

Feladatgyűjtemény a “Formális nyelvek és fordítóprogramok” tárgyhoz

Szavak és nyelvek műveletei

1. Az alábbiak közül mely szavak állíthatók elő összefűzés (konkatenáció) és megfordítás műveletek segítségével néhány további felsorolt szóból?

abcd, dcba, aa, 321b2, b123, b2, aab2, 321321b321, 123, abc, dcbaaaaaabcd

2. Sorolja fel az összes, legfeljebb 6 hosszú szót, ami az alábbi kettőből előállítható összefűzés (konkatenáció) és megfordítás műveletek segítségével!



3. Legyen $V = \{a, b, c\}$ és legyen $u_1 = cca$, $u_2 = aabc$ egy-egy V feletti szó.
Soroljuk fel u_1 és u_2 valódi részzavait, adjuk meg a hosszukat, konkatenáltjukat, tükörképüket, 0-adik, 1., 2. és 3. hatványukat!

4. Adja meg lexikografikus rendezés szerint az alábbi nyelv szavait!

\{x, \epsilon, abc, xy, yx, bca, a, c, ca, ba, xabc, bxyz\}

5. Az ábécé legyen az $\{a, b\}$ halmaz! Adja meg a legbővebb olyan nyelvet, amelyben

- pontosan 3 hosszú szavak vannak;
- legfeljebb 2 hosszú szavak vannak;
- pontosan 1 hosszú szavak vannak;
- legfeljebb 0 hosszú szavak vannak!

6. Adja meg az alábbi két nyelv egyesítését (unióját), összefűzését (konkatenációját) és metszetét!

\{@\#, !\%\%, \{\&\@, @\#, !\%\}

7. Adja meg az alábbi nyelv nulladik, első, második és harmadik hatványát!

\{x, xy, yxx\}

8. Adjon meg olyan nyelveket, amelyeknek második hatványa egyenlő önmagukkal! Hány ilyen tulajdonságú véges nyelv van?

9. Melyik nyelv(ek)nek a második hatványa a $\{\varepsilon, 0, 1, 00, 01, 10, 11, 100, 010, 101, 110, 1010\}$?

10. Melyik nyelv(ek)nek a harmadik hatványa a $\{aaa, aabc, abca, abcbc, bcaa, bcabc, bcbca, bcbcbc\}$?

11. Adja meg az alábbi nyelvek lezártját! (Elég a legfeljebb 6 hosszú szavaikat felsorolni.)

- $\{\}$
- $\{\varepsilon\}$
- $\{a\}$
- $\{x, yz\}$

12. $L_1 = \{ab, b\}$, $L_2 = \{ab^n \mid n \in \mathbb{N}\}$. Határozzuk meg az alábbi nyelveket!

- $L_2 \setminus L_1$
- $L_1 \setminus L_2$
- $L_2 \cap L_1^*$
- $L_2 \setminus L_1^*$

13. $L_1 = \{ab, ba, b\}$, $L_2 = \{aba, a\}$, $L_3 = \{a^n b^n \mid n \in \mathbb{N}\}$. Határozzuk meg az alábbi nyelveket!

- $L_1 L_3 \cup L_2 L_3$
- $L_1^* \cap L_2^*$
- $L_1^* \setminus L_3$
- $(L_1 \cup L_2)^* \cap L_3$

14. Döntsük el, hogy az alábbi nyelvek végesek vagy végtelenek! A végteleneket kezdjük el felsorolni lexikografikusan!

- \emptyset (üres nyelv)
- $\{\varepsilon\}$ (üres szót tartalmazó nyelv)
- $\{a^n b^n \mid n \geq 0\}$
- $\{a\}^* \{b\}^*$
- $\{a^n b^k \mid n \geq 0, k \geq 0\}$
- $\{u \in \{a, b\}^* \mid u = u^{-1}\}$ (palindrom szavak)

- $\{ uu^{-1} \mid u \in \{a,b\}^* \}$ (szimmetrikus szavak, ami részhalmaza a palindromáknak)

Nyelvtanok

Online eszközök:

- <http://mdaines.github.io/grammophone/#>
- <https://web.stanford.edu/class/archive/cs/cs103/cs103.1156/tools/cfg/>

1. Mely szavak vezethetők le az alábbi nyelvtanokból? (A kezdőszimbólumot S jelöli.)
 - a. $S \rightarrow aA$
 $A \rightarrow c \mid bA$
 - b. $S \rightarrow \varepsilon \mid aaSb$
 - c. $S \rightarrow YSb$
 $S \rightarrow V$
 $YV \rightarrow Vaa$
 $YV \rightarrow aaa$
2. Készítsünk nyelvtant, mely azokat az $u \in \{a, b\}^*$ szavakat fogadja el, melyek a -val kezdődnek és b -vel végződnek!
3. Készítsünk nyelvtant a 4-gyel osztható bináris számok nyelvéhez!
4. Készítsünk nyelvtant, mely $\{a^n b^n \mid n \geq 1\}$ szavait fogadja el!
5. Készítsünk nyelvtant ahhoz az a, b betűk feletti nyelvhez, melynek szavai ugyanannyi a -t és b -t tartalmaznak!
6. Készítsünk nyelvtant ahhoz az a, b betűk feletti nyelvhez, melynek szavai palindrómák (megegyeznek a megfordításukkal)!
7. Készítsünk nyelvtant a helyes zárójelezés leírására, $abc: \{ (,) \}$
8. Készítsünk nyelvtant, mely $\{a^n b^n c^n \mid n \geq 1\}$ szavait fogadja el!
9. $T = \{a, b, c, d\}$. $L = \{a^n b^n u \mid n \in \mathbb{N}, l_a(u) = 1, u \in \{a, c, d\}^*\}$.
 Generáljuk L -et nyelvtannal! Milyen típusú ez az L -t generáló nyelvtan?
10. $T = \{a, b, c, d\}$. $L = \{(ba)^n u (ab)^n \mid n \in \mathbb{N}, l_d(u) = 2, u \in \{b, c, d\}^*\}$.
 Generáljuk L -et nyelvtannal! Milyen típusú ez az L -t generáló nyelvtan?

11. Adjunk nyelvtant, amely az alábbi függvénykifejezéseknek megfelelő jelsorozatokat generálja!

Egy függvénykifejezés egy azonosítóval kezdődik és zárójelben egy vagy több argumentuma lehet. Az argumentumokat vessző választja el. Argumentum egy azonosító vagy függvénykifejezés lehet. Az azonosító betűk sorozata lehet.

Példák függvénykifejezésekre: $\sin(f(x,y),z)$ $f(\text{alma})$

Reguláris kifejezések

Online eszköz: <https://regex101.com/>

1. Mely alábbi szavakra illeszkednek az alábbi reguláris kifejezések?

Reguláris kifejezések:

`""`, `a`, `ab`, `a|b`, `a*`, `a|b*`, `ab*`, `a*b`, `a*b*`, `a*/b*`, `(ab)*`

Szavak:

`ε`, `a`, `b`, `aa`, `ab`, `ba`, `aaa`, `aab`, `aba`, `abb`, `baa`, `bab`, `bba`, `bbb`, `aaaaaa`, `bbbbbb`, `ababab`, `babab`

2. Mely szavakra illeszkedik az `a?` reguláris kifejezés? Írjuk fel a standard műveletek segítségével!
3. Mely szavakra illeszkedik az `a+` reguláris kifejezés? Írjuk fel a standard műveletek segítségével!
4. Mely szavakra illeszkedik az `[a-d]` reguláris kifejezés? Írjuk fel a standard műveletek segítségével!
5. Mely szavakra illeszkedik az `^a-z` reguláris kifejezés? Kezdjük el felírni a standard műveletek segítségével!
6. Adjon meg reguláris kifejezést a következő számformátumokhoz!
- Decimális egész szám (legalább egy számjegy 0-9-ig)
 - Olyan decimális egész szám, amely több számjegy esetén nem kezdődhet nullával
 - Előjeles decimális egész szám (opcionális + vagy – az elején)
 - Decimális törtszám (tizedespont előtt legalább egy számjegy)
7. Készítsen reguláris kifejezést a helyes `hh:mm` óraformátum ellenőrzésére, ahol a `hh` a `00..23`, míg az `mm` a `00..59` értékeket veheti fel!
8. Adjon meg reguláris kifejezést a következő azonosítókhoz! (Betűk alatt az angol ábécé kis- és nagybetűit, számjegyek alatt a decimális számjegyeket értjük.)
- Betűvel kezdődik, számjeggyel, betűvel vagy `_` jellel folytatódik
 - Szigorítás: az utolsó karakter nem lehet `_`
 - További szigorítás: nem lehet egymás mellett két `_`
9. Adjon meg reguláris kifejezést a következő, trükkösebb azonosítókhoz!
- Az első fele csak betűket illetve `_` jelet tartalmazhat, második fele pedig csak számjegyeket
 - Szigorítás: Ha a betűkből álló rész legelső és legutolsó karaktere is `_` akkor közte csak nagybetűk szerepelhetnek, egyébként pedig csak kisbetűk
 - További szigorítás: Ha a számjegyekből álló rész első számjegye páros akkor utána még páros darab számjegy következhet, ha az első páratlan akkor pedig még páratlan darab.
10. Adjon meg reguláris kifejezést a következő, furcsa azonosítókhoz! Az azonosító csak az `a`, `b`, `c` betűket tartalmazhatja (akár többször is, de nem kötelező mindegyiket), viszont a betűknek be

kell tartania az abc sorrendet, azaz egy a előtt soha nem szerepelhet b vagy c , illetve egy b előtt nem szerepelhet c .

- Helyes: abc , $aaabcc$, ac , c
- Helytelen: bac , $abcb$, $bc bc$

11. Adjon meg reguláris kifejezést egysoros megjegyzésekhez, amely $//$ -től a sor végéig tartanak!

12. Adjon meg reguláris kifejezést többsoros megjegyzésekhez, amelyek $/*$ -tól $*/$ -ig tartanak!

- Helyes: $/**/, /*ab"+!%/.*/$, $/***/$, $/*//////*/$
- Helytelen: abc , $*$, $/$, $/*$, $*/$, $/*/*$, $/*abc*/abc*/$

13. Adjon meg reguláris kifejezést sztring leírására! A sztringek idézőjellel kezdődnek és végződnek, a belsejükben tetszőleges karakterek szerepelhetnek az alábbiak betartásával:

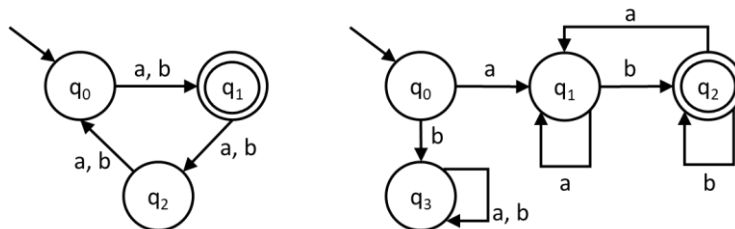
- Idézőjel nem állhat belül, kizárólag \backslash után.
- $\\$ és \backslash szabályosak a sztring belsejében, más kombinációban \backslash nem szerepelhet.
- Példák: $"alma"$, $"a \backslash"$ egy idézőjel a sztringben", $"a \backslash \backslash"$ egy backslash a sztringben"

14. Adjon meg reguláris kifejezést fehér szóközök (space, tab vagy sorvége) nem üres sorozataira!

Véges automaták

1. Mely alábbi szavakat fogadják el az alábbi véges determinisztikus automaták?

Automaták:



Szavak:

ϵ , a , b , aa , ab , ba , aaa , aab , aba , abb , baa , $aaaa$, $abab$

2. Adjon meg az alábbi számformátumokat elfogadó véges determinisztikus automatákat!

- Decimális egész szám (legalább egy számjegy 0-9-ig)
- Olyan decimális egész szám, amely több számjegy esetén nem kezdődhet nullával
- Előjeles decimális egész szám (opcionális $+$ vagy $-$ az elején)
- Decimális törtszám (tizedespont előtt legalább egy számjegy)

3. Adjon az előző órán már megismert többsoros komment felismerésére véges determinisztikus automatát!

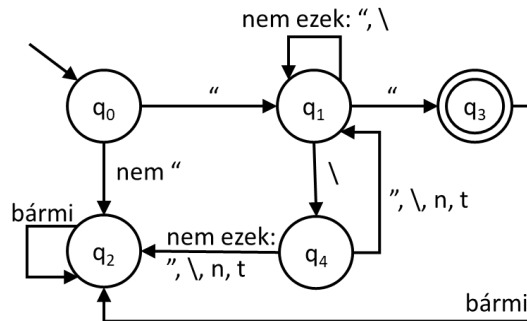
4. Fejezze ki az alábbi reguláris kifejezéseket véges determinisztikus automatával illetve reguláris nyelvtannal is!

- $a?$

- a^+
- $_{[0-9]}$
- $(a|b|c)d^*$

5. Készítsen programot, amely egy végtelen ciklusban a standard inputról olvas be egy-egy sornyi szöveget! A program mindegyik sorról döntse el, hogy az alábbi véges determinisztikus automata elfogadja-e, és az eredményt írja a standard outputra! Az automata működését egy külön függvényben valósítsa meg, amely egy stringet kap paraméterként, és egy logikai értékkel tér vissza! Ennek a függvénynek két változatát is készítse el:

- Az egyikben az automata logikáját egymásba ágyazott elágazásokkal,
- a másikban egy táblázattal valósítsa meg! (A táblázatnak legyen egy-egy sora az automata átmenetein szereplő néhány karakternek, és további egy sora az összes többi karakternek!)



6. Készítsünk véges determinisztikus automatát, mely az alábbi nyelvtan által generált nyelvet fogadja el!

$S \rightarrow aA \mid aC \mid bB \mid \varepsilon$

$A \rightarrow aC \mid bS \mid bC$

$B \rightarrow bS \mid bC \mid \varepsilon$

$C \rightarrow aS \mid aC \mid bB$

7. Értsük meg, majd próbáljuk minimalizálni az alábbi automata állapotszámait!

	a	b
-> 1	7	7
2	1	2
3	9	1
<- 4	1	4
<- 5	6	4
<- 6	6	2
7	9	8
<- 8	8	2
9	1	8

Környezetfüggetlen nyelvtanok

1. Írjon fel környezetfüggetlen nyelvtant az alábbi alakú listák szintaxisának leírására:

[alma, körte, szilva]

A nyelvtan terminális szimbólumai legyenek a következők:

n (nyitó szögletes), *e* (lista eleme), *v* (vessző), *c* (csukó szögletes)

A fenti példa tehát a *nevevec* szónak felel meg. További helyes szavak: *nc, nec, nevec* stb.

Rajzolja fel a fenti példa szintaxisfáját, és adja meg a szó levezetését!

2. Írjon fel környezetfüggetlen nyelvtant C stílusú deklarációsorozatok szintaxisának leírására:

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

Terminális szimbólumok:

a (azonosító: típusok és paraméterek nevei), *n* (nyitó), *c* (csukó), *v* (vessző), *p* (pontosvessző)

A fenti példa az *aanaavaacpaanaavaacp* szónak felel meg.

További helyes szavak: *ε, aancp, aanaacp, aancpaancp* stb.

3. Írjon fel környezetfüggetlen nyelvtant egy egyszerű blokkstrukturált nyelv szintaxisának leírására:

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

Terminális szimbólumok: *b* (begin), *e* (end), *s* (skip)

A fenti példa a *bsbeebbsessse* szónak felel meg.

További helyes szavak: *ε, s, ss, be, sss, bse, sbe, bes, ssss, bebe* stb.

4. Az alábbi példaprogramhoz készítsen egyszerű reguláris kifejezéseket, melyek felismerik a változókat, számkonstansokat, *ha, paros, paratlan, akkor* kulcsszavakat, értékadás operátort, whitespaceket. Majd ezen egységek, mint terminális szimbólumok segítségével írjunk fel egy környezetfüggetlen nyelvtant a szintaxisra!

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

- a. Bővítsük a szintaktikus szabályokat olyan módon, hogy értékadás jobb oldalán változónév is állhasson.
- b. Bővítsük ki a nyelvet az összeadás és kivonás operátorokkal (új terminális elemek), majd ezekből (illetve a már meglévő változó, konstans, *paros*, *paratlan* segítségével) adjunk minél általánosabb nyelvtani szabályokat a kifejezések leírására.
- c. Bővítsük úgy a nyelvet, hogy mostantól több utasítás is tartozhasson a *ha* utasítás *akkor* részébe. Gondoljuk végig, hogy ez milyen további kiegészítéseket eredményez.
- d. Bővítsük a nyelvtanunkat egy előltesztelő ciklus konstrukció szabályaival is!

5. Írjon fel környezetfüggetlen nyelvtant a nulladrendű logika szintaxisának leírására:

$$(a \leftrightarrow b) \rightarrow (\neg a \vee b) \wedge \uparrow$$

Terminális szimbólumok: *a* (azonosító: változók nevei), *i* (igaz: \uparrow), *h* (hamis: \downarrow), *n* (nyitó), *c* (csukó), *g* (negáció: \neg), *k* (konjunkció: \wedge), *d* (diszjunkció: \vee), *m* (implikáció: \rightarrow), *e* (ekvivalencia: \leftrightarrow)

A fenti példa a *naeacmngadacki* szónak felel meg.

További helyes szavak: *a*, *i*, *h*, *ga*, *gi*, *gh*, *nac*, *nic*, *nhc*, *ada*, *aki*, *hmi*, *ima*, *gimi* stb.

Adja meg a fenti példához tartozó szintaxisfát! Vizsgálja meg, hogy adható-e több különböző szintaxisfa ehhez a szóhoz!

Lexikális elemzés

Az itt feltüntetett feladatokat a gyakorlatvezető által adott *lexikális elemző* példaszoftver segítségével oldja meg!

1. Próbálja ki a példaszoftvert, futtassa le néhány tesztfájltra, elemezze az eredményt!
2. Helyezze át az azonosítók felismerésére szolgáló reguláris kifejezést az összes többi reguláris kifejezés elé! Hogyan változik meg az elemző működése? Miért?
Vonja vissza a módosítást!
3. Adja hozzá a *.** reguláris kifejezést az elemzőhöz valamilyen üzenet kiírásával együtt! Hogyan változik meg az elemző működése? Miért?
Vonja vissza a módosítást!
4. Változtassa meg az azonosítók definícióját úgy, hogy csak nagybetűvel kezdődhessenek, több nagybetű ne szerepelhessen, és számokat ne követhessen betű! Készítsen tesztfájlt a megoldás ellenőrzéséhez!
Mentse el más néven, és térjen vissza az eredeti változathoz!
5. Tegye alkalmassá a lexikális elemzőt a *?* és *:* operátorok felismerésére!
6. Adjon hozzá az elemzőhöz egy reguláris kifejezést, amely pontosan két tizedesjegyet és tetszőleges (de legalább egy) egész jegyet tartalmazó törtszámok felismerésére alkalmas!

Fontos-e, hogy ez a reguláris kifejezés hol helyezkedik el az egész számliterálok reguláris kifejezéséhez képest? Miért?

7. Egészítsük ki a lexikális elemzőt olyan módon, hogy különböző számrendszerekben megadott számkonstansokat is fel tudjon ismerni.
 - Bináris literálokat, melyek "0b"-vel kezdődnek és ezt követően 0 és 1 számjegyeket tartalmaznak.
 - Oktális literálokat, melyek "0o"-val kezdődnek, majd 0-7 közötti számjegyejeit tartalmaznak.
 - Hexadecimális literálokat, melyek "0x"-el kezdődnek majd számjegyekkel illetve a - f közötti betűkkel folytatódhatnak (a betűk lehetnek kis- és nagybetű is).

Például 0xAF vagy 0o7 helyes literálok, de 0o9 vagy 0xx helytelenek.
8. Módosítsa az azonosító felismerésért felelős reguláris kifejezést olyan módon, hogy a továbbiakban a helyes, felismert azonosítók így nézzenek ki: _ karakterrel kezdődnek, melyet páratlan számú számjegy, majd páros számú (legalább kettő) kisbetű követ.
9. Egészítsük ki a nyelvet/lexikális elemzőt dátum típus támogatással. Ehhez az alábbiakat vegyük fel:
 - "date" kulcsszó ami a típus nevét jelöli
 - Dátum literál YYYY-MM-DD formában, ahol az YYYY rész minimális értéke 1900, maximális nincs, az MM a 01-12 intervallumból, míg a DD a 01-31 intervallumból vehet fel értéket (függetlenül attól, hogy melyik hónapban vagyunk)
10. Adjon hozzá C stílusú többsoros megjegyzések felismerésére képes reguláris kifejezést az elemzőhöz!

Szintaktikus elemzés

Szemantikus elemzés

Assembly programozás

Kódgenerálás