

Forrás: <http://users.itk.ppke.hu/~csuba/files/digszam/gyakzh2018.pdf>

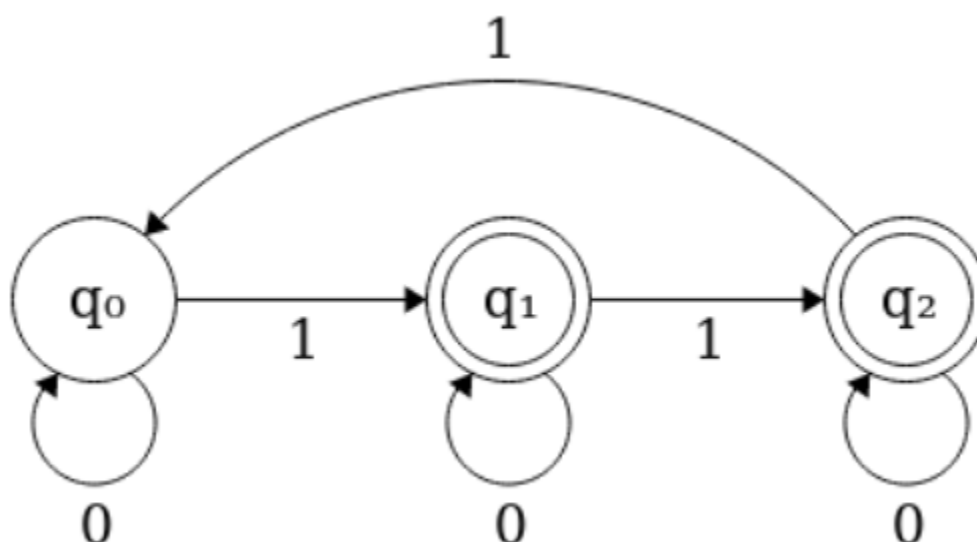
1. L nyelv azokból az $\{0, 1\}$ feletti szavakból áll, amelyekben az 1-ek száma nem osztható hárommal. Adjon meg egy L-t reprezentáló reguláris kifejezést és DVA-t!

Regex: $0^*10^*(1 \cup \epsilon)0^*(10^*10^*10^*)^*$

A string kezdődhet bármennyi 0-val, egy 1-es biztos van benne (a nulla osztható hárommal), utána is jöhetnek 0-k, ezután jöhet még egy egyes (ha végül az 1-esek száma hárommal osztva kettő maradékot ad; ha egyet, akkor lesz epsilon). Ezután a string hátralevő részében hárommal osztható számú 1-es van, köztük lehet bármennyi 0.

DVA:

A kezdőállapot q_0 , lemaradt a nyíl.

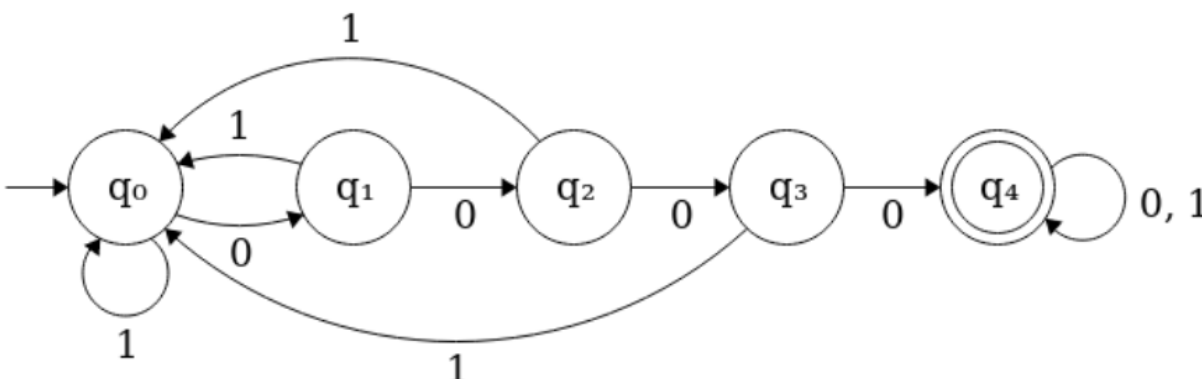


2. Adjon meg egy olyan generatív grammatikát, amely az $L = \{w \in \{0, 1\}^* : w \text{ páratlan hosszú}\}$ nyelvet generálja!

$S \rightarrow 00S \mid 01S \mid 10S \mid 11S \mid 0 \mid 1$

A rekurzív esetekben mindig kettővel nő a string hossza, bármilyen kombinációban lehetséges, a nemrekurzív esetekben viszont eggyel, tehát összességébe páratlan lesz a string hossza.

3. Konstruáljon meg egy olyan determinisztikus véges automatát, mely azokat a $\{0, 1\}$ feletti szavakat fogadja el, amelyekben van négy egymás utáni '0'!



4. Mondja ki a reguláris nyelvcsaládba nem tartozást tanúsító pumpálási lemmát! Mi a környezetfüggetlen nyelvcsaládba nem tartozást tanúsító pumpálási lemma? (Nem kell bizonyítani!) Bizonyítsa be, hogy az $L = \{0^n 1^m 0^n \mid n \geq 0\}$ nyelv nem reguláris!

A tételekmondások megtalálhatók a 2. és 3. diáorban, <http://ba1.itk.ppke.hu/>.

Bizonyítás:

Tegyük fel indirekten, hogy L reguláris. Ekkor létezik egy p pumpálási konstans, amire igaz, hogy a legalább p hosszú $w \in L$ szavak felbonthatók $w = xyz$ alakban, ahol

- $|y| \geq 0$ (szerintem itt nincs egyenlőség, sőt biztos)
- $|xy| \leq p$

és xy^nz is eleme L -nek minden $n \geq 0$ esetén.

Legyen $w = 0^p 1^m 0^p$. Ennek hossza nagyobb, mint p , tehát a fentieknek teljesülnie kell. Viszont xy -ban csak 0-k vannak, mivel $|xy| \leq p$. Ha ennek egy nemüres részét, y -t pumpáljuk, a kapott szóban az m darab 1-es bal oldalán több 0 lesz, mint a jobb oldalán, az így kapott szó nincs benne L -ben, ellentmondásra jutottunk. Ezért L nem reguláris

Figyelem: Ez természetesen csak akkor igaz, ha $m \geq 1$, ha $m = 0$, akkor $L = \{0^{2n} \mid n \geq 0\}$ nyelv természetesen reguláris.

5. Eldönthető-e algoritmikusan, hogy egy P verem automata elfogad-e egy adott w szót?

Igen. Konstruálunk egy verem automatát megvalósító (szimuláló) Turing gépet és betápláljuk a w szót. Mivel a w szó véges hosszú a Turing gép véges időn belül megáll. (M. B)

Megjegyzés: szerintem a Turing-gép nem biztos, hogy véges időn belül megáll (Cs. B)

Igen, eldönthető. Mivel minden P verem automatának megfelel egy kontext-free nyelv, lényegében azt kell eldönteni, hogy a veremautomatának megfelelő G grammatika generál-e egy adott w szót.

Tudjuk (tétel), hogy ha a G Chomsky-normálformában van, akkor egy n hosszú szó $2n-1$ lépésben generálható. Azaz, a kérdés eldöntéséhez elég az összes $2n-1$ hosszú levezetést megvizsgálni, ami véges sok lépés.

Megjegyzés: ez a 2016-os ZH 5. Feladatára is megoldás :)

(Cs. B. - lásd hivatalos tankönyv 4.6 tételének bizonyítása, ez pont ezt írja le)

6. Bizonyítsa be, hogy algoritmikusan nem eldönthető, hogy egy adott M Turinggéphez van-e olyan (haszontalan) $q \in Q$ állapot, amibe soha (semmilyen inputon indítva) nem kerül!

Másik megoldás: soroljuk fel a turing gép összes állapotát és kössük össze az állapotokat ha létezik direkt átmenet az egyik állapotból a másikba. Ezután keressünk benne utat a kezdő állapotból q -ba. (vagy hamilton utat ha azt akarjuk bizonyítani hogy nem létezik olyan q' amire sose lépnünk és a Hamilton út az NP teljes probléma tehát algoritmikusan nem megoldható) (M. B)

A problémát visszavezetjük arra, hogy eldönthető-e, hogy egy M Turing gép elfogad-e egy w szót. A bizonyítás lényege, hogy létrehozunk egy olyan M' gépet, ami az elfogadón kívül nem tartalmaz useless állapotokat, az elfogadóba pedig akkor kerül, ha M elfogadja a w szót. Ez lényegében azt jelenti, hogy M' -nek pontosan akkor van useless állapota, ha M nem fogadja el w -t (hiszen ekkor az elfogadó állapot useless). Ha el tudnánk dönteni, hogy van-e useless state, azzal lényegében eldöntenénk, hogy M elfogadja-e a w szót, amiről tudjuk, hogy nem lehetséges.

A bizonyítás lépései:

Tegyük fel, hogy eldönthető, hogy egy adott Turing-gépnek van-e haszontalan állapota. Konstruáljunk egy M' Turing gépet, ami a következőt teszi egy x bemenetre:

- Ha $x \neq w$, akkor elutasítja,
- ha $x=w$ akkor lefuttatja az M gépet, és annak eredménye szerint dönt

Módosítsuk a gépet, hogy mikor elutasító állapotba kerülne, előtte lépegessen végig az összes állapotán, az elfogadót kivéve (ez megtehető egy új szalag-szimbólum bevezetésével, és ezek közti jobbra-balra lépegetéssel, részletekért lásd a linkelt megoldás). Legyen ez M'' .

A feltételezésnek megfelelően eldöntjük, hogy M'' -nek van-e haszontalan állapota, ha van, akkor M elutasítja, különben pedig elfogadja a w -t. Tekintve, hogy ez egy eldönthetetlen probléma, ellentmondáshoz jutottunk.

(Cs. B - <https://cseweb.ucsd.edu/classes/sp08/cse105/hw4s.pdf>)

7. Adja meg a P és az NP nyelvosztályok definícióját! Mikor mondjuk azt, hogy $L1$ polinomilisan visszavezethető $L2$ -re? Mikor mondjuk azt, hogy L NP -teljes? A $HCYCLE_e$ probléma azt hivatott eldönteni, hogy van-e egy gráfban olyan Hamilton kör, amely tartalmazza a gráf e élét. Bizonyítsa be, hogy $HCYCLE_e$ NP -teljes feltételezvé, hogy tudjuk, hogy $HCYCLE$ NP -teljes. Bizonyítsa be, azt is hogy $HCYCLE$ NP -teljes feltételezvé, hogy tudjuk, hogy $HCYCLE_e$ NP -teljes.

Adja meg a P és az NP nyelvosztályok definícióját!

9. és 10. Előadás diák (<http://ba1.itk.ppke.hu/>)

$P = \bigcup_{k \geq 1} TIME(n^k)$ a polinom időben egyszalagos TG-pel eldönthet nyelvek osztálya.

$NP = \bigcup_{k \geq 1} NTIME(n^k)$ a polinom időben egyszalagos NDTG-pel eldönthet nyelvek osztálya.

Mikor mondjuk azt, hogy $L1$ polinomilisan visszavezethető $L2$ -re?

$L1$ és $L2$ Σ^* feletti nyelvek. $L1$ polinom időben visszavezethető $L2$ -re, ha van $f: \Sigma^* \rightarrow \Sigma^*$ polinomiális időben kiszámítható fv, hogy $w \in L1 \Leftrightarrow f(w) \in L2$. Ezt $L1 \leq_P L2$ jelöli

Mikor mondjuk azt, hogy L NP -teljes?

Egy L NP -beli nyelv NP -teljes, ha abból a feltételezésből, hogy L P -beli is, következik, hogy az összes NP -beli nyelv P -beli, azaz $NP=P$.

Bizonyítsa be, hogy $HCYCLE_e$ NP-teljes feltételezvé, hogy tudjuk, hogy $HCYCLE$ NP-teljes.

1. $HCYCLE_e$ benne van NP-ben. Merlin-Arthur példával (tudományosan: polinomiális tanú): Merlin megadja a $HCYCLE_e$ megoldását, élek sorozataként amit Arthur ellenőrizni tud a megoldás éleinek és a gráf csúcsainak felsorolásával.
2. Oldjuk meg $HCYCLE_e$ segítségével $HCYCLE$ -t. Adjunk hozzá két új pontot amit kössünk össze egymással és a gráf egy tetszőleges pontjával. (ezzel háromszöget alkotva). Ezután kérdezzük meg a $HCYCLE_e$ algoritmust hogy van-e az új gráfban Hamilton kör ami tartalmazza az újonnan alkotott háromszög bármely tetszőleges élét. És ha van, akkor az eredeti gráfban volt Hamilton kör. (Azért kell az új pontokat felvenni és azok egyik élét megadni, mert ha csak véletlenszerűen kiválasztunk egy élt a gráfból és nincs olyan Hamilton kör ami azt a kiválasztott élt tartalmazza attól még lehetséges, hogy van Hamilton kör a gráfban csak épp más éleket használ. Az újonnan felvett éleket nem lehet más élekkel helyettesíteni a Hamilton körben ezért alkalmasak a feladat megoldására) (M. B)

Szerintem egyszerűbben megoldja a problémát, ha lefuttatjuk $HCYCLE_e$ -t az összes élre a gráfban. Mivel polinom számú (egész pontosan $n(n-1)/2$ -nél) kevesebb él van, ez még mindig polinom idejű. (Cs. B)

Bizonyítsa be, azt is hogy $HCYCLE$ NP-teljes feltételezvé, hogy tudjuk, hogy $HCYCLE_e$ NP-teljes.

1. $HCYCLE$ benne van NP-ben. Merlin-Arthur megint: Merlin megadja a Hamilton kör csúcsait, Arthur végig megy a csúcsokon és ellenőrzi, hogy valóban megvannak köztük az élek és hogy egyetlen csúcs sem maradt ki. (ehhez csak a csúcsok és a gráf éleinek felsorolására van szükség tehát polinomiális idejű => polinomiális tanú)
2. Oldjuk meg $HCYCLE$ segítségével $HCYCLE_e$ -et bármely g gráfra. Hozzunk létre egy új g' gráfot amiben kitöröljük az e csúcsot és felveszünk két új pontot amit összekötünk egymással és az e él korábbi csúcsaival (tehát ha eddig e él A és B csúcsokat kötötte össze akkor most felveszük C és D csúcsokat és összekötjük A -t C -vel, C -t D -vel és D -t B -vel) Ezt az új gráfot beadjuk a $HCYCLE$ -t megoldó fekete doboznak (algoritmusnak, függvénynek, etc.) és ha ez elfogadható állapotban fejeződik be akkor az eredeti g gráfban is van Hamilton kör ami átmegy az e élen. (M. B)

Az elejével egyetértek (Arthur-Merlin). Második részt nem értem teljesen, szóval itt egy másik verzió rá, aztán válassza ki mindenki amelyik neki tetszik :)

Legyen az e él által összekötött két csúcs A és B . Kitörölöm az e élet, létrehozok egy C csúcsot, illetve összekötöm A -t C -vel és C -t B -vel. Lefuttatom a $HCYCLE$ algoritmust. Mivel C

mindenképp benne kell legyen a Hamilton-körben, ebben a módosított gráfban pontosan akkor lesz Hamilton-kör, ha az eredetiben van Hamilton kör az e élen keresztül. Ezzel megoldottam a HCYCLe problémát a HCYCLe segítségével. (Cs. B)

<http://users.itk.ppke.hu/~csuba/files/digszam/ZH2016.jpg>

ZH 2016 (csak amiben ettől különbözik)

5. Feladat: Eldönthető-e algoritmikusan, hogy egy CF grammatika generál-e egy w szót.

Ez lényegét tekintve ugyanaz, mint amit fent megoldottunk, hiszen a veremautomata egy az egyben megfeleltethető egy CF grammatikának.

6. Feladat: Bizonyítsa be, hogy algoritmikusan nem eldönthető, hogy egy M turing gép egy S inputon végzett számítás során leírja-e az ABC^x elemét a szalagra

Módosítsuk a Turing-gépet a következő módon:

- Vezessünk be egy új (y) szimbólumot az ABC -be
- Mindenhol, ahol a gép x -et használ cseréljük ki y -ra (így a gép sehol nem ír x -et)
- A gép elfogadó állapotába lépéskor írjunk egy x -et a szalagra.

Ekkor annak eldöntése, hogy a számítás során kerül-e x a szalagra ekvivalens annak eldöntésével, hogy a gép elfogadja-e az s szót - ezt pedig tudjuk, hogy nem eldönthető.

7. Feladat: Bizonyítsa, hogy NP-teljes annak eldöntése, hogy egy gráf kiszínezhető-e 17 színnel.

Ez volt az utolsó gyakorlaton, szóval csak röviden: elfogadtuk tételként (bizonyítás nélkül), hogy a 3 színnel való színezhetőség NP-teljes, így ha ezt meg tudjuk oldani a 17 szín segítségével, akkor kész vagyunk.²

1. A probléma benne van az NP-ben: Arthur-Merlin mese, Merlin megad egy színezést, Arthur ezt az élek száma lépésben ellenőrzi (polinom időben).

2. El akarom dönteni, hogy egy gráf színezhető-e 3 színnel. Felveszek 14 új csúcsot, összekötöm őket egymással (K_{14}), majd minden új csúcsot összekötök minden eredeti csúccsal. Mivel a K_{14} színezéséhez 14 szín kell, és a kapcsolatok miatt ezek egyike sem használható a régi csúcsokhoz, az új gráf pontosan akkor színezhető 14 színnel, ha az eredeti 3 színnel színezhető volt. Ha a 17-szín el tudom dönteni P-ben, akkor - a fent leírt módon - a 3 színt is el tudom. Ellentmondás, készen vagyunk :D