Regular Expressions Project

Eckel, TJHSST AI1, Fall 2019

# Background & Explanation

Regular expressions are a powerful tool for accomplishing a variety of unexpected tasks going far beyond a simple text search. We’ll become masters of the tool in this assignment, then later discuss implementing it in Python code. For now, you’re just writing the expressions and learning everything they can do.

# Assignment & Two Outstanding Work Specifications

If you go to the website ai.sites.tjhsst.edu you’ll notice there are several assignments labelled RE in the drop down menu. Each one requires you to type or submit a file containing several regular expressions. Your code will not run them, your code will simply present them as strings. The grader will run your regular expressions and either give you failures or, if all expressions are correct, a character count.

This assignment makes up four lab grades, with opportunities for outstanding work. The grades are based on the total character count for each problem set, as follows. For each target, you want at most the number of characters given, precisely hitting the target or getting a *smaller* character count.

|  |  |  |  |
| --- | --- | --- | --- |
| Assignment | Target for credit | Target for 1 outstanding work credit | Target for 2 outstanding work credits |
| RE1 | 170 | 150 | 144 (not the minimum; 143 is possible) |
| RE2 | 236 | 204 | 184 (not the minimum; 181 is possible) |
| RE3 | 226 | 210 | 190 (maybe the minimum?) |
| RE5 (note: not RE4!) | 247 | 231 | 211 (not the minimum; see what you can find…) |

To be totally clear, writing RegEx expressions within a certain character count is not, in and of itself, a necessary skill. The character counts above, though, will force you to use many different capabilities of the RegEx engine, ensuring that you’ve explored all of its aspects thoroughly.

To get credit on each assignment, you must get the target in the first column. These are four *separate* 10 point grades.

For one outstanding work credit, you must get *every value* in the second column. This is *one* outstanding work credit for all four assignments *together*.

For two outstanding work credits, you must get *every value* in the third column. This is an *additional* outstanding work credit for all four assignments *together*.

# Another Outstanding Work Option: RegEx Assignment #4

A separate option for outstanding work on this project is to complete RE4. RE4 is a particularly advanced set of problems, and each problem includes its own character target. To get outstanding work this way, you must complete every RE4 question correctly. You must also hit the character target on at least 6 out of the 10 problems.

**Note: you may receive AT MOST TWO outstanding work credits on this project. You can choose between optimizing the required assignments or stretching to the advanced assignment, but you can’t receive three outstanding work credits for doing all of the options.**

# Plagiarism & Collaboration

Since this is not a coding task, the rules for collaboration are different. Please read carefully! These rules apply to students of mine and students of any other teacher also doing Regular Expressions.

On this assignment:

* You **MAY** explain how a RegEx feature works *in general* to another student in your own words. You **MAY NOT** explain how your RegEx solution to a problem works.
* If you have already solved a RegEx problem, you **MAY** explain why another student’s RegEx fails on a certain counter-example (ie, when the grader says a RegEx fails, you can explain why that test case is matching incorrectly). You **MAY NOT** directly give another student advice for how to fix a RegEx that isn’t working.
* You **MAY** tell another student your character count **for an entire problem set**. You **MAY NOT** tell another student your character count **for a particular problem**. \*THIS IS EXTREMELY IMPORTANT!\*

So, this is ok: “Let me explain to you how backreferences work.” …but this is not: “I used backreferences for number 22!”

And this is ok: “Your RegEx will match here, where you don’t want it to, because your \* allows it to skip this part of the word” …but this is not: “You need to replace this \* with a +”.

And this is ok: “I’m still at 243 characters total on RE5” …but this is **very much not ok**: “Number 23 is 17 characters, number 24 is 19 characters, and number 25 is 23 characters.” **NOT OK.**

# Topics Covered on Each Assignment

Assignment #1 does **not** allow parentheses. Use everything you learned in taking notes in class.

Assignment #2 **does** allow parentheses. **Not** every question needs them. You may **not** use lookarounds or back references. Parentheses may be nested.

The **first three questions** of Assignment #3 do **not** allow lookarounds, but they do allow back references:

* Back references are referenced by \1, \2, etc, and remember that (?:…) doesn’t capture.

After that, for the rest of Assignment #3 and all the remaining assignments, all parenthetical options are available:

* Look aheads – positive is (?=…) and negative is (?!…). A look ahead is an assertion and does not munch; it asserts that there is a match to the expression inside the look ahead at this location. Parentheses inside the look ahead may capture and remember, but the look ahead parentheses themselves do not. Parentheses inside a negative look ahead can be back referenced within the negative look ahead, but make no sense outside of that context.
* Look behinds – positive is (?<=…) and negative is (?<!…). Remember that all look behinds must be fixed width; no variable quantifiers.

If you’re reading this and this doesn’t make sense, don’t worry; I’ll show a few examples in class of how to use these advanced features. You may wish to use the blank back of this assignment sheet for notes so they’re all in the same place.

Alternately, it’s totally ok with me for you to google these things, find online tutorials, and work ahead.

# Submission Instructions

First, you must submit your expressions to the grader.

Since regular expressions contain a lot of characters that Python treats as special characters, you’ll want to add an “r” in front of each string to make it a raw text string where Python doesn’t interpret any of those special characters.

You can either type a file on your computer and upload it or write it on the screen. Start with this basic code:

**import** sys  
idx = int(sys.argv[1])-30  
myRegexList = [  
 **r""**, *# Question 30* **r""**,   
 **r""**,  
 **r""**,  
 **r""**,  
 **r""**,  
 **r""**,  
 **r""**,  
 **r""**,**r""**] *# Question 39*print(myRegexList[idx])

Note that this takes a command line argument (an integer specifying the question number) and then returns the string you’ve written to correspond to that question. The strings correspond to questions 30 to 39 in that order, and each should contain a regular expression as shown in class – delimited by slash marks, with any modifiers on the right.

Once you receive full credit from the grader, you must screenshot and **submit your work to me as well**.

Take a screenshot of the grader’s output showing your correct responses and character count as well as your name. Submit the screenshot as a .jpg file. On Windows, you can accomplish this by hitting the “print screen” key to take a screenshot, then pasting into Paint, cropping the part of the screen you want to see, and saving as a jpg. I’m sure on a Mac it’s even easier. Submit your screenshots to the links below. (Be aware: these *will* be verified against the grader’s records, and the usual plagiarism penalties will apply if any discrepancies are found.)

# Required Submission Links

Submit a single screenshot to each link below; you do not need to submit your scripts.

RE1: <https://tinyurl.com/F19EckelRE1> RE2: <https://tinyurl.com/F19EckelRE2>

RE3: <https://tinyurl.com/F19EckelRE3> RE5: <https://tinyurl.com/F19EckelRE5>

# Outstanding Work Submission Links

For one outstanding work credit, submit **four screenshots** to <https://tinyurl.com/F19EckelREOW1>

For two outstanding work credits, submit **four screenshots** to <https://tinyurl.com/F19EckelREOW2>

For outstanding work credit for doing RegEx assignment 4, submit a screenshot **and your Python script itself** to <https://tinyurl.com/F19EckelRE4OW> (this is the only one that requires the Python file as well as screenshots)