european
social fund in the
czech republic

EUROPEAN UNION

MINISTRY OF EDUCATION,
YOUTH AND SPORTS

OP Education
for Competitiveness

INVESTMENTS IN EDUCATION DEVELOPMENT

1 Definite Clause Grammars (DCG)

Exercise 1.1: We need a grammar which recognizes/generates the language a^{2n} for $n > 0$. Analyse the behaviour of the following grammars.

```
1.
s --> s, [a,a].
s --> [].

2.
s --> [a,a].
s --> s, [a,a].

3.
s --> [a,a].
s --> [a,a], s.
```

What is the native Prolog representation of the correct grammar?

Exercise 1.2: Write a DC grammar for recognition/generation of the (context-sensitive) language $a^n b^n c^n$ for $n \geq 0$. The grammar should return as its argument an appropriate n for every word generated/recognized.

Exercise 1.3: Write a DC grammar for the recognition of correct arithmetic expressions in the postfix notation containing operators $+, -$ and nonnegative integers. (To be more simple, the grammar can also accept isolated nonnegative integers.)

For example, the grammar should recognize the expression $5\ 2 - 4\ 3\ 2 - + +$. Suppose that the expression is already represented as the appropriate list of terminals: `[5,2,'-',4,3,2,'-','+','+']`.

Extend the grammar so that it evaluates the recognised expression. Extend it further to return the parse tree as one of its arguments.