

Zadanie č. 4

Vo zvolenom (podľa seba) programovacím jazyku zostavte program, ktorý:

1. pre zadanú gramatiku vypočíta pre jednotlivé terminály a neterminály množinu *FIRST* a pre jednotlivé neterminály množinu *FOLLOW*.
2. **(BONUS):** z množín *FIRST*, *FOLLOW* určí rozkladovú tabuľku a zostrojí príslušný syntaktický analyzátor *LL(1)*, ktorý následne dokáže spracovať zadané slovo. Počas jeho spracovania vypisuje používané pravidlá pri expanzii na výstup. Ak vstupná gramatika **nie je LL(1)**, tiež to musí oznámiť na výstupe. V takom prípade samozrejme slovo spracúvať nebude.

Vstupy a výstupy

Vstupom programu je bezkontextová gramatika, načítavaná zo súboru „**grammar.ctf**“

V prípade riešenia bonusu je vstupom taktiež slovo, ktoré má spracovať zostrojený analyzátor. Na zadanie slova **vyzve program** používateľa počas behu.

Výstupom programu je súbor „**rules.ctf**“, ktorý obsahuje popis množín *FIRST* a *FOLLOW* v prehľadnom formáte. V prípade riešenia bonusu obsahuje súbor navyše aj **rozkladovú tabuľku** príslušného *LL(1)* analyzátora.

LL(1) analyzátor počas svojej činnosti **musí** vypisovať čísla pravidiel používaných pri expanzii (pravidlá sú implicitne očíslované od prvého číslom 1). Na záver oznámi, či vstupné slovo patrí do jazyka gramatiky alebo nie.

Implementácia algoritmov

Implementujte algoritmy z prednášok! T.j. **je nutné**, aby v zdrojovom kóde boli priamo algoritmy z prednášok **implementované**, nie len prilinkovaná nejaká externá knižnica, z ktorej sa zavolá príslušná funkcia.

Formát vstupu a výstupu

Vstupný súbor obsahujúci príslušnú gramatiku môže mať 2 špeciálne formáty, vysvetlené na s. 3, resp. 5 tohto zadania. **Jeden z nich si vyberte a implementujte Váš program tak, aby pracoval s daným formátom!.** Výstupný súbor nech je prehľadným zobrazením množín *FIRST*, *FOLLOW* a rozkladovej tabuľky.

Bodovanie

- Základné zadanie je za 10 bodov - 5 bodov za zostrojenie množiny *FIRST* a 5 body za zostrojenie množiny *FOLLOW* (hodnotené podľa výstupného súboru).
- Za bonus je 10 bodov - 5 bodov za zostrojenie rozkladovej tabuľky a 5 bodov za činnosť *LL*(1) analyzátora. (hodnotené podľa výstupného súboru a podľa výstupov analyzátora).
- Za typ formátu vstupného súboru nie sú žiadne bonusové body.

Čo odovzdať do AIS

Do AIS odovzdajte zdrojové kódy Vášho programu spolu s textovým súborom, v ktorom napíšete, či program spracováva prvý formát vstupu/výstupu (s. 3 tohto dokumentu), alebo druhý formát vstupu/výstupu (s. 5 tohto dokumentu).

Deadline zadania je nedeľa, 22. 5. 2016, 24:00 (polnoc na pondelok).

FORMÁT SÚBORU v. 1

Na prvom riadku sa nachádza číslo, udávajúce počet neterminálov.

Na druhom riadku sa nachádza číslo, udávajúce počet terminálov.

Na treťom riadku sa nachádza číslo, udávajúce počet pravidiel.

Na nasledujúcich riadkoch sa nachádzajú mená neterminálov, každé meno na novom riadku. Pre jednoduchosť uvažujte, že mená neterminálov sú veľké písmená abecedy. Počiatočný neterminál má **vždy** označenie S.

Na nasledujúcich riadkoch sa nachádzajú mená terminálov, každé meno na novom riadku. Pre jednoduchosť uvažujte, že terminály sú malé písmená abecedy a že každý terminál má max. 1 znak.

Na nasledujúcich riadkoch po nich sa nachádzajú jednotlivé symboly vstupnej abecedy.

Na nasledujúcich riadkoch po nich sa nachádzajú pravidlá danej gramatiky, pričom sú zapísané v tvare:

(neterminál->pravá strana pravidla)

Pravidlo obsahujúce na pravej strane **epsilon** je zapísané ako pravidlo, ktoré má na pravej strane číslicu **nula**. Každé pravidlo je zapísané na novom riadku. Tá šípka je textovo reprezentovaná ako symbol mínus (–) za ktorým ide symbol väčší než (>). Medzi neterminálom a šípkou, resp. šípkou a pravou stranou pravidla **nie je medzera**.

Na nasledujúcej strane je ukážka.

Gramatika

$$S \rightarrow ABc$$

$$A \rightarrow a \mid \varepsilon$$

$$B \rightarrow b \mid \varepsilon.$$

by bola zapísaná ako:

3

3

5

S

A

B

a

b

c

S->ABc

A->a

A->0

B->b

B->0

FORMÁT SÚBORU v. 2 - XML verzia z jFlapu verzie 7

1. Vytvorte si v jFlape automat, uložte ho (.jff) a pozrite sa, v akom formáte je výsledný súbor, t.j. ako vyzerá XMLko, generované jFlapom.
2. Napíšte Váš program tak, aby vstup načítaval zo súboru, ktorého formát bude spĺňať XML formát výstupu jFlapu VERZIE 7.

Ukázka činnosti programu

Vstupný súbor:

```
3
3
5
S
A
B
a
b
c
S->ABc
A->a
A->0
B->b
B->0
```

Výstupný súbor: (0 predstavuje prázdny reťazec, t.j. ε)

```
FIRST:
S: a,b,c
A: a,0
B: b,0
a: a
b: b
c: c
```

```
FOLLOW:
S: 0
A: b,c
B: c
```

Výstupný súbor navyše obsahuje v prípade bonusu

```
RT: a b c
S:  1 1 1
A:  2 3 3
B:  - 4 5
```

a počas svojej činnosti by program vypísal na výstup pre slovo *abc*:

Vstup: abc

Cinnost:

E1

E2

P

E4

P

P

AKCEPTUJEM

prípadne pre slovo *aa*:

Vstup: aa

Cinnost:

E1

E2

P

ZIADNE DOSTUPNE PRAVIDLO

NEAKCEPTUJEM