REGULAR AND CONTEXT-FREE LANGUAGES

Natural language comprises a set of words and has syntax, the rules that define the order in which words may be put together, and semantics, the rules that define the relationship between a sentence and real world concepts (rules of semantics allow are meaningful sentences). Difficult to write semantic rules therefore we don't program with natural languages.

Formal languages: do NOT use semantics. Language defined using and alphabet (a finite set of language symbols) and syntax rules.

Meta-language: a language used to describe a real language. Real language consists of strings of characters chosen from a set / alphabet of characters. Two distinct meta-languages

- Regular Expressions Regular
- Backus-Naur Form (BNF) Context-free

Regular expressions

Regular expression: notation that contains strings of characters that can be matched to the contents of a set

A regular language can be defined equally by a regular expression or a finite state machine.

Metacharacters:

| or

? zero or one of the preceding element

***** zero or more of the preceding element

+ one or more of the preceding element

( ) logical grouping part of the expression

**** used to escape metacharacters

^ start of the string or negates a character

$ end of the string

. match any character

[ ] or : [ae] ←→ a | e

specification of a contiguous set of characters e.g. [a-z] or [^a-z]

Context-free language

Regular expressions (→ FSM) work when they are counting or matching symbols where there is a finite limit, when counting and matching is infinite then a context-free grammar is used.

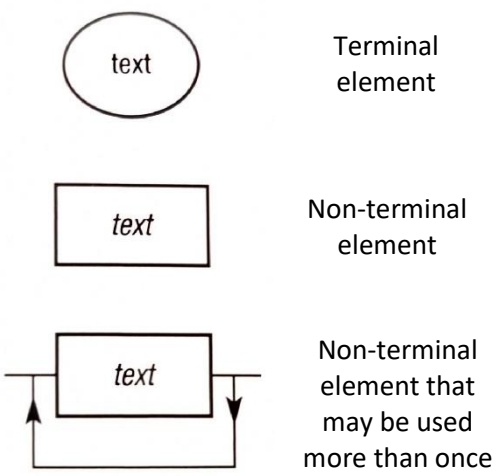
Context-free language: an unambiguous way of describing the syntax of a language, useful where the language is complex.

Backus-Naur Form (BNF) is a form of notation for describing the syntax used by a programming language.

Terminal: in BFN, is the final element that requires no further rules

```
<integer> ::= <digit> | <digit> <integer>
<digit> ::= 0|1|2|3|4|5|6|7|8|9
```

Syntax diagrams: method of visualising rules written in BNF or any other context-free language. It uses directed arrows and the following symbols:



Sets

Set comprehension / set building: creating sets by describing the properties rather than listing the elements

Countable set: finite set, it has a cardinality

Countably infinite set: when the elements can be put into a one-to-one correspondence with the set of natural numbers

Subset: where all the elements of one set are also contained within another set. Proper subset if the sets are not equal