

Fordítóprogramok (A,C,T szakirány)

Feladatgyűjtemény

ELTE IK

1 Lexikális elemzés

1. Add meg reguláris nyelvtannal, reguláris kifejezéssel és véges determinisztikus automatával a következő lexikális elemeket!
 - (a) egész szám (legalább egy számjegy 0-9-ig)
 - (b) olyan egész szám, amely több számjegy esetén nem kezdődhet nullával
 - (c) előjeles egész szám (opcionális + vagy - az elején)
 - (d) törtszám (tizedespont előtt legalább egy számjegy)
2. Add meg reguláris kifejezéssel és véges determinisztikus automatával a következő lexikális elemeket!
 - (a) azonosító (betűvel kezdődik, számmal vagy betűvel vagy _ jellel folytatódik)
 - (b) egysoros megjegyzés //-től a sor végéig
 - (c) többsoros megjegyzés /*-tól */-ig
 - (d) sztring
 - "alma"
 - "a \" egy idézőjel a sztringben"
 - "a \\ egy backslash a sztringben"
 - backslash után csak idézőjel vagy backslash állhat
 - (e) fehér szóközök (legalább egy *space*, *tab* vagy *sorvége*)
3. Rajzold fel a lexikális elemző véges determinisztikus automatáját, ha a következő szimbólumokat szeretnénk felismerni:

```
++ += -- -= -> >= >> >>= <<=
```
4. A következőkben az ábécé a $\{a, b\}$ halmaz.
 - (a) Ha a változók egybetűsek, *ab* és *aba* kulcsszavak, milyen lexikális elemekre kell felbontani az *abab* szöveget?
 - (b) Ha a változók egybetűsek, *aa*, *ab* és *aba* kulcsszavak, milyen lexikális elemekre kell felbontani az *aabaa* szöveget?
 - (c) Hogyan kell ezt lexikális elemző generátor (pl. *flex*) segítségével implementálni?
5. Készíts programot a + ++ += és - szimbólumok felismeréséhez (lexikális elemző generátor használata nélkül!)
 - (a) Készítsd el az egyetlen szimbólum felismerésre képes automata implementációját!
 - (b) Fejleszd tovább a programot egy teljes lexikális elemzővé, ami több egymást követő szimbólumot is képes felismerni!
6. Lexikális elemző generátor (pl. *flex*) segítségével készíts programot az alábbi feladatokra!
 - (a) a teljes bemenetet a kimenetre másolni

- (b) a szövegben a *username* szót a saját felhasználói nevedre cserélni
- (c) programkód \rightsquigarrow HTML konverter
 - *tab* \rightsquigarrow ` `;
 - *sorvége* \rightsquigarrow `
`
 - `<` \rightsquigarrow `<`;
 - `>` \rightsquigarrow `>`;
- (d) minden sor végén kiírni a sor számát és hosszát
- (e) minden szóhoz kiírni a pozícióját (sor, oszlop) és magát a szót

7. Egy nyelvben a *begin* és az *end* kulcsszavak, a változók betűkből és számjegyekből állnak, de csak betűvel kezdődhetnek, a számkonstansok pedig számjegyekből állhatnak. A fehér szóközök szóközökből, tabokból és sorvégekből állhatnak.

- (a) Milyen lexikális elemekre bontja ennek a nyelvnek a lexikális elemzője az alábbi szöveget?

```
alma123 beginend begin 12
```

- (b) Ha ezt az elemzőt *flex* segítségével írjuk, milyen sorrendben kell megadni az egyes lexikális elemek reguláris kifejezéseit?

8. Egyes programozási nyelvekben a hexadecimális (16-os számrendszerben ábrázolt számok) a következő alakúak:

- a "számjegyek": 0, 1, ..., 9, *A*, *B*, ..., *F*
- a szám nem kezdődhet betűvel (*A*, *B*, ..., *F*)
- a szám végén a *h* karakter áll

Példák helyes hexadecimális számokra: *12AFh 123h 0h 0DEh*

Példák helytelen hexadecimális számokra: *DEh D1h h 0hh 123*

- (a) Írd fel az ilyen hexadecimális számokat leíró reguláris kifejezést!
- (b) Rajzold fel az ilyen hexadecimális számokat felismerő véges determinisztikus automatát!
- (c) Sorszámold az automata állapotait! Írd fel azt az állapotsorozatot, amin az automata keresztül megy, miközben a *0DEh* számot felismeri!

9. Egy nyelvben a változók betűkből és számjegyekből állnak, és felkijáltójellel kell kezdődniük. A számkonstansok számjegyekből, a fehér szóközök pedig szóközökből, tabokból és sorvégekből állhatnak. A nyelvben értékadás és elágazás van az alábbi példaprogram szerint:

```
!x1 := 5
ha !x1 paros akkor !2y := 4
ha !2y paratlan akkor !x1 := 2
```

Az elágazás feltétele csak párosságot és páratlanságot tud vizsgálni, az elágazás belsejében pontosan egy értékadás lehet.

- (a) Határozd meg a nyelv lexikális elemeit és írd le őket reguláris kifejezéssel!
- (b) Milyen sorrendben kell ezeket megadni *flex*-ben és miért így?
- (c) Készíts a változóneveket felismerő véges determinisztikus automatát!

10. Készíts reguláris kifejezéseket és véges determinisztikus automatákat az alábbi lexikális elemekhez!

- (a) *a*, *b* és *c* betűkből és pontokból álló nem üres sorozat
- (b) *a*, *b* és *c* betűkből és pontokból álló nem üres sorozat, amely nem kezdődhet és nem végződhet ponttal és két pont nem állhat egymás mellett

(c) a , b és c betűkből és pontokból álló tetszőleges sorozat, amelyben az egymás mellett álló betűknek ábécé-sorrendben kell lennie

- példák helyes elemekre: ϵ \dots a $b.a.$ $.aabcc.bc$
- példák helytelen elemekre: ba $.c.ba.$

11. Egy nyelv kulcsszavai legyenek $abba$ és abc , változónevei pedig a 10. feladat (c) pontjának megfelelő lexikális elemek.

(a) Milyen sorrendben kell feltüntetni ezeket a reguláris kifejezéseket Flex-ben?

(b) Milyen lexikális elemekre bontja az elemző az alábbi szövegeket?

- $abbaabc$
- $abbaabc.$
- $.abba.abc$

2 LL elemzések

1. Döntsd el, hogy az alábbi nyelvtanok közül melyik egyszerű LL(1)-es!

(a) $S \rightarrow aSA \mid A$
 $A \rightarrow bA \mid a$

(b) $S \rightarrow aSA \mid bA$
 $A \rightarrow \epsilon \mid bA$

(c) $S \rightarrow aSA \mid bA$
 $A \rightarrow a$

Amelyik igen, ahhoz készítsd el az elemző táblázatot és elemezd az alábbi szövegeket!

- $aabaaa$
- $aabbaa$
- $aabaa$

2. Döntsd el, hogy az alábbi nyelvtanok közül melyik ϵ -mentes LL(1)-es!

(a) $S \rightarrow aSA \mid A$
 $A \rightarrow bA \mid a$

(b) $S \rightarrow SA \mid Bc$
 $A \rightarrow aA \mid c$
 $B \rightarrow b \mid A$

(c) $S \rightarrow dSdS \mid A \mid eB$
 $A \rightarrow a \mid BA$
 $B \rightarrow b \mid c$

Amelyik igen, ahhoz készítsd el az elemző táblázatot! Válassz egy olyan szöveget, amelynek levezetéséhez mindegyik szabályt pontosan egyszer kell felhasználni és elemezd!

3. Milyen mondatai vannak az alábbi grammatika által definiált nyelvnek?

$S \rightarrow [A]$
 $A \rightarrow \epsilon \mid nB$
 $B \rightarrow \epsilon \mid ,nB$

Mutasd meg, hogy ez egy LL(1) nyelvtan, készítsd el az elemző táblázatot, és elemezz egy 5 szimbólumból álló mondatot. (Hány 5 szimbólumból álló mondat van a nyelvben?)

4. Elemzőt szeretnénk készíteni egy nyelvhez, aminek egy példaprogramja:

```

if ?a1 then
  ?a1 := false
endif
while ?a3 do
  ?a3 := ?a2
  ?a2 := ?a1
done

```

A nyelvben minden változó ? jellel kezdődik, deklarálni nem kell őket, mindegyik logikai típusú (**true**, **false**). A nyelvben van értékadás, elágazás és ciklus. Az értékadás baloldalán változó, jobboldalán változó, **true** vagy **false** állhat. A ciklusok és elágazások feltétele mindig egyetlen változó. Az egész program és a ciklusok, elágazások törzse is lehet üres.

- (a) Határozd meg a lexikális elemeket és írd hozzájuk reguláris kifejezéseket!
- (b) Írd fel a nyelvtan szabályait!
- (c) Ellenőrizd, hogy LL(1)-e a nyelvtan!
- (d) Készíts hozzá rekurzív leszállásos elemzőt!

5. *C++* függvénydeklarációt szeretnénk szintaktikus elemzővel elemezni.

A lexikális elemek legyenek: a () , ;

A nyelvtan szabályai pedig:

$S \rightarrow \underline{aa}(L);$

$L \rightarrow \epsilon \mid \underline{aa}F$

$F \rightarrow \epsilon \mid ,\underline{aa}F$

- (a) Igaz-e, hogy a grammatika egyszerű LL(1), ϵ -mentes LL(1), általános LL(1)?
- (b) Készítsd el hozzá az elemző táblázatot és elemezd az aa(aa,aa); szöveget!
- (c) Milyen eljárás tartozik a nyelvtan rekurzív leszállásos elemzőjében az *L* nemterminális szimbólumhoz?

6. Készíts elemzőt a 9. feladatban adott nyelvhez!

- (a) Írd fel a nyelv nyelvtanát!
- (b) Készíts LL elemzőt a nyelvtanhoz!
- (c) Elemezd a **ha 2 paros akkor !x := 0** mondatot!
 - Írd fel hozzá a lexikális elemző kimenetét!
 - Elemezd az elemző táblázat segítségével!

7. Egy olyan programozási nyelvhez szeretnénk szintaktikus elemzőt írni, amelyikben három lexikális elem van: begin, end és skip.

A nyelv szintaxisát a következő nyelvtan adja meg:

$S \rightarrow \underline{skip} \mid BS$

$B \rightarrow \underline{begin} S \underline{end}$

- (a) Igaz-e, hogy a grammatika egyszerű LL(1), ϵ -mentes LL(1), általános LL(1)?
- (b) Készítsd el hozzá az elemző táblázatot és elemezd a begin skip end skip szöveget!

3. LR elemzések

1. Írj fel nyelvtanokat az alábbi nyelvi elemek szintaxisának leírásához! Elemzőgenerátor (pl. *bisonc++*) segítségével készíts hozzájuk szintaktikus elemző programokat!

- (a) Lista

[alma, körte, szilva]

Terminális szimbólumok: nyitó- és csukó zárójel, vessző, listaelem

- (b) C stílusú függénydeklarációk sorozata

```
char betuje( string s, int index );  
int osszeg( int x, int y );
```

Terminális szimbólumok: azonosító, nyitó- és csukó zárójel, vessző
(A típusok, függvények és paraméterek neve egyaránt azonosító.)

- (c) Blokkstrukturált nyelv *skip* utasítással.

```
begin  
  skip  
  begin end  
end  
begin  
  begin skip end  
  skip skip skip  
end
```

Terminális szimbólumok: begin, end és skip

- (d) A nulladrendű logika nyelve

```
(a <-> b) -> (!a | b) & true
```

Terminális szimbólumok: true, false, változónév, nyitó- és csukó zárójelek, továbbá az öt logikai összekötőjel a precedencia szerint csökkenő sorrendben: negáció, konjunkció, diszjunkció, implikáció, ekvivalencia

2. Adott az alábbi nyelvtan:

$$S \rightarrow \underline{x} \mid AS$$
$$A \rightarrow \underline{a} S \underline{b}$$

- Alakítsd át kiegészített nyelvtanná!
- Számold ki az $LR(0)$ kanonikus halmazait!
- Rajzold fel az elemző automatáját és töltsd ki az elemző táblázatot!
- Elemezd az axbx szöveget!

3. Adott a következő nyelvtan:

$$S' \rightarrow S$$
$$S \rightarrow \epsilon \mid \underline{\text{skip}} S \mid BS$$
$$B \rightarrow \underline{\text{begin}} S \underline{\text{end}}$$

- Mutasd meg, hogy a nyelvtan nem $LR(0)$! Milyen konfliktusok vannak az $LR(0)$ elemző táblázatában?
- Igaz-e, hogy a nyelvtan $SLR(1)$? Számold ki a szükséges *FOLLOW* halmazokat és töltsd ki az $SLR(1)$ elemző táblázatot!
- Elemezd a skip begin begin end szöveget és figyeld meg, hol jön rá az elemző a hibára!

4. Adott a következő nyelvtan:

$$S' \rightarrow S$$
$$S \rightarrow \underline{\text{id}} \mid \underline{\text{id}} + S$$

- Mutasd meg, hogy a nyelvtan nem $LR(0)$! Milyen konfliktusok vannak az $LR(0)$ elemző táblázatában?
- Igaz-e, hogy a nyelvtan $SLR(1)$? Számold ki a szükséges *FOLLOW* halmazokat és töltsd ki az $SLR(1)$ elemző táblázatot!
- Elemezd az id + + id szöveget és figyeld meg, hol jön rá az elemző a hibára!

5. Adott a következő nyelvtan:

$$S' \rightarrow S$$
$$S \rightarrow B \underline{=} J \mid J$$
$$B \rightarrow * J \mid \underline{\text{id}}$$
$$J \rightarrow B$$

- (a) Mutasd meg, hogy a nyelvtan nem $SLR(1)$!
- (b) Igaz-e, hogy a nyelvtan $LR(1)$?
- (c) Készítsd el az $LR(1)$ elemző táblázatot és elemezd az id=*id szöveget!
6. Készíts $LALR(1)$ elemzőt az alábbi nyelvtanhoz!
- $$S' \rightarrow S$$
- $$S \rightarrow \underline{x} \mid y S y$$
7. Adott a következő nyelvtan:
- $$S' \rightarrow S$$
- $$S \rightarrow A\underline{b} \mid \underline{a}c \mid \underline{d}A\underline{c}$$
- $$A \rightarrow \underline{a} \mid \underline{a}A$$
- (a) Számítsd ki az $LR(0)$ kanonikus halmazokat!
- (b) Add meg az összes nemterminálishoz a $FOLLOW_1$ halmazait!
- (c) Az előzőek segítségével töltsd ki az $SLR(1)$ elemző táblázatot! Jelöld a táblázatban a konfliktusokat!
- (d) Számítsd ki az $LR(1)$ kanonikus halmazokat!
- (e) Melyek az összevonható halmazok?
- (f) Töltsd ki az $LALR(1)$ elemző táblázatot!
- (g) Elemezd az ab mondatot!

4 Assembly programozás

- Definiálj olyan adatterületet, ami az X és Y címkéktől kezdve 4-4 bájtban a 10 és 5 értékeket tárolja, és a Z címkétől kezdve 4 bájtban lefoglal kezdeti érték nélkül.
Írj olyan asszemly kódreszletet, ami
 - megnöveli az X címkétől kezdődő 4 bájt értékét
 - átmásolja az Y-től kezdődő 4 bájtban az X-től kezdődő 4 bájtba
 - kiszámolja a szorzatukat a Z-től kezdődő 4 bájtba
 - bitenként összeeseli őket és az eredményt a Z-től kezdődő 4 bájtba teszi
- Írj olyan kódreszletet, ami az abszolút értékét számolja ki eax értékének!
- Írj olyan kódreszletet, ami bekér két számot és a másodikat kiírja annyiszor, amennyi az első értéke.
- Írj olyan kódreszletet, ami kiszámolja két pozitív szám legnagyobb közös osztóját!
- Írj programot, ami egészeket olvas be, tárolja őket tömbben tömbben, majd kiírja a tömb tartalmát.
- Írj programot, ami tömbben tárolt számokat összegez.
- Írj programot, ami egy karaktertömbben megszámlálja az 'a' karaktereket.
- Írj programot, ami egészek tömbjében megkeresi a legnagyobb számot.
- Írj programot, ami bekér egy nemnegatív egész számot, majd átalakítja szöveggé (10-es számrendszer szerint). Az eredményt egy karaktertömbbe kell írni majd kiírni a képernyőre.

5 Kódgenerálás

1. Írd fel a szimbólumtábla tartalmát (változó neve, típusa, címkéje) majd rögzített kódgenerálási szabályok alapján add meg a generált assembly kódot az alábbi kódrészlethez:

```
int x;  
int y;  
x = 2*x+y;
```

2. Rögzített kódgenerálási szabályok alapján add meg a generált assembly kódot az alábbi kódrészlethez:

```
while(x < 10)  
  if( b )  
    x = x+1;  
  else  
    x = 10;
```