

## Timed Lab 3 – Grade Cutoffs

This is a Timed Lab; this Timed Lab is worth 25 **Exam & Timed Lab** points.

For this Timed Lab, you <i>may</i> use	However, you <i>may not</i>
<ul style="list-style-type: none"> <li>• Course notes , Homeworks and Recitation Assignments.</li> <li>• Other course material</li> <li>• Any material you may find on the Internet that don't involve communicating "live" with other people.</li> </ul>	<ul style="list-style-type: none"> <li>• Communicate with other people/students in real-time via any means. This means no Facebook, email, Piazza, IM, IRC, cell phones, Google Talk, smoke signals, etc.</li> <li>• Share code with other students.</li> <li>• Look at other students work.</li> </ul>

The TAs will be available to answer clarifying questions about the problem, but they are not permitted to give assistance with debugging code, program logic, etc. You will have an entire recitation period to work on this assignment; this time begins *exactly* when your recitation begins and ends *exactly* when your recitation ends: **No extra time will be given if you arrive late**, except in highly extenuating circumstances that must be approved by Dr. Summet or the Head TA.

T-Square will not permit any late submissions; ensure that you submit your code to T-Square several times to prevent earning a zero due to you being unable to submit. Your TAs will give a verbal warning 10 and 5 minutes before the end of the recitation period; you should submit at these times.

In your collaboration statement, if you use code from somewhere that is *not* a class resource (i.e. not listed on the course calendar), please list where this code came from. Ensure that you fill out the header at the top of the file.

Note that you must check out with your TA before you leave the recitation room. If you do not check out with your TA or you modify your submission after you leave the recitation room, **you will receive a grade of zero on the timed lab**. No submissions will be accepted after T-Square closes the assignment (i.e. it will not let you submit).

### Problem Description:

You work for the Dean of the College of Engineering who has discovered that the grade cutoffs between classes in the College vary too wildly and need to be standardized. The College of Engineering has determined that the following are acceptable cutoffs for each letter grade. Note that for each range, both numbers are inclusive.

- A:** 84 – 100
- B:** 75 – 89
- C:** 65 – 79
- D:** 55 – 69
- F:** 0 – 59

Your job is to scrape the websites of various professors in the College of Engineering and ensure that their letter grade cutoffs are within the acceptable limits described above. To be within the limit for the letter grade, the lower bound and upper bound for the professor's grade must be within or equal to the bounds given above. For example, if the professor's A range was 90-100, this is acceptable as both the lower bound (90) and upper bound (100) are both within the given range for an A (84-100). However, if the professor's range for a C was 60-75, this would not pass because the lower bound for a C (60) does not fall within the bounds for a C given above (65-79).

On each webpage, you need to retrieve two main pieces of data: the professor's name and the grading scale. Professor's names will always be in the format <TITLE> <NAME> where <TITLE> is either Dr. or Prof., <NAME> is two words, separated by a space, and letter grade cutoffs are given in the format <LETTER>: <LOWER\_BOUND> - <UPPER\_BOUND>. <LETTER> is A,B,C,D, or F, <LOWER\_BOUND> is an integer,

and `<UPPER_BOUND>` is an integer. If you're unsure of the format, check out the three sample HTML files you should use for testing, found at the following URLs:

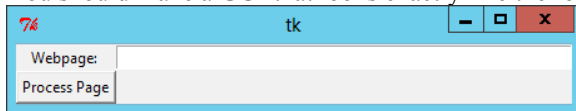
[http://www.cc.gatech.edu/classes/AY2013/cs2316\\_spring/codesamples/spring2013-tl3-syllabus\\_1.html](http://www.cc.gatech.edu/classes/AY2013/cs2316_spring/codesamples/spring2013-tl3-syllabus_1.html)

[http://www.cc.gatech.edu/classes/AY2013/cs2316\\_spring/codesamples/spring2013-tl3-syllabus\\_2.html](http://www.cc.gatech.edu/classes/AY2013/cs2316_spring/codesamples/spring2013-tl3-syllabus_2.html)

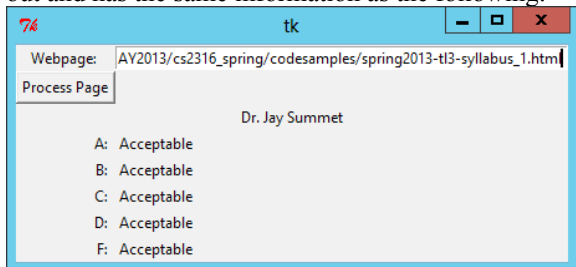
[http://www.cc.gatech.edu/classes/AY2013/cs2316\\_spring/codesamples/spring2013-tl3-syllabus\\_3.html](http://www.cc.gatech.edu/classes/AY2013/cs2316_spring/codesamples/spring2013-tl3-syllabus_3.html)

Note that your solution should validly process the above webpages **AS WELL AS ANY OTHER WEBPAGE THAT HAS THE REQUIRED INFORMATION**. Note that each page will only have one valid professor name and will have all contain values for all letter grades in the format specified above.

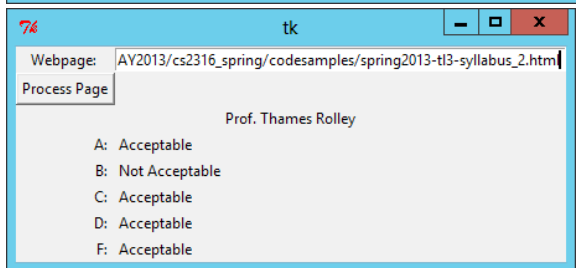
You should make a GUI that looks exactly like the following when it is first run (the entry is width 60):



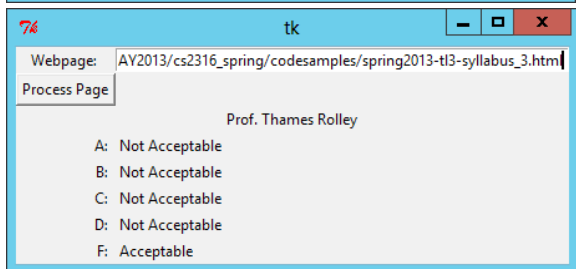
Upon clicking Process Page, the webpage should be scraped as described above, and you should determine if the letter grades are either acceptable (in the range) or not acceptable (not in the range). Then, display a GUI that is laid out and has the same information as the following:



(syllabus\_1.html)



(syllabus\_2.html)



(syllabus\_3.html)

## Grading

### GUI (10 points):

- +3: Initial GUI has label, entrybox of width 60, and process page button (in correct locations)
- +2: After button click, Professor name centered on line below button
- +2: Labels for all valid letter grades added after button click (in correct locations)
- +3: Labels for all acceptability of letter grades added after button click (in correct locations)

### Data Scraping/Processing (15 points):

- +4: Correctly downloads data from given webpage
- +3: Correctly retrieves professor names in the given format (and only from this format)
- +3: Correctly retrieves letter grades in the given format (and only from this format)
- +4: Correctly determines whether a given letter grade falls within the acceptable range above
- +1: Correctly updates GUI with the information after scraping