

# Funkcionální projekt **PLG-2-NKA**

(Funkcionální a logické programování **2016/2017**)

Marek Kidoň  
ikidon@fit.vutbr.cz

29. ledna 2017

## 1 Úvod

Tento dokument popisuje požadavky na projekt **PLG-2-NKA** v předmětu FLP v akademickém roce **2016/2017**. Obecné pokyny pro všechna zadání jsou sepsány ve zvláštním dokumentu, který si nezapomeňte důkladně pročíst.

## 2 Zadání

Vytvořte program převádějící pravé lineární gramatiky (dále PLG) na ne-deterministické konečné automaty (dále NKA).

### 2.1 Rozhraní programu

Program se bude jmenovat **plg-2-nka** a bude jej možné spustit takto:

```
plg-2-nka [volby] [vstup]
```

kde

- **vstup** je jméno vstupního souboru (pokud není specifikováno, program čte standardní vstup) obsahujícího PLG ve formátu popsaném v sekci 3.
- **volby** jsou parametry ovlivňující chování programu, standardně tyto:
  - i dojde pouze k vypsání načtené a do Vaší vnitřní reprezentace převedené PLG na stdout ve formátu popsaném v sekci 3. Nevypisujte jen načtený řetězec, tato volba ověřuje, že Váš program dokáže gramatiku naparsovat a znovu vypsát.
  - 1 dojde k vypsání gramatiky která je výsledkem věty 3.2 z opory předmětu TIN na stdout ve formátu popsaném v sekci 3.
  - 2 dojde k vypsání NKA přijímajícího stejný jazyk jaký je generován PLG na vstupu na stdout ve formátu popsaném v sekci 4. Postupujte podle věty 3.6 z opory předmětu TIN. Stavů automatu označujte nezápornými celými čísly (0, 1, 2, ...).

### 3 Formát vstupu

PLG  $G = (N, \Sigma, P, S)$  na vstupu odpovídá standardní definici pravé lineární gramatiky (viz opora předmětu TIN). Pro zjednodušení je ale abeceda  $N$  podmnožinou množiny velkých písmen  $[A-Z]$ , a abeceda  $\Sigma$  je podmnožinou množiny malých písmen  $[a-z]$ . Epsilon je reprezentováno znakem  $\#$ . Vstupní textová reprezentace PLG o  $N$  pravidlech má následující formát:

```
<seznam všech nonterminálů>\n
<seznam všech terminálů>\n
<počáteční nonterminál>\n
<pravidlo 1>\n
...
<pravidlo N>\n
```

Seznam o  $M$  symbolech má tvar `symbol_1,symbol_2,...,symbol_M`. Pravidlo  $A \rightarrow \alpha$  má tvar `A->alpha`.

Například, reprezentace PLG z příkladu 2.23 opory předmětu TIN je následující:

```
A,B
a,b,c
A
A->aaB
A->ccB
B->bB
B->#
```

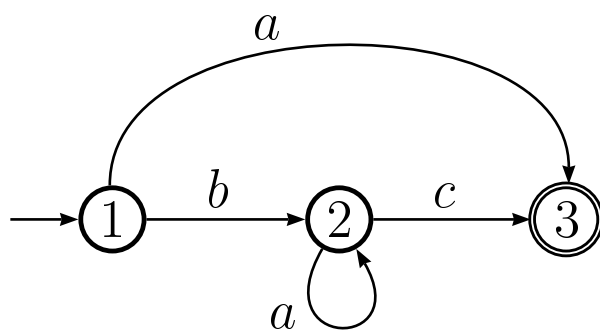
### 4 Formát výstupu

Výstupní textová reprezentace NKA o  $N$  pravidlech má následující formát:

```
<seznam všech stavů>\n
<počáteční stav>\n
<seznam koncových stavů>\n
<pravidlo 1>\n
...
<pravidlo N>\n
```

Seznam o  $M$  stavech má tvar `stav_1,stav_2,...,stav_M`. Pravidlo pro přechod ze stavu 1 do stavu 2 pomocí symbolu  $x$  má tvar `1,x,2`.

Například, NKA:



bude na výstupu jako:

1,2,3  
1  
3  
1,a,3  
1,b,2  
2,a,2  
2,c,3