Professor Jean-Gabriel GANASCIA
Université Pierre et Marie Curie

Erasmus Mundus
master course in
Data Mining and
Knowledge Managemen
a european master

Logic & Knowledge Representation Course
Exercice

Programming a syntactical parser in PROLOG and with DCG

Deadline: the projects have to be submitted to the Doodle website before Friday, January 5, 2014
Each individual submission is composed of one PROLOG file, the explanations being given as comments.

The goal of this project is to detect lists of strings in a text file. For that purpose, it is required to detect, for each string, the number of its occurrences in a given text. Note that a string is any sequence of characters.

1. Write a PROLOG program `read_file(F, L)` that reads a file `F` and returns the list `L` of ascii codes that compose the file.

2. Using the DCG (Definite Clause Grammar) formalism, build a syntactical parser `detect_noccs_string(F, S, N)` that unifies `N` with the number of occurrences of a string `S` in a file `F`.

3. Generalize this parser to allow the evaluation of the occurrences of all the strings belonging to a list LS of strings. The resulting program `detect_noccs_strings(F, LS, R)` unify `R` with an *association list* that associates to each string of `LS` its number of occurrences.

4. Recall what is a tail recursion and rewrite this program in PROLOG to make it tail recursive.

5. Pretty print the results by showing for each pattern and for each file of a directory the number of its occurrences.

6. Test the resulting programs with entire novels, e.g. with Twain's "Adventures of Huckleberry Finn" and/or with Mary Shelley's "Frankenstein". These texts can be freely downloaded from the Gutenberg project website (http://www.gutenberg.org/).