CS 519 - Homework 8 Paul ReFalo 11/17/17

For my project I choose the XKCD download comics project at the end of chapter 15 (https://automatetheboringstuff.com/chapter15/). This is a project about multithreading so I first got the project working, then I devised a way to time the downloading of 48 comics from the xkcd.com website with 2, 4, 6, 8, 10, and 12 threads. I choose 48 comics because that was large enough to show some effect on the use of multi-threading and choose threads by 2 for arithmetic convenience. The data is plotted below. Since adding a thread means there are that many times the amount of processing being done, the rate change should be multiplicative and not additive.

I believe this will make the time complexity of this process will be O(n log(n)). Here's a typical output of the script (python script attached at the bottom):

Example output of multidownloadXkcd.py script timed with timit module

Done downloading 48 XKCD comics with 2 threads. Runtime is 14.68 seconds.

Done downloading 48 XKCD comics with 4 threads. Runtime is 5.27 seconds.

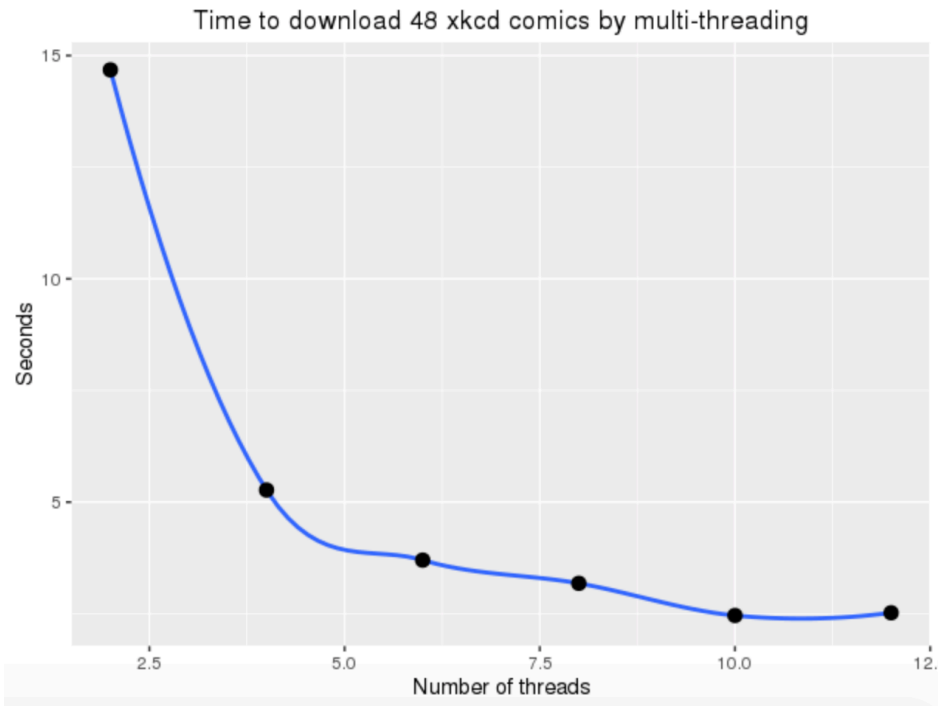
Done downloading 48 XKCD comics with 6 threads. Runtime is 3.7 seconds.

Done downloading 48 XKCD comics with 8 threads. Runtime is 3.18 seconds.

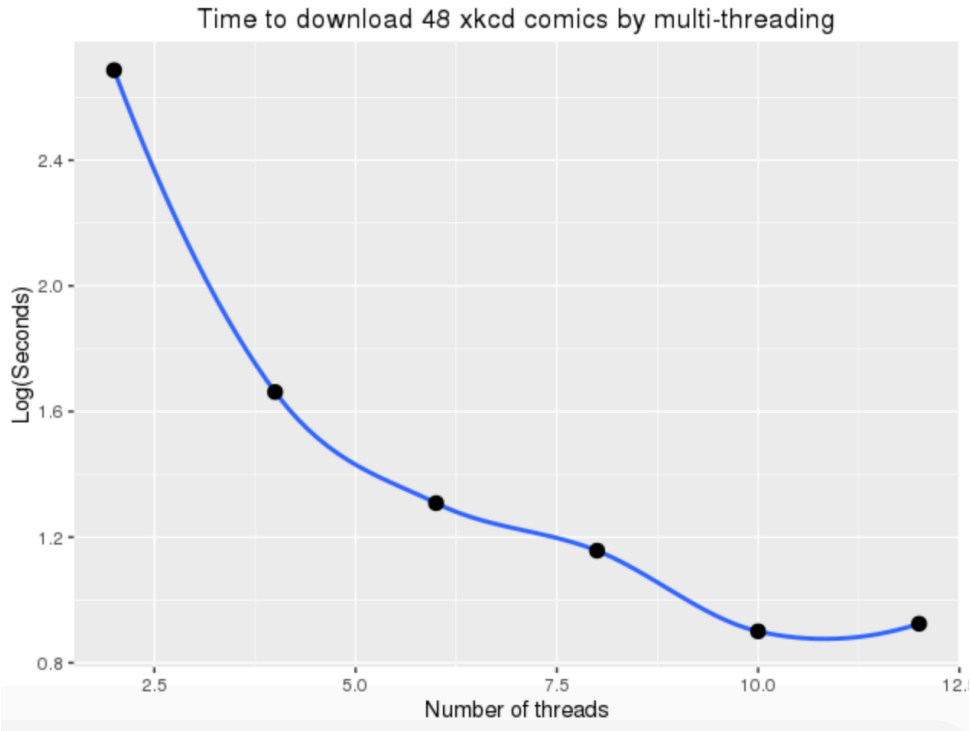
Done downloading 48 XKCD comics with 10 threads. Runtime is 2.46 seconds.

Done downloading 48 XKCD comics with 12 threads. Runtime is 2.52 seconds.

Here is a plot of the data using R.

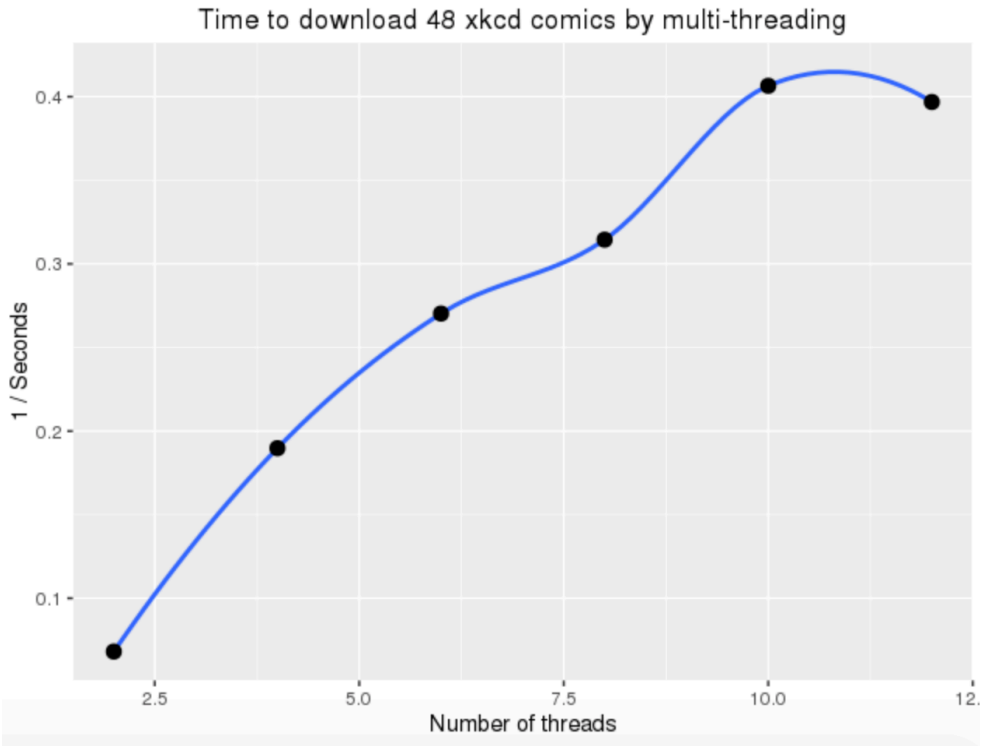


Here is the same data with a Log transformation of the y-axis using R.



The result of the log plot is not exactly in a one-to-one relationship with the number of threads working to download the images but it is closer to a linear relationship that the un-transformed plot of the same data.

Here's a plot of 1 / seconds on the y-axis. The relationship is much more linear than the log plots with the notable exception of the last data point (threads = 12). This may indicate the limitations of the laptop I am using for analysis.



multidownloadXkcd.py script

```
#OSU CS 519

#! python3
# multidownloadXkcd.py - Downloads XKCD comics using multiple threads.
# by Paul ReFalo 17 Nov 2017

import requests, os, bs4, threading, sys, timeit
from pprint import pprint

os.makedirs('xkcd', exist_ok=True) # store comics in ./xkcd

def downloadXkcd(startComic, endComic):
    for urlNumber in range(startComic, endComic):
        # Download the page.
        #print('Downloading page http://xkcd.com/%s...' % (urlNumber))
        res = requests.get('http://xkcd.com/%s' % (urlNumber))
        res.raise_for_status()

        soup = bs4.BeautifulSoup(res.text, "html.parser")

        # Find the URL of the comic image.
        comicElem = soup.select('#comic img')
        if comicElem == []:
            print('Could not find comic image.')
        else:
            # comicUrl = comicElem[0].get('src')
            comicUrl = 'https:' + comicElem[0].get('src')

            # Download the image.
            #print('Downloading image %s...' % (comicUrl))
            res = requests.get(comicUrl)
            res.raise_for_status()

            # Save the image to ./xkcd
            imageFile = open(os.path.join('xkcd', os.path.basename(comicUrl)), 'wb')
            for chunk in res.iter_content(100000):
                imageFile.write(chunk)
            imageFile.close()
```

```
def manageThreads(comics, threads):
    segments = int(comics / threads)
    # Create and start the Thread objects.
    start = timeit.default_timer()

    downloadThreads = [] # a list of all the Thread objects
    # was range(0, 1400, 100)
    for i in range(101, 101 + comics, segments): # loops comics/threads times, creates
the number of threads
        downloadThread = threading.Thread(target=downloadXkcd, args=(i, i + segments))
        downloadThreads.append(downloadThread)
        downloadThread.start()
        #print("========= " + str(i))

    # Wait for all threads to end.
    for downloadThread in downloadThreads:
        downloadThread.join()

    stop = timeit.default_timer()
    runTime = stop - start
    runTime = str(round(runTime, 2))

    print('Done downloading ' + str(comics) + ' XKCD comics with ' + str(threads) + '
threads.  Runtime is ' + runTime + ' seconds.')

for t in range(2, 13, 2):   # loop from 2 to 12 by 2's to get data on these number of
threads
    manageThreads(comics=48, threads=t) # comics and threads
```