COMP 141: Lexics and CFGs

Instructions: In this exercise, we are going to review

1. regular expressions as a mechanism to specify tokens
2. a toy scanner implementation

1 Regex

1. Define the regex for the following description of tokens:
   (a) Any string that starts with character `t`.
   (b) Any string of at least length 3 that starts with `t` and ends with `u`.
   (c) Any string that specifies the range of numbers between 11 and 23.
   (d) Any string that specifies a date in MM:DD:YYYY format.
2. In C, an identifier is defined as a string of characters (both upper-case and lower-case), digits, and underscore "_", starting with either a character or underscore. Define the regex for identifiers in C.
3. Give five strings that conform with the regex: **`[0-9]+((E|e)(\+|\-)?[0-9]+)?`**

2 Toy scanner

1. Write a simple program in C++ that receives **a single token** as input and checks
   - if the received token is a positive integer number
   - if the received token is a punctuation character: `+`, `*`, `(`, `)`.

   You must use **regular expressions** in contrast to the example in lecture, and check if the input matches the token definitions above.

   The output of the program may look like the following:

```
$ ./main
716235
number

$ ./main
(
punc

$ ./main
*
punc
```