

Python RegEx Worksheet - June 30, 2015
Printed Name of Both Partners

1. Give the nicely clean and compact RegEx to match and return all occurrences of cs2316, CS2316, cs 2316, CS 2316, Cs 2316, Cs2316, cS2316, cS 2316. Capture and return only the course matches (not strings that contain the course as a substring). (Notice this doesn't care about regular word boundaries.)

```
>>> findall("[cC][sS] ?2316", "cs2316 cS 2316 CS2316foobar fourcs 2316three")
['cs2316', 'cS 2316', 'CS2316', 'cs 2316']
```

2. Give the RegEx for any alpha word that contains three or more x's as a substring. Return the entire whitespace delimited word, not just the xxx part. (Definitely word boundary is helpful.)

```
>>> findall("\\b[A-Za-z]*x{3}[A-Za-z]*\\b", "xxxdrop airxx xxwide fooxxxbar 123xxx456")
['xxxdrop', 'fooxxxbar']
```

3. Give the RegEx for any alpha word that contains "gt" as a subsequence. (Subsequence means there could be letters between the g and the t.) (Not exactly caring about word boundaries, but only snagging the letters not other chars.)

```
>>> findall("[A-Za-z]*g[A-Za-z]*t[A-Za-z]*",
            "gt cS 2316 CS2316foobar fourcs 2316gthree fivecs 2316 g tsix gcats")
['gt', 'gthree', 'gcats']
```

4. Give the RegEx to find words that are exclusively g's and t's. (Word boundary is handy.)

```
>>> findall("\\b[gt]+\\b", "gt ggt gtgt catgt")
['gt', 'ggt', 'gtgt']
```

```
>>> findall(r"\b[gt]+\b", "gt ggt gtgt catgt")
['gt', 'ggt', 'gtgt']
```

5. Give the RegEx to find words that contain an odd number of t's. (They can contain other alpha characters too.) Capture and return the whole word. (Using word boundary.)

```
>>> findall("\\b[a-su-zA-SU-Z]*t[a-su-zA-SU-Z]*(?:[a-su-zA-SU-Z]*t[a-su-zA-SU-Z]*t[a-su-
zA-SU-Z]*)*\\b", "atbtctd etftgth ittttj xttt ytttt")
['atbtctd', 'etftgth', 'ittttj']
```

6. Give the RegEx for words that contain "cat" as a substring. It should allow "cat", "Cat", "CaT", etc. (Any mix of case is allowed.) (Word boundary and being sure to ignore words with non-letters in them.)

```
>>> findall("\\b[a-zA-Z]*[cC][aA][tT][a-zA-Z]*\\b", "Cat cat CAT cats catfood scats ccaatt
cats2")
['Cat', 'cat', 'CAT', 'cats', 'catfood', 'scats']
```