

Main Questions

1. What is challenging about deciding whether to write a regex vs. reusing a regex vs. writing your own non-regex code for a string matching problem?
 - a. Other people have mentioned their regex problem being simple as both a reason to write their own regex and as a reason to reuse it instead. Would you attribute this finding to the decision being challenging or to instead different people's preferences?
 - b. How do you address/solve these challenges?
2. What is challenging about composing regexes? How do you address/solve these challenges?
 - a. Match too much vs. match too little? (added 5/6)
3. What is challenging about reusing regexes?
 - a. Other people have mentioned challenges in all these aspects: (1) describing the problem itself that the regex needs to solve, (2) finding a regex to solve that problem, (3) assessing which of the found regexes is the right one to reuse, and (4) dealing with syntax differences between the reused-regex programming language and the programming language in which it will be used. Could you comment in these particular challenges?
 - b. How do you address/solve these challenges?
4. What is challenging about validating regexes?
 - a. Other people have mentioned challenges in all these aspects: (1) assessing whether enough of the input space has been tested, (2) assessing their runtime/performance. Could you comment in these particular challenges?
 - b. How do you address/solve these challenges?
5. What is challenging about maintaining/modifying existing regexes in code?
 - a. Other people have mentioned challenges in all these aspects: (1) reading regexes, (2) documenting regexes, and (3) code-reviewing regexes. Could you comment in these particular challenges?
 - b. How do you address/solve these challenges?
6. What are your perceptions of regex? added (5/6)

Extra Time Topics/Questions

Background:

- On your survey you indicated you were an expert regarding regular expressions. Can you give me a little bit more background on how you use regex, and how you developed your expertise?

Process:

- What is your typical process when creating a regular expression?
 - Why do you follow this process?
 - Which do you find the most problematic part of this process?
- How is your process different when modifying an existing regex you are not familiar with?
 - What steps do you take to make this process easier or safer?
 - What about the original code helps you in this process? Or make it harder?
- When and how do tools or other resources feature in your process?
 - Why do you use tools in this way/not use tools?
 - Are there any situations where you would be more likely to consider using a tool?
- Could you explain how you usually validate your regular expressions?
 - Do you ever write automated tests for them? Why/why not?
 - How important is it to you to validate regular expressions?
 - Do you ever ask colleagues for input or advice when working with regular expressions?
- How do you obtain sample input for testing regular expressions?
 - Do you preserve any sample input that you use?
 - Have you ever heard of/used input generation tools? (e.g. brics, rex)
- Do you document your regular expressions?
 - Why do you document in this way/not document?
- What is your approach to reviewing code that includes regular expressions?
 - Why do you code review regexes in this way?
 - What are the biggest challenges you face in such a task?
 - What would help you the most? (e.g. documentation, tooling)
- What recommendations would you give an inexperienced developer about regex development?
 - Are there anything's you would advise them against doing?

ReDoS:

- Have you ever had to deal with performance issue in a regular expression?
 - How did/would you go about resolving it?
- Are you familiar with the concept of Regular Expression Denial-of-Service (ReDoS)?
 - If not, give example
- How would you handle a confirmed report of a ReDoS vulnerability in your code?
 - How serious a risk do you consider ReDoS?

- What tools or information would help you handle vulnerabilities more effectively?

Regex Reuse:

- What triggers you to want to reuse a regex?
 - Is this similar to what triggers you to want to reuse other code?
 - Once you have started to write a regex what might make you try to reuse one instead?
 - Once you have started trying to find a regex to reuse what might make you want to write one instead?
- Do you encounter any specific issues when trying to reuse a regex?
 - Do you encounter cross language semantic problems?

General Struggles:

- What do you feel is challenging about regex?
 - Are there problems that you have encountered that you solve with tool specifically?
 - Are there issues you felt as a novice that you feel have gone away with your development of more expertise?
 - Are there challenges you still face with regex that you don't have good solutions for?