Aditya Chunduru
00335780
Mini project 3: Log Analysis
CIS 153 L8
Program description: This program takes apache server log file as input and provides analytics on resources and requesters.

How did you identify the key parts of the file?

I identified the key parts of the file by using a regular expression pattern according to the file format of apache access log file. I examined the sample apache log file provided and referred to the apache server documentation on access logs and formats. Since log file formats can be customized, I went with file format of the sample server access log to come up with the regex pattern as shown below

```
pattern_of_log = re.compile(r'(.+\..+\..+) - - \[.*?\] "(.*?) (.*?) HTTP/\d\.\d"')
```

These are typical/sample access log entries:

```
  #64.242.88.10 - - [07/Mar/2004:16:06:51 -0800] "GET /twiki/bin/rdiff/TWiki/NewUserTemplate?rev1=1.3&rev2=1.2 HTTP/1.1" 200 4523
   #cr020r01-3.sac.overture.com - - [11/Mar/2004:13:06:17 -0800] "GET /twiki/bin/view/Know/WebNotify HTTP/1.0" 200 4472
   #64-249-27-114.client.dsl.net - - [11/Mar/2004:14:53:12 -0800] "GET /SpamAssassin.html HTTP/1.1" 200 7368
```

The apache server documentation on access log file format is here:
https://httpd.apache.org/docs/2.4/mod/mod_log_config.html#logformat

The pattern identified is:

1. Ip address/remote host name
2. - - : unsure as almost all access log entries are blank in the file providedp; not critical for log analysis.
3. Date & time
4. URI
5. HTTP version and status code(s).

Here is the regex to match the access log entries:

```
pattern_of_log = re.compile(r'(.+\..+\..+) - - \[.*?\] "(.*?) (.*?)
HTTP/\d\.\d"')
```

How did you figure out who requested the most?

Overall, log analysis program is organized in two functions. Read function parses the log access file and builds dictionaries for tracking resources, requesters and counts around resources and requesters.

For each log access line, using regex identify resource being requested, requestor determined by ip address/remote host name and keep count of resource requests; all done through four dictionaries:

- requesters
- resources
- Requester_to_resources
- Resource_to_requesters

Once the parsing is done in read function(read_log_file), here is the code that identifies who requested the most as shown below:

```
    #Get most common requester among list of requesters
    most_common_requester = max(requesters, key=requesters.get)
```

How did you find out what resources were most requested?

Just like above, after the read function is executed, the below code identifies what resource was most requested:

```
#Most_common_resource searches for the key with the highest value in the resources dictionary.
    most_common_resource = max(resources, key=resources.get)
```

What is the most commonly accessed resource?

The most commonly accessed resource is “/twiki/pub/TWiki/TWikiLogos/twikiRobot46x50.gif”

How many times was it accessed?

It was accessed 64 times.

Who (what IP or domain) is requesting this the most?

The IP address that’s requesting this the most is 10.0.0.153

How many requests were made by that top requester?

The amount of requests that were made by that top requester is 4.

Who (what IP or domain) made the most requests?

The IP address that made the most requests was also 64.242.88.10

How many requests were made?

452 requests were made

.

What was their most requested resource?

Their most requested resource was “/twiki/bin/view/Main/WebHome”

how many times was this resource requested

This resource was requested 3 times.