Regular expressions unit:

- Regular expressions—principles and Python (**today**)
- Lab: Building a RegEx-based chatbot (Friday)
- The edit distance algorithm [stand alone topic] (next week Monday)

Today:

- Retrospective from lab last time
- Why we care about regular expressions
- Review and practice of regular expressions by definition
- Overview and demo of regular expressions in Python

- An **alphabet** is a set of symbols, Σ.
- A **string** over an alphabet is a sequence of symbols from that alphabet. $\Sigma*$ is the set of all strings over alphabet Σ.
- A **language** over an alphabet is a set of strings, that is, a subset of $\Sigma*$.

- **Regular expressions** constitute a system for specifying languages. (J&M, "a language for specifying text search strings. . . an algebraic notation for characterizing a set of strings.", §2.7, pg 18.).
  An individual regular expression denotes a language, that is, a set of strings.

base cases	$\emptyset$	the empty set of strings
	ε	the set containing the empty string, $\{""\}$
	a	the set containing only the string with only a, for some $a \in \Sigma$, $\{"a"\}$
recursive cases	rs	the set of strings made from concatening strings from r and s, $\{x + y \mid x \in r \wedge y \in s\}$, for some regular expressions r and s
	$r\|s$	the set of strings from r or s, $r \cup s$ for some regular expressions r and s
	$r*$	the set of strings made from concatenating 0 or more strings from r for some regular expression r

Abbreviation	**Meaning**	**Equivalence**
`[abc]`	One occurrence of any of these symbols	`(a|b|c)`
`[a-c]`	One occurrence of any symbol in this range	`(a|b|c)`
$r?$	Optionally an occurrence of a string defined by r	$(r\|\varepsilon)$
$r\{5\}$	5 occurrences of a string defined by r	$rrrrr$
$r\{3,5\}$	Between 3 and 5 occurrences of a string defined by r	$(rrr\|rrrr\|rrrrr)$
$r+$	One or more occurrences of a string defined by r	$rr*$

- *DNA sequences:* `(A|C|G|T)*`.
- *Identifiers:* `[A-Za-z_][A-Za-z0-9_]*`.
- *Phone numbers:* `[2-9][0-9]{2} – [2-9][0-9]{2} – [0-9]{4}`.
- *Dates:* `((1[0-2])|[1-9])/(30|31|([12][0-9])|[1-9])/[1-9][0-9]{0,3}`.
- *US Postal Addresses:*
  `[0-9]+ [NSEW]{0,2} [A-Z][a-z]* (St|Ave|Rd|Ln|Dr|Terr|Blvd),`
  `( [A-Z][a-z]*)*, [A-Z]{2}[0-9]{5}`.

`\b[a-z]{3,4}\b` `[aeiou]ll\b` `[aeiou]{2}` `a.e`

Lord, you have been our dwelling place in all generations.

Coming up:

- Python warm-up assignment (Wed, Sept 3)
- Regular expressions quiz (Fri, Sept 5)
- Read Weizenbaum, excerpt from "Computer Power and Human Reason" (Fri, Sept 5)
- Read J&M, Section 2.9 (Mon, Sept 8)
- Regular expressions programming assignment (Mon, Sept 8)

Next time: Regular expression chatbot *in the lab*.