Natural Language Processing

Thema:	Exercise sheet 3: Chart-Parsing with the CKY algorithm
Von:	Timo Baumann, `timo.baumann@oth-regensburg.de`
Datum:	11.04.2024
Abgabe bis:	24.04.2024 24:00
	(there will, however, be a second part with the same due date)

In-class exercises

C.1 Context-free grammars

Work on the following tasks in small groups (3–4 students; **NOT** your submission team members!) for 30 minutes. Take care (everyone!) to understand and be able to follow along what your fellow students are discussing and doing.

Consider the following grammar:

```
  S → NP VP
 NP → N | D N | NP PP
 VP → V NP | V NP PP
 PP → P NP
  N → "I" | elephant | pajamas | man | telescope
  V → shot | lost | held | saw
  P → in | with
  D → my | an | a | the
```

1. Using the grammar, generate 3 different sentences, each with at least 10 words.
2. State all possible derivations of the sentence
   `I saw the man with the telescope`
   Use the bracket notation. (There's a converter between bracket notation and tree visualization at `http://mshang.ca/syntree`.)
3. Does the grammar accept the following sentences? If not, specify how the grammar could be extended to accept both sentences.
   `the I lost my I`
   `the elephant lost all pajamas`
4. Extend the grammar to support attributive adjectives (*A → bold | green | large*). Which other rules do you need to extend (and how) to generate sentences like the following:
   `the green elephant saw a large telescope`

C.2 Chart parsing by hand

1. Remind yourself of the CKY algorithm known from the lecture and how it is applied.
2. Why is the grammar from above unsuitable for use with the CKY algorithm? (Ignore all rules that are unsuitable for the CKY algorithm in the following.)
3. Try to parse the sentence from above (I saw the man with the telescope) with the CKY algorithm.
4. Pay special attention to the ordering of when (and what) data is stored in the chart (and why).

H 3 Take-home exercises

Please hand in the exercise solutions via the Moodle course, one submission per submission team of 2-3 students.

In Moodle, you'll find:

- a grammar similar to the one shown above that can be used for syntax analysis of (a small subset of) English sentences in SRGS format[1].
- a code template, which
    - reads SRGS grammars and yields a corresponding data structure, and
    - represents parsing results (in corresponding data structures) that can be turned into graphs (via GraphViz) which hopefully simplifies your result analyses.
- a minimal code example of how the tepmlate can be used (in `scratch.py`).

1. implement the necessary checks that ensure that a grammar is in Chmosky normal form (`is_CNF` in the class `Grammar`).
2. implement the CKY algorithm so that you can find out whether a given sentence is licensed by the grammar (`is_in_language()` in `parser.py`).
3. extend your implementation of the CKY algorithm so that you generate all possible derivations of the input sentence as a list of (alternative) tree structures (`ParseTrees`). (This should go into `parse()` in `parser.py`).

Comment your code in an appropriate way, specifically with high-level comments up front, as well as detailed comments line-by-line where necessary.

In addition to your code, please hand in a README file and an executable file (e.g. `main.py`).

---

[1] https://www.w3.org/TR/speech-grammar/