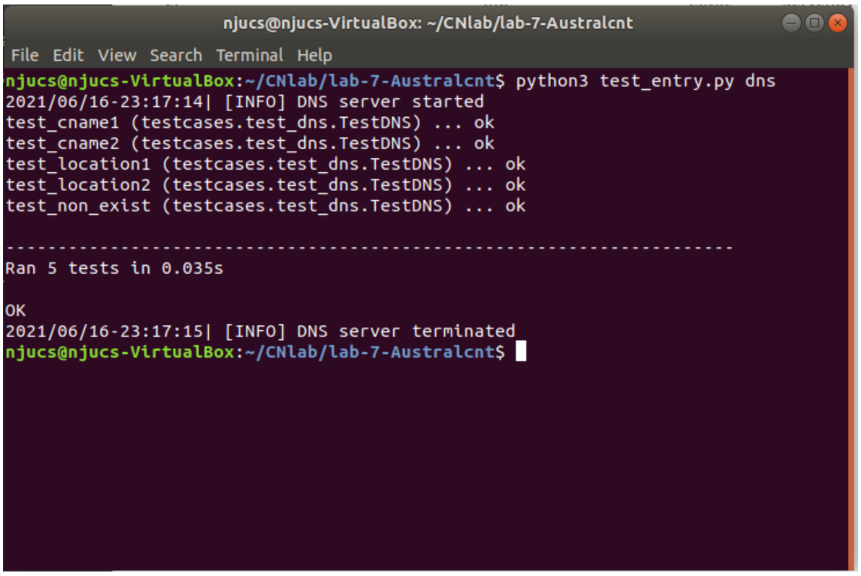
```
def get_response(self, request_domain_name):
        response_type, response_val = None, None
        # ------------------------------------------------
        # TODO: your codes here.
        # Determine an IP to response according to the client's IP address.
        #       set "response_ip" to "the best IP address".
        client_ip, _ = self.client_address
        entry = None
        for cur_entry in self.table:
            name1 = request_domain_name
            name2 = cur_entry.domain_name
            if name1[-1] == '.':
                name1 = name1[:len(name1) - 1]
            if name2[-1] == '.':
                name2 = name2[:len(name2) - 1]
            #print(name1, name2)
            if fnmatch.fnmatch(name1, name2):
                entry = cur_entry
                #print(entry.domain_name, entry.record_type,
entry.record_values)
                break
        if entry == None: # 未匹配到表项
            pass
        else:
            index = 0
            if entry.record_type == 'CNAME': # 类型为CNAME
                pass
            else: # 类型为A
                if len(entry.record_values) == 1: # 长度为1
                    pass
                else: # 长度大于1
                    client_x, client_y = IP_Utils.getIpLocation(client_ip)
                    if client_x is None or client_y is None:
                        index = randint(0, len(entry.record_values) - 1)
                    else:
                        entry_x, entry_y =
IP_Utils.getIpLocation(entry.record_values[0])
                        for i in range(len(entry.record_values)):
                            cur_entry_x, cur_entry_y =
IP_Utils.getIpLocation(entry.record_values[i])
                            if cur_entry_x is None or cur_entry_y is None:
                                continue
                            if (entry_x is None or entry_y is None) or
math.fabs(cur_entry_x - client_x) ** 2 + math.fabs(cur_entry_y - client_y) ** 2
<  math.fabs(entry_x - client_x) ** 2 + math.fabs(entry_y - client_y) ** 2:
                                index = i
            response_type = entry.record_type
            response_val = entry.record_values[index]

        # -------------------------------------------------
        return response_type, response_val
```

匹配时采用fn_match，但要注意最后有'.'的情况，可以先删去，再进行匹配。

3、测试结果


njucs@njucs-VirtualBox: ~/CNlab/lab-7-Australcnt
File Edit View Search Terminal Help

```
njucs@njucs-VirtualBox:~/CNlab/lab-7-Australcnt$ python3 testDNS.py
10.0.0.1
None
njucs@njucs-VirtualBox:~/CNlab/lab-7-Australcnt$ 
```


njucs@njucs-VirtualBox: ~/CNlab/lab-7-Australcnt
File Edit View Search Terminal Help

```
njucs@njucs-VirtualBox:~/CNlab/lab-7-Australcnt$ python3 test_entry.py dns
2021/06/16-23:17:14| [INFO] DNS server started
test_cname1 (testcases.test_dns.TestDNS) ... ok
test_cname2 (testcases.test_dns.TestDNS) ... ok
test_location1 (testcases.test_dns.TestDNS) ... ok
test_location2 (testcases.test_dns.TestDNS) ... ok
test_non_exist (testcases.test_dns.TestDNS) ... ok

----------------------------------------------------------------------
Ran 5 tests in 0.035s

OK
2021/06/16-23:17:15| [INFO] DNS server terminated
njucs@njucs-VirtualBox:~/CNlab/lab-7-Australcnt$ 
```

Task 3: Caching server

1、思路分析

Cacheing server会处理接收到的域名。通过touchItem查找是否存在于cachetable。

- 若在，则返回code OK，headers和body；（DO_HEAD不返回body）
- 若不在，则返回code NOT_FOUND。

在touchItem中，我完成了选做内容。

在cachetable中查找，若找到且未超时，则直接返回headers和body。

若未找到或超时，则向主服务器请求，若未收到回复，则返回None；

若收到回复，则直接返回headers和body（用yield一块一块返回）。

2、核心代码

touchItem：

```
 def touchItem(self, path: str):
        headers, body = None, None
        if path in self.cacheTable and not self.cacheTable.expired(path):
            cache_item = self.cacheTable.get(path)
            headers, body = cache_item.headers, cache_item.body
            yield headers
            yield body
            raise StopIteration
        else:
            response = self.requestMainServer(path)
            if not response is None:
                headers = self._filterHeaders(response.getheaders())
                self.cacheTable.setHeaders(path, headers)
                yield headers
                len = response.readinto(buffer)
                while len:
                    self.cacheTable.appendBody(path, buffer[0 : len])
                    yield buffer[0 : len]
                    len = response.readinto(buffer)
                raise StopIteration
            else:
                yield headers
                yield body
```

注意，可以通过是否抛出异常StopIteration来区别是否传输成功。

sendHeaders：

```
def sendHeaders(self, headers):
        ''' Send HTTP headers to client'''
        # TODO: implement the logic of sending headers
        for header_item in headers:
            self.send_header(header_item[0], header_item[1])
        self.end_headers()
```

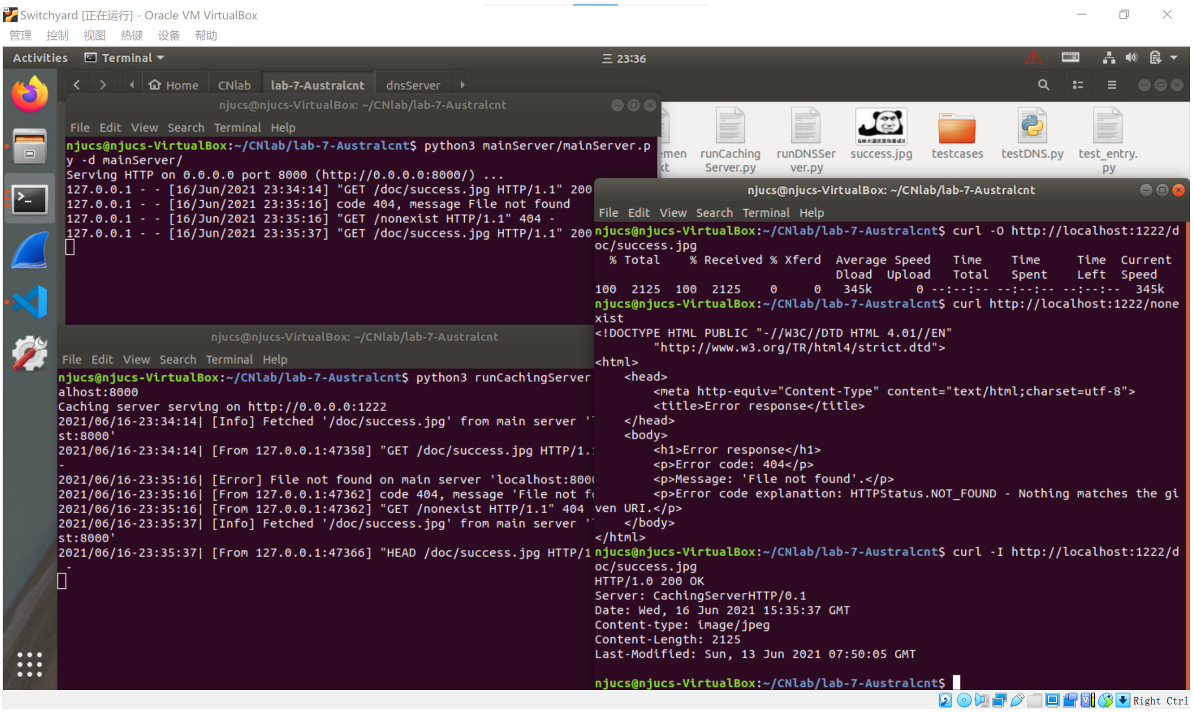
注意，在最后必须加上end_headers。

do_GET：

```
def do_GET(self):
        ''' Logic when receive a HTTP GET.
        Notice that the URL is automatically parsed and the path is stored in
        self.path.
        '''

        # TODO: implement the logic to response a GET.
        # Remember to leverage the methods in CachingServer.

        item = self.server.touchItem(self.path)
        self.headers = next(item)
        if self.headers is None:
            self.send_error(HTTPStatus.NOT_FOUND, "'File not found'")
        else:
            self.send_response(HTTPStatus.OK)
            self.sendHeaders(self.headers)
            '''
            self.body = next(item)
            while self.body:
                self.sendBody(self.body)
                self.body = next(item)
            '''
            while True:
                try:
                    self.body = next(item)
                    self.sendBody(self.body)
                except StopIteration:
                    break
```

可以通过next来访问iterator，当接收到抛出的异常即结束。

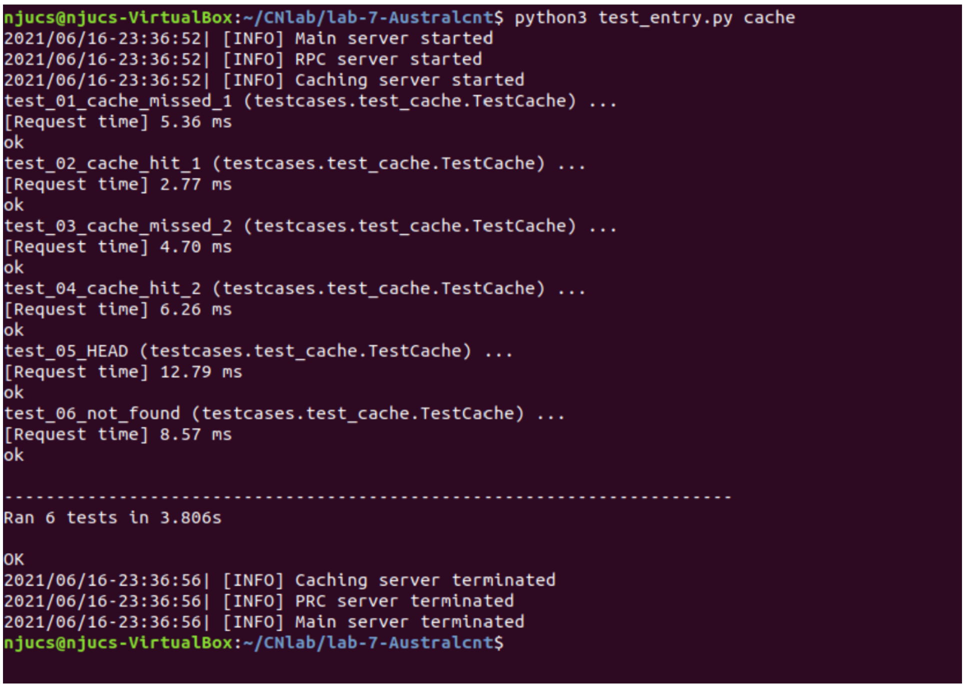
3、测试结果


Switchyard [正在运行] - Oracle VM VirtualBox
管理 控制 视图 热键 设备 帮助
Activities
Terminal
23:36
Home
CNlab
lab-7-Australcnt
dnsServer
runCaching Server.py
runDNSSer ver.py
success.jpg
testcases
testDNS.py
test_entry. py
njucs@njucs-VirtualBox: ~/CNlab/lab-7-Australcnt
File Edit View Search Terminal Help
Right Ctrl

```
njucs@njucs-VirtualBox:~/CNlab/lab-7-Australcnt$ python3 mainServer/mainServer.p
y -d mainServer/
Serving HTTP on 0.0.0.0 port 8000 (http://0.0.0.0:8000/) ...
127.0.0.1 - - [16/Jun/2021 23:34:14] "GET /doc/success.jpg HTTP/1.1" 200
127.0.0.1 - - [16/Jun/2021 23:35:16] code 404, message File not found
127.0.0.1 - - [16/Jun/2021 23:35:16] "GET /nonexist HTTP/1.1" 404 -
127.0.0.1 - - [16/Jun/2021 23:35:37] "GET /doc/success.jpg HTTP/1.1" 200
```

```
njucs@njucs-VirtualBox:~/CNlab/lab-7-Australcnt$ python3 runCachingServer
alhost:8000
Caching server serving on http://0.0.0.0:1222
2021/06/16-23:34:14| [Info] Fetched '/doc/success.jpg' from main server '
st:8000'
2021/06/16-23:34:14| [From 127.0.0.1:47358] "GET /doc/success.jpg HTTP/1.
-
2021/06/16-23:35:16| [Error] File not found on main server 'localhost:800
2021/06/16-23:35:16| [From 127.0.0.1:47362] code 404, message 'File not f
2021/06/16-23:35:16| [From 127.0.0.1:47362] "GET /nonexist HTTP/1.1" 404
2021/06/16-23:35:37| [Info] Fetched '/doc/success.jpg' from main server '
st:8000'
2021/06/16-23:35:37| [From 127.0.0.1:47366] "HEAD /doc/success.jpg HTTP/1
 -
```

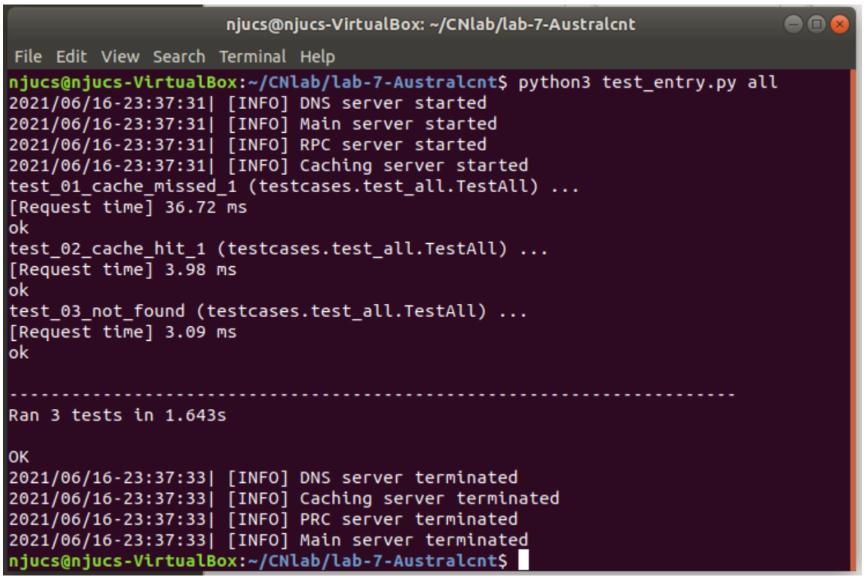
```
njucs@njucs-VirtualBox:~/CNlab/lab-7-Australcnt$ curl -O http://localhost:1222/d
oc/success.jpg
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
100  2125  100  2125    0     0   345k      0 --:--:-- --:--:-- --:--:--  345k
njucs@njucs-VirtualBox:~/CNlab/lab-7-Australcnt$ curl http://localhost:1222/none
xist
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN"
        "http://www.w3.org/TR/html4/strict.dtd">
<html>
    <head>
        <meta http-equiv="Content-Type" content="text/html;charset=utf-8">
        <title>Error response</title>
    </head>
    <body>
        <h1>Error response</h1>
        <p>Error code: 404</p>
        <p>Message: 'File not found'.</p>
        <p>Error code explanation: HTTPStatus.NOT_FOUND - Nothing matches the gi
ven URI.</p>
    </body>
</html>
njucs@njucs-VirtualBox:~/CNlab/lab-7-Australcnt$ curl -I http://localhost:1222/d
oc/success.jpg
HTTP/1.0 200 OK
Server: CachingServerHTTP/0.1
Date: Wed, 16 Jun 2021 15:35:37 GMT
Content-type: image/jpeg
Content-Length: 2125
Last-Modified: Sun, 13 Jun 2021 07:50:05 GMT

njucs@njucs-VirtualBox:~/CNlab/lab-7-Australcnt$
```



```
njucs@njucs-VirtualBox:~/CNlab/lab-7-Australcnt$ python3 test_entry.py cache
2021/06/16-23:36:52| [INFO] Main server started
2021/06/16-23:36:52| [INFO] RPC server started
2021/06/16-23:36:52| [INFO] Caching server started
test_01_cache_missed_1 (testcases.test_cache.TestCache) ...
[Request time] 5.36 ms
ok
test_02_cache_hit_1 (testcases.test_cache.TestCache) ...
[Request time] 2.77 ms
ok
test_03_cache_missed_2 (testcases.test_cache.TestCache) ...
[Request time] 4.70 ms
ok
test_04_cache_hit_2 (testcases.test_cache.TestCache) ...
[Request time] 6.26 ms
ok
test_05_HEAD (testcases.test_cache.TestCache) ...
[Request time] 12.79 ms
ok
test_06_not_found (testcases.test_cache.TestCache) ...
[Request time] 8.57 ms
ok

----------------------------------------------------------------------
Ran 6 tests in 3.806s

OK
2021/06/16-23:36:56| [INFO] Caching server terminated
2021/06/16-23:36:56| [INFO] PRC server terminated
2021/06/16-23:36:56| [INFO] Main server terminated
njucs@njucs-VirtualBox:~/CNlab/lab-7-Australcnt$
```

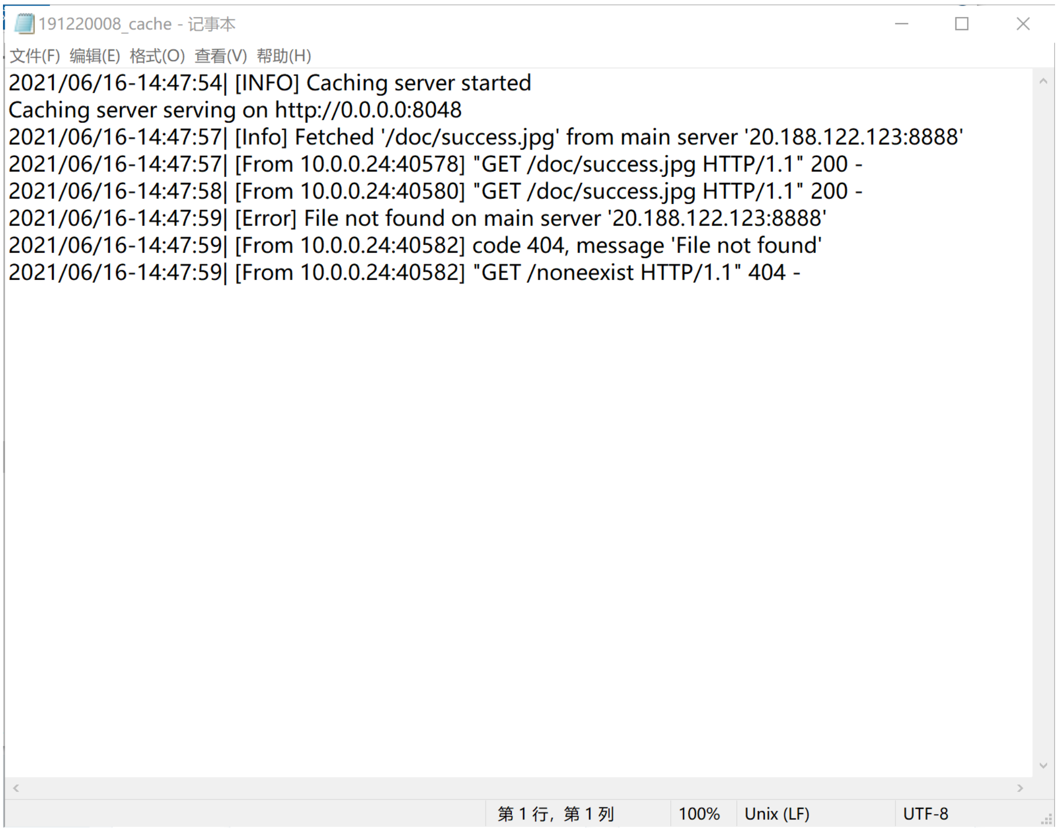
Task 4: Deployment

1、测试结果

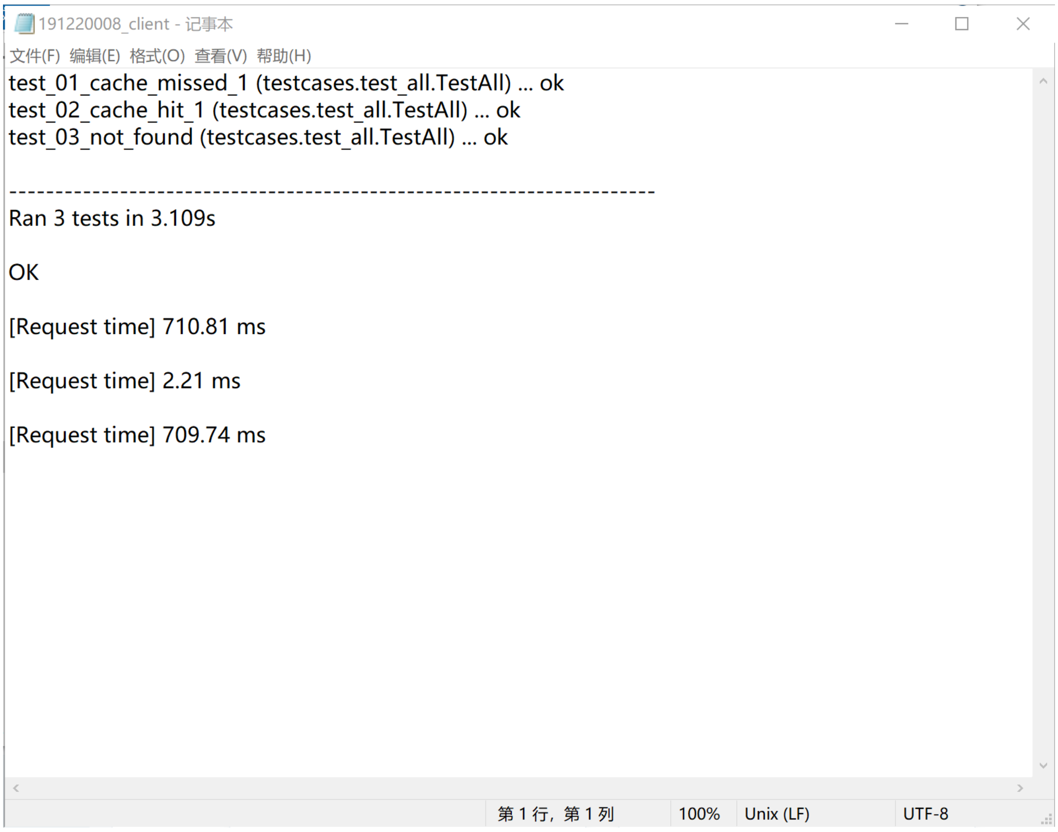


njucs@njucs-VirtualBox: ~/CNlab/lab-7-Australcnt
File Edit View Search Terminal Help

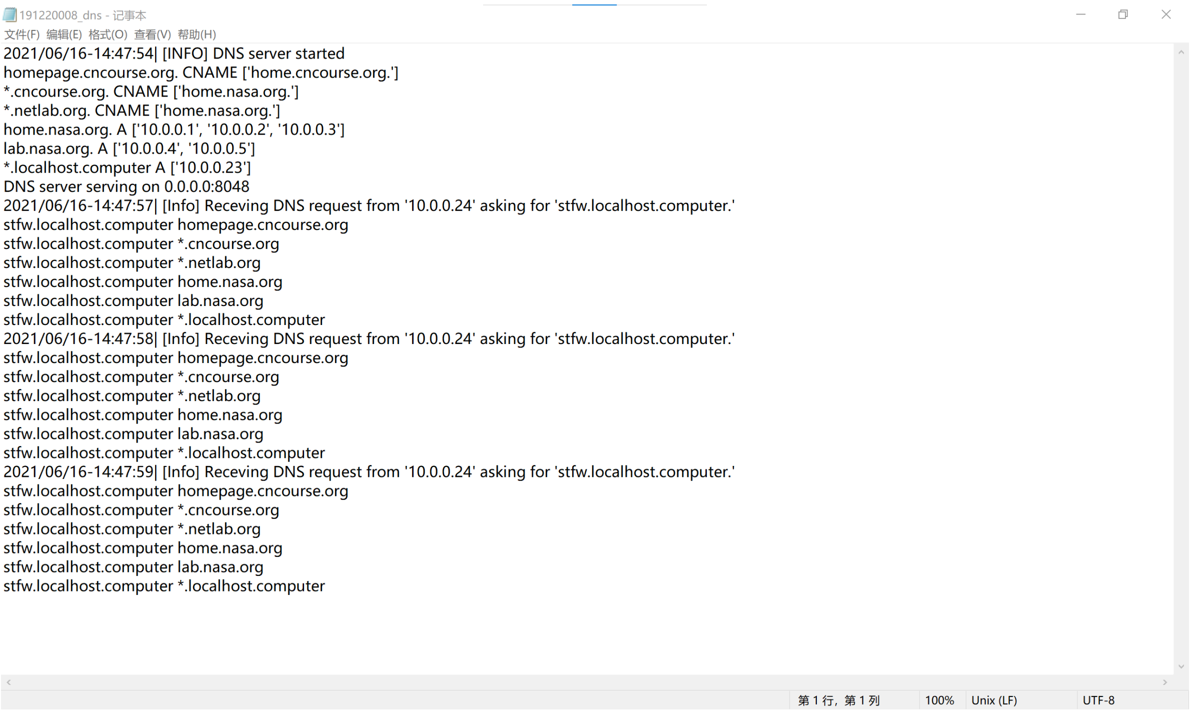
```
njucs@njucs-VirtualBox:~/CNlab/lab-7-Australcnt$ python3 test_entry.py all
2021/06/16-23:37:31| [INFO] DNS server started
2021/06/16-23:37:31| [INFO] Main server started
2021/06/16-23:37:31| [INFO] RPC server started
2021/06/16-23:37:31| [INFO] Caching server started
test_01_cache_missed_1 (testcases.test_all.TestAll) ...
[Request time] 36.72 ms
ok
test_02_cache_hit_1 (testcases.test_all.TestAll) ...
[Request time] 3.98 ms
ok
test_03_not_found (testcases.test_all.TestAll) ...
[Request time] 3.09 ms
ok

----------------------------------------------------------------------
Ran 3 tests in 1.643s

OK
2021/06/16-23:37:33| [INFO] DNS server terminated
2021/06/16-23:37:33| [INFO] Caching server terminated
2021/06/16-23:37:33| [INFO] PRC server terminated
2021/06/16-23:37:33| [INFO] Main server terminated
njucs@njucs-VirtualBox:~/CNlab/lab-7-Australcnt$
```



```
2021/06/16-14:47:54| [INFO] Caching server started
Caching server serving on http://0.0.0.0:8048
2021/06/16-14:47:57| [Info] Fetched '/doc/success.jpg' from main server '20.188.122.123:8888'
2021/06/16-14:47:57| [From 10.0.0.24:40578] "GET /doc/success.jpg HTTP/1.1" 200 -
2021/06/16-14:47:58| [From 10.0.0.24:40580] "GET /doc/success.jpg HTTP/1.1" 200 -
2021/06/16-14:47:59| [Error] File not found on main server '20.188.122.123:8888'
2021/06/16-14:47:59| [From 10.0.0.24:40582] code 404, message 'File not found'
2021/06/16-14:47:59| [From 10.0.0.24:40582] "GET /noneexist HTTP/1.1" 404 -
```


191220008_client - 记事本
文件(F) 编辑(E) 格式(O) 查看(V) 帮助(H)
第 1 行，第 1 列
100%
Unix (LF)
UTF-8

```
test_01_cache_missed_1 (testcases.test_all.TestAll) ... ok
test_02_cache_hit_1 (testcases.test_all.TestAll) ... ok
test_03_not_found (testcases.test_all.TestAll) ... ok

----------------------------------------------------------------------
Ran 3 tests in 3.109s

OK

[Request time] 710.81 ms

[Request time] 2.21 ms

[Request time] 709.74 ms
```



```
2021/06/16-14:47:54| [INFO] DNS server started
homepage.cncourse.org. CNAME ['home.cncourse.org.']
*.cncourse.org. CNAME ['home.nasa.org.']
*.netlab.org. CNAME ['home.nasa.org.']
home.nasa.org. A ['10.0.0.1', '10.0.0.2', '10.0.0.3']
lab.nasa.org. A ['10.0.0.4', '10.0.0.5']
*.localhost.computer A ['10.0.0.23']
DNS server serving on 0.0.0.0:8048
2021/06/16-14:47:57| [Info] Receving DNS request from '10.0.0.24' asking for 'stfw.localhost.computer.'
stfw.localhost.computer homepage.cncourse.org
stfw.localhost.computer *.cncourse.org
stfw.localhost.computer *.netlab.org
stfw.localhost.computer home.nasa.org
stfw.localhost.computer lab.nasa.org
stfw.localhost.computer *.localhost.computer
2021/06/16-14:47:58| [Info] Receving DNS request from '10.0.0.24' asking for 'stfw.localhost.computer.'
stfw.localhost.computer homepage.cncourse.org
stfw.localhost.computer *.cncourse.org
stfw.localhost.computer *.netlab.org
stfw.localhost.computer home.nasa.org
stfw.localhost.computer lab.nasa.org
stfw.localhost.computer *.localhost.computer
2021/06/16-14:47:59| [Info] Receving DNS request from '10.0.0.24' asking for 'stfw.localhost.computer.'
stfw.localhost.computer homepage.cncourse.org
stfw.localhost.computer *.cncourse.org
stfw.localhost.computer *.netlab.org
stfw.localhost.computer home.nasa.org
stfw.localhost.computer lab.nasa.org
stfw.localhost.computer *.localhost.computer
```

五、总结与感想

本次实验和以往的实验不太一样，比以往更加有趣。在实验过程中，查找了很多API来辅助编程，脱离了以往实验聚焦于细节处理的烦恼，但也产生了许多意想不到的困难。并且，在和其他同学的交流中，也发现了别人不同的实现思维，也学习到别人查找API的方法，学到许多。网络实验以此结束，很开心。