

Languages

Recall that an *alphabet* is a set of symbols. (And in this course, we almost always use the alphabet of 128 ASCII symbols. All the examples that follow will use this alphabet.) A *string* is a finite sequence of symbols from a given alphabet, such as “abcdef”, “sd%3>n!”, or “COMP314”. A *language* is a set of strings from a given alphabet. Examples of languages include:

- **Binary strings:** strings consisting only of 0s and 1s, such as “1000110”.
- **English words:** strings that appear in (say) the 1989 2nd edition of the Oxford English Dictionary.
- **Any finite set of strings:** for example, the set {apple, banana, orange} is a language.
- **The empty language:** the language that has no strings in it.
- **3 or more g’s:** strings containing at least three g’s, such as “agbgcg”, “gggggggg”, and “ggzxcgyy”.
- **Java programs:** strings that produce no errors when compiled by (some particular version of) the Java compiler.
- **Prime numbers:** numeric strings that represent prime numbers, such as “5” and “31”.
- **Halting Python programs:** strings that represent Python programs that halt on empty input.
- **Strings accepted by someProg.py:** given some fixed Python program `someProg.py`, this language consists of any input string for which `someProg.py` decides Yes.

A Python program *decides* a language L if it decides Yes on all strings in L and No on all strings not in L .