

# Beadandó feladat specifikációja

## Veremautomata implementálása lépésről lépésre

1. A feladatot grafikus felületen kell elkészíteni (Nem kötelező C# nyelven).
2. A programban a vezetéshez szükséges táblázat és az input szalag tartalma cserélhető és a program működése közben módosítható.
3. A program megoldása önálló munkát igényel, a kész munka átesik plágium vizsgálaton.
4. A program működését dokumentálni kell vagy szöveges formában, vagy a program működés közbeni bemutatásához videó felvétellel.

A feladatot implementálni kell és bemutatni a programot működés közben. A program az alábbi lépések alapján készíthető el:

1. Szükség van két változóra,
  - a) amit inputszalagként kezelünk (string típusú),
  - b) amit indexelésre használunk (int típusú).
2. A táblázatot tárolni kell egy DataGrid, vagy hasonló típusú WinForm-elemre. A tárolás történhet grafikusan, vagy egy  $n \times m$ -es string típusú mátrixban.
3. Szükség van egy Stack típusú változóra.  
Az elemzés során ebbe a verembe terminális és nemterminális jelek is fognak kerülni. A verem kezdetben a szabályrendszer startszimbólumát tartalmazza (E).
4. Az inputszalag kezdetben a teljes elemzendő kifejezést tartalmazza, például: `String input = "(i+i) * i#"`. Az input végére egy `#` jelet kell elhelyezni.
5. Ciklus: Be kell olvasni a soron következő elemet (minden lépésben az 'index' változó alapján az aktuális karaktert) az inputszalagról.
6. Ki kell venni a verem legfelső elemét (pop-művelet).
7. Az így kapott adatokat (inputról érkező jel és a veremből kivett terminális/nemterminális jel) sor- és oszlopindexként kell használni a szabályokat tartalmazó mátrix indexeléséhez.
8. A mátrixban megtalált cellából vegyük ki az ott található elemet (az a mátrixelem, amit a két helyről kivett elemek alapján azonosítunk).  
Például ha az 'input' aktuális eleme a '+', a veremben pedig az E' nemterminális elem található, akkor az azonosított cella a (+TE', 2), ahol a vessző bal oldalán egy szabály visszatérési értéke (veremre írandó elemek, kvázi az új állapot), a vessző jobb oldalán pedig a szabály sorszáma található.
9. Az azonosított cellában négyféle elem fordulhat elő.
  - I. Ha a cella üres, az azt jelenti, hogy a kifejezésben hibát találtunk.  
(a ciklusból kilépve, az 'index' aktuális értéke megadja a legelőször megtalált hiba helyét)
  - II. Ha a cella az 'elfogad' szót tartalmazza (tehát a mi esetünkben az inputról és a

veremről is '#' karakter érkezik), akkor az elemzés végére értünk, a kifejezés helyesnek bizonyult.

III. Ha a 'pop' szó található a cellában, akkor el kell távolítani a verem tetején található elemet (egy karaktert, ami lehet terminális vagy nemterminális jel), és ekkor az indexet is léptetni kell (megnövelni a változó értékét eggyel). Kizárólag 'pop'-szabály esetén léptetünk az inputon.

IV. Ha a cella egy zárójeles szabályt tartalmaz, akkor:

- a) el kell távolítani a zárójeleket.
- b) a szöveget szét kell bontani vessző szeparátor mentén verem
- c) a szöveg első fele (például ) karakterenként kerüljön a veremre (itt probléma lehet az E' kezelése, mivel egy jelnek számít, de két karakter!)
- d) a szöveg második fele ( ) egy listában (például: ) eltárolásra kerül. Ez a kiíratáshoz, a levezetés követéséhez, valamint a szintaxisfa elkészítéséhez fog kelleni, az elemzés szempontjából nincs vezérlő szerepe.

10. Az 5.-9. lépéseket addig kell ismételni, amíg az input végére nem érünk, vagy közben hibát nem találunk (üres cellára futunk).

Amennyiben a működési ciklus végén a verem kiürült, és az inputszalag végére értünk, a kifejezés helyes.

Ezt úgy is megállapíthatjuk, ha a két változó által azonosított cella a szabályokat tartalmazó mátrixban az 'elfogad' szót tartalmazza.

Hogy az elemzési folyamatot követni (monitorozni) tudjuk, a programnak folyamatosan mutatnia kell a lépéseket egy rendezett hármas (tuple, ordered triplet, ez az automata konfigurációja is lehetne egyben) adatszerkezetben a következő formában:

( $i+1$ \*, $i$ #,E,emptylist) kezdetben

Ezt követően minden lépésben:

( $+i$ \*, $i$ #,+TE',14862)

ahol

- az első elem az aktuális inputszalag maradék része (optimális esetben ezek az elemek pop-művelet esetén elfogynak, míg a végén csak '#' karakter marad),
- a középső elem a verem aktuális tartalma
- a jobboldali elem pedig az eddig alkalmazott szabályok bővülő sorozata.

Az alábbi kódrészletek olyan egyszerű kifejezések elemzése közben mutatják a verem automata állapotát, vagyis a konfigurációt, amely az elemzés lépéseit tartalmazza.

Az első listában a korábban is említett ( $i+1$ )\* $i$  kifejezés levezetése látható, de ebben az esetben zárójeleztük az összeadást tartalmazó részkifejezést.

$(TE',1) ((i+i)*i\#, TE'\#, 1)$   
 $(FT',4) ((i+i)*i\#, FT'E'\#, 14)$   
 $((E),7) ((i+i)*i\#, (E)T'E'\#, 147)$   
 $(TE',1) (i+i)*i\#, TE')T'E'\#, 1471)$   
 $(FT',4) (i+i)*i\#, FT'E')T'E'\#, 14714)$   
 $(i,8) (i+i)*i\#, iT'E')T'E'\#, 147148)$   
 $(\varepsilon,6) (+i)*i\#, E')T'E'\#, 1471486)$   
 $(+TE',2) (+i)*i\#, +TE')T'E'\#, 14714862)$   
 $(FT',4) (i)*i\#, FT'E')T'E'\#, 147148624)$   
 $(i,8) (i)*i\#, iT'E')T'E'\#, 1471486248)$   
 $(\varepsilon,7) ()*i\#, E')T'E'\#, 14714862487)$   
 $(\varepsilon,3) ()*i\#, )T'E'\#, 147148624873)$   
 $(*FT',5) (*i\#, *FT'E'\#, 1471486248735)$   
 $(i,8) (i\#, iT'E'\#, 14714862487358)$   
 $(\varepsilon,6) (\#, E'\#, 147148624873586)$   
 $(\varepsilon,3) (\#, \#, 1471486248735863)$   
 elfogadva

3. A második levezetés a jóval egyszerűbb  $i+i$  kifejezés elemzésének a lépéseit mutatja be.

$(TE',1) (i+i\#, TE'\#, 1)$   
 $(FT',4) (i+i\#, FT'E'\#, 14)$   
 $(i,8) (i+i\#, iT'E'\#, 148)$   
 $(\varepsilon,6) (+i\#, E'\#, 1486)$   
 $(+TE',2) (+i\#, +TE'\#, 14862)$   
 $(FT',4) (i\#, FT'E'\#, 148624)$

(i,8) (i#, iT'E'#, 1486248)

(ε,6) (#, E'#, 14862486)

(ε,3) (#, #, 148624863)

elfogadva

4. A harmadik lista egy kicsivel bonyolultabb az előző kettőnél, ez az  $i+i^*(i+i)$  kifejezés levezetését mutatja.

(TE',1) (i+i\*(i+i)#, TE'#, 1)

(FT',4) (i+i\*(i+i)#, FT'E'#, 14)

(i,8) (i+i\*(i+i)#, iT'E'#, 148)

(ε,6) (+i\*(i+i)#, E'#, 1486)

(+TE',2) (+i\*(i+i)#, +TE'#, 14862)

(FT',4) (i\*(i+i)#, FT'E'#, 148624)

(i,8) (i\*(i+i)#, iT'E'#, 1486248)

(\*FT',5) (\*(i+i)#, \*FT'E'#, 14862485)

((E),7) ((i+i)#, (E)T'E'#, 148624857)

(TE',1) (i+i)#, TE')T'E'#, 1486248571)

(FT',4) (i+i)#, FT'E')T'E'#, 14862485714)

(i,8) (i+i)#, iT'E')T'E'#, 148624857148)

(ε,6) (+i)#, E')T'E'#, 1486248571486)

(+TE',2) (+i)#, +TE')T'E'#, 14862485714862)

(FT',4) (i)#, FT'E')T'E'#, 148624857148624)

(i,8) (i)#, iT'E')T'E'#, 1486248571486248)

(ε,7) ()#, E')T'E'#, 14862485714862487)

(ε,3) ()#, )T'E'#, 148624857148624873)

(ε,6) (#, E'#, 1486248571486248736)

(ε,3) (#, #, 14862485714862487363)

elfogadva

5. Érdemes a levezetések elkészíteni önállóan papíron, vagy egy szöveg fájl használatával. Amennyiben ez sikerül és azonos eredményre jutunk, mint a fentiek, akkor biztosan megértettük az elemző működését.

A program, miután befejezte a futását, kirajzolja a szitaxisfát a képernyőre. Ez szöveges formában, vagy grafikusan is történhet.

Az implementációhoz az alábbi videó is a segítségünkre lehet:

<https://www.youtube.com/watch?v=WtDa52u1tR4>

	+	*	(	)	i	#
<i>E</i>			$(TE', 1)$		$(TE', 1)$	
<i>E'</i>	$(+TE', 2)$			$(\epsilon, 3)$		$(\epsilon, 3)$
<i>T</i>			$(FT', 4)$		$(FT', 4)$	
<i>T'</i>	$(\epsilon, 6)$	$(*FT', 5)$		$(\epsilon, 6)$		$(\epsilon, 6)$
<i>F</i>			$((E), 7)$		$(i, 8)$	
+	<i>pop</i>					
*		<i>pop</i>				
(			<i>pop</i>			
)				<i>pop</i>		
i					<i>pop</i>	
#						<i>elfogad</i>

$(i + i * i\#, S\#, \varepsilon)$	$\xrightarrow{(TE',1)}$	(	$i + i * i\#$ ,	$TE'\#$ ,	1	)
	$\xrightarrow{(FT',4)}$	(	$i + i * i\#$ ,	$FT'E'\#$ ,	14	)
	$\xrightarrow{(i,8)}$	(	$i + i * i\#$ ,	$iT'E'\#$ ,	148	)
	$\xrightarrow{pop}$	(	$+i * i\#$ ,	$T'E'\#$ ,	148	)
	$\xrightarrow{(\varepsilon,6)}$	(	$+i * i\#$ ,	$E'\#$ ,	1486	)
	$\xrightarrow{(+TE',2)}$	(	$+i * i\#$ ,	$+TE'\#$ ,	14862	)
	$\xrightarrow{pop}$	(	$i * i\#$ ,	$TE'\#$ ,	14862	)
	$\xrightarrow{(FT',4)}$	(	$i * i\#$ ,	$FT'E'\#$ ,	148624	)
	$\xrightarrow{(i,8)}$	(	$i * i\#$ ,	$iT'E'\#$ ,	1486248	)
	$\xrightarrow{pop}$	(	$*i\#$ ,	$T'E'\#$ ,	1486248	)
	$\xrightarrow{(*FT',5)}$	(	$*i\#$ ,	$*FT'E'\#$ ,	14862485	)
	$\xrightarrow{pop}$	(	$i\#$ ,	$FT'E'\#$ ,	14862485	)
	$\xrightarrow{(i,8)}$	(	$i\#$ ,	$iT'E'\#$ ,	148624858	)
	$\xrightarrow{pop}$	(	$\#$ ,	$T'E'\#$ ,	148624858	)
	$\xrightarrow{(\varepsilon,6)}$	(	$\#$ ,	$E'\#$ ,	1486248586	)
	$\xrightarrow{(\varepsilon,3)}$	(	$\#$ ,	$\#$ ,	14862485863	)
	$\xrightarrow{elfogad}$		O.K.			

Jó munkát!