

Algoritmuselmélet 15. előadás

Katona Gyula Y.
Budapesti Műszaki és Gazdaságtudományi Egyetem
Számítástudományi Tsz.
I. B. 137/b
kiskat@cs.bme.hu

2002 Április 23.

NP-beli nyelvek

A tanú tétel segítségével könnyű belátni, hogy egy nyelv NP-beli.

NP-beli nyelvek

A tanú tétel segítségével könnyű belátni, hogy egy nyelv NP-beli.

Csak azt kell belátni, hogy **létezik** tanú, megkeresni nem kell tudni.

NP-beli nyelvek

A tanú tétel segítségével könnyű belátni, hogy egy nyelv NP-beli.

Csak azt kell belátni, hogy **létezik** tanú, megkeresni nem kell tudni.

A mi szerepünk a bíró szerepe, nem a nyomozóé.

NP-beli nyelvek

A tanú tétel segítségével könnyű belátni, hogy egy nyelv NP-beli.

Csak azt kell belátni, hogy **létezik** tanú, megkeresni nem kell tudni.

A mi szerepünk a bíró szerepe, nem a nyomozóé.

3 színnel színezhető gráfok

$G \rightarrow$ pl. adjacencia mátrix sorai egymásután fűzve

NP-beli nyelvek

A tanú tétel segítségével könnyű belátni, hogy egy nyelv NP-beli.

Csak azt kell belátni, hogy **létezik** tanú, megkeresni nem kell tudni.

A mi szerepünk a bíró szerepe, nem a nyomozóé.

3 színnel színezhető gráfok

$G \rightarrow$ pl. adjacencia mátrix sorai egymásután fűzve

$x \in 3\text{-SZÍN} \implies$ ha az x szónak megfelelő gráf 3 színnel színezhető.

NP-beli nyelvek

A tanú tétel segítségével könnyű belátni, hogy egy nyelv NP-beli.

Csak azt kell belátni, hogy **létezik** tanú, megkeresni nem kell tudni.

A mi szerepünk a bíró szerepe, nem a nyomozóé.

3 színnel színezhető gráfok

$G \rightarrow$ pl. adjacencia mátrix sorai egymásután fűzve

$x \in 3\text{-SZÍN} \implies$ ha az x szónak megfelelő gráf 3 színnel színezhető.

Tétel. $3\text{-SZÍN} \in \text{NP}$.

Bizonyítás: Alkalmas tanú G egy jó színezése

NP-beli nyelvek

A tanú tétel segítségével könnyű belátni, hogy egy nyelv NP-beli.

Csak azt kell belátni, hogy **létezik** tanú, megkeresni nem kell tudni.

A mi szerepünk a bíró szerepe, nem a nyomozóé.

3 színnel színezhető gráfok

$G \rightarrow$ pl. adjacencia mátrix sorai egymásután fűzve

$x \in 3\text{-SZÍN} \implies$ ha az x szónak megfelelő gráf 3 színnel színezhető.

Tétel. $3\text{-SZÍN} \in \text{NP}$.

Bizonyítás: Alkalmas tanú G egy jó színezése

Ez leírható $2n$ bittel \implies (pl. legyen 01=**piros**, 10=**sárga**, 11=**zöld**)

NP-beli nyelvek

A tanú tétel segítségével könnyű belátni, hogy egy nyelv NP-beli.

Csak azt kell belátni, hogy **létezik** tanú, megkeresni nem kell tudni.

A mi szerepünk a bíró szerepe, nem a nyomozóé.

3 színnel színezhető gráfok

$G \rightarrow$ pl. adjacencia mátrix sorai egymásután fűzve

$x \in 3\text{-SZÍN} \implies$ ha az x szónak megfelelő gráf 3 színnel színezhető.

Tétel. $3\text{-SZÍN} \in \text{NP}$.

Bizonyítás: Alkalmas tanú G egy jó színezése

Ez leírható $2n$ bittel \implies (pl. legyen 01=**piros**, 10=**sárga**, 11=**zöld**)

G , színezés \rightarrow bíró \implies jó színezés-e

NP-beli nyelvek

A tanú tétel segítségével könnyű belátni, hogy egy nyelv NP-beli.

Csak azt kell belátni, hogy **létezik** tanú, megkeresni nem kell tudni.

A mi szerepünk a bíró szerepe, nem a nyomozóé.

3 színnel színezhető gráfok

$G \rightarrow$ pl. adjacencia mátrix sorai egymásután fűzve

$x \in 3\text{-SZÍN} \implies$ ha az x szónak megfelelő gráf 3 színnel színezhető.

Tétel. $3\text{-SZÍN} \in \text{NP}$.

Bizonyítás: Alkalmas tanú G egy jó színezése

Ez leírható $2n$ bittel \implies (pl. legyen 01=**piros**, 10=**sárga**, 11=**zöld**)

G , színezés \rightarrow bíró \implies jó színezés-e

Ez polinom időben megtehető TG-vel.

NP-beli nyelvek

A tanú tétel segítségével könnyű belátni, hogy egy nyelv NP-beli.

Csak azt kell belátni, hogy **létezik** tanú, megkeresni nem kell tudni.

A mi szerepünk a bíró szerepe, nem a nyomozóé.

3 színnel színezhető gráfok

$G \rightarrow$ pl. adjacencia mátrix sorai egymásután fűzve

$x \in 3\text{-SZÍN} \implies$ ha az x szónak megfelelő gráf 3 színnel színezhető.

Tétel. $3\text{-SZÍN} \in \text{NP}$.

Bizonyítás: Alkalmas tanú G egy jó színezése

Ez leírható $2n$ bittel \implies (pl. legyen 01=**piros**, 10=**sárga**, 11=**zöld**)

G , színezés \rightarrow bíró \implies jó színezés-e

Ez polinom időben megtehető TG-vel.

Ha G nem 3-színezhető, akkor nem lehet tanúja.

Hamilton-körrel rendelkező gráfok

$H \implies$ Azon gráfok szavai, amik tartalmaznak Hamilton-kört.

Hamilton-körrel rendelkező gráfok

$H \implies$ Azon gráfok szavai, amik tartalmaznak Hamilton-kört.

Tétel. $H \in \text{NP}$.

Bizonyítás: A $G \in H$ állításnak rövid tanúja egy Hamilton-kör.

Hamilton-körrel rendelkező gráfok

$H \implies$ Azon gráfok szavai, amik tartalmaznak Hamilton-kört.

Tétel. $H \in \text{NP}$.

Bizonyítás: A $G \in H$ állításnak rövid tanúja egy Hamilton-kör.
csúcsok sorrendje $\implies O(n \log n)$ bit

Hamilton-körrel rendelkező gráfok

$H \implies$ Azon gráfok szavai, amik tartalmaznak Hamilton-kört.

Tétel. $H \in \text{NP}$.

Bizonyítás: A $G \in H$ állításnak rövid tanúja egy Hamilton-kör.
csúcsok sorrendje $\implies O(n \log n)$ bit
a bíró ellenőrzi, hogy van-e él a következő csúcsba G -ben.

Hamilton-körrel rendelkező gráfok

$H \implies$ Azon gráfok szavai, amik tartalmaznak Hamilton-kört.

Tétel. $H \in \text{NP}$.

Bizonyítás: A $G \in H$ állításnak rövid tanúja egy Hamilton-kör.
csúcsok sorrendje $\implies O(n \log n)$ bit
a bíró ellenőrzi, hogy van-e él a következő csúcsba G -ben.

Hasonlóan Hamilton-útra, irányított Hamilton-körre, -útra

Legyen NH a Hamilton-kört **nem** tartalmazó gráfok nyelve.

Tétel. $NH \in \text{coNP}$.

Síkba rajzolható gráfok

Legyen L a síkba rajzolható gráfok nyelve

Síkba rajzolható gráfok

Legyen L a síkba rajzolható gráfok nyelve

Tétel. $L \in \text{NP}$

Bizonyítás: G tanúja egy síkbarajzolása.

Síkba rajzolható gráfok

Legyen L a síkba rajzolható gráfok nyelve

Tétel. $L \in \text{NP}$

Bizonyítás: G tanúja egy síkbarajzolása.

Fáry-Wágner \implies van olyan is, ami egyenes szakaszokat használ. Sőt olyan is van, hogy a koordináták nem túl nagyok.

Síkba rajzolható gráfok

Legyen L a síkba rajzolható gráfok nyelve

Tétel. $L \in \text{NP}$

Bizonyítás: G tanúja egy síkbarajzolása.

Fáry-Wágner \implies van olyan is, ami egyenes szakaszokat használ. Sőt olyan is van, hogy a koordináták nem túl nagyok.

\implies Tanú a csúcsok koordinátái.

Síkba rajzolható gráfok

Legyen L a síkba rajzolható gráfok nyelve

Tétel. $L \in \text{NP}$

Bizonyítás: G tanúja egy síkbarajzolása.

Fáry-Wágner \implies van olyan is, ami egyenes szakaszokat használ. Sőt olyan is van, hogy a koordináták nem túl nagyok.

\implies Tanú a csúcsok koordinátái.

A bíró ellenőrzi, hogy az élek nem metszik egymást.

Síkba rajzolható gráfok

Legyen L a síkba rajzolható gráfok nyelve

Tétel. $L \in \text{NP}$

Bizonyítás: G tanúja egy síkbarajzolása.

Fáry-Wágner \implies van olyan is, ami egyenes szakaszokat használ. Sőt olyan is van, hogy a koordináták nem túl nagyok.

\implies Tanú a csúcsok koordinátái.

A bíró ellenőrzi, hogy az élek nem metszik egymást.

Tétel. $L \in \text{coNP}$ ($\implies L \in \text{NP} \cap \text{coNP}$: *jól karakterizált*)

Bizonyítás: Van tanú a $G \notin L$ állításra is.

Síkba rajzolható gráfok

Legyen L a síkba rajzolható gráfok nyelve

Tétel. $L \in \text{NP}$

Bizonyítás: G tanúja egy síkbarajzolása.

Fáry-Wágner \implies van olyan is, ami egyenes szakaszokat használ. Sőt olyan is van, hogy a koordináták nem túl nagyok.

\implies Tanú a csúcsok koordinátái.

A bíró ellenőrzi, hogy az élek nem metszik egymást.

Tétel. $L \in \text{coNP}$ ($\implies L \in \text{NP} \cap \text{coNP}$: *jól karakterizált*)

Bizonyítás: Van tanú a $G \notin L$ állításra is.

Vagy nem gráf vagy **Kuratowski** \implies van benne vagy K_5 -tel vagy $K_{3,3}$ -mal topologikusan izomorf részgráf.

Síkba rajzolható gráfok

Legyen L a síkba rajzolható gráfok nyelve

Tétel. $L \in \text{NP}$

Bizonyítás: G tanúja egy síkbarajzolása.

Fáry-Wágner \implies van olyan is, ami egyenes szakaszokat használ. Sőt olyan is van, hogy a koordináták nem túl nagyok.

\implies Tanú a csúcsok koordinátái.

A bíró ellenőrzi, hogy az élek nem metszik egymást.

Tétel. $L \in \text{coNP}$ ($\implies L \in \text{NP} \cap \text{coNP}$: *jól karakterizált*)

Bizonyítás: Van tanú a $G \notin L$ állításra is.

Vagy nem gráf vagy **Kuratowski** \implies van benne vagy K_5 -tel vagy $K_{3,3}$ -mal topologikusan izomorf részgráf.

Tanú egy ilyen leírása, ezt a bíró könnyen ellenőrizheti.

Síkba rajzolható gráfok

Legyen L a síkba rajzolható gráfok nyelve

Tétel. $L \in \text{NP}$

Bizonyítás: G tanúja egy síkbarajzolása.

Fáry-Wágner \implies van olyan is, ami egyenes szakaszokat használ. Sőt olyan is van, hogy a koordináták nem túl nagyok.

\implies Tanú a csúcsok koordinátái.

A bíró ellenőrzi, hogy az élek nem metszik egymást.

Tétel. $L \in \text{coNP}$ ($\implies L \in \text{NP} \cap \text{coNP}$: *jól karakterizált*)

Bizonyítás: Van tanú a $G \notin L$ állításra is.

Vagy nem gráf vagy **Kuratowski** \implies van benne vagy K_5 -tel vagy $K_{3,3}$ -mal topologikusan izomorf részgráf.

Tanú egy ilyen leírása, ezt a bíró könnyen ellenőrizheti.

Tétel. $L \in \text{P}$

Síkba rajzolható gráfok

Legyen L a síkba rajzolható gráfok nyelve

Tétel. $L \in \text{NP}$

Bizonyítás: G tanúja egy síkbarajzolása.

Fáry-Wágner \implies van olyan is, ami egyenes szakaszokat használ. Sőt olyan is van, hogy a koordináták nem túl nagyok.

\implies Tanú a csúcsok koordinátái.

A bíró ellenőrzi, hogy az élek nem metszik egymást.

Tétel. $L \in \text{coNP}$ ($\implies L \in \text{NP} \cap \text{coNP}$: *jól karakterizált*)

Bizonyítás: Van tanú a $G \notin L$ állításra is.

Vagy nem gráf vagy **Kuratowski** \implies van benne vagy K_5 -tel vagy $K_{3,3}$ -mal topologikusan izomorf részgráf.

Tanú egy ilyen leírása, ezt a bíró könnyen ellenőrizheti.

Tétel. $L \in \text{P}$

Sejtés: $H \notin \text{coNP}$ és 3-SZÍN $\notin \text{coNP}$

A prímszámok nyelve

Jelölje Π a (binárisan ábrázolt) prímszámok nyelvét.

A prímszámok nyelve

Jelölje Π a (binárisan ábrázolt) prímszámok nyelvét.

Tétel. [V. R. Pratt, 1975]

$$\Pi \in \text{NP} \cap \text{coNP}.$$

A prímszámok nyelve

Jelölje Π a (binárisan ábrázolt) prímszámok nyelvét.

Tétel. [V. R. Pratt, 1975]

$$\Pi \in \text{NP} \cap \text{coNP}.$$

Bizonyítás: $\Pi \in \text{coNP}$: Ha egy szám nem prímszám, akkor tanú egy osztója, pl. 6 nem prímszám, mert $2|6$.

A prímszámok nyelve

Jelölje Π a (binárisan ábrázolt) prímszámok nyelvét.

Tétel. [V. R. Pratt, 1975]

$$\Pi \in \text{NP} \cap \text{coNP}.$$

Bizonyítás: $\Pi \in \text{coNP}$: Ha egy szám nem prím, arra tanú egy osztója, pl. 6 nem prím, mert $2|6$.

$\Pi \in \text{NP}$:

Lemma. Legyen $p \geq 2$ egy egész szám. A p pontosan akkor prímszám, ha van olyan $1 \leq g < p$ egész, melyre teljesülnek az alábbiak:

1. $g^{p-1} \equiv 1 \pmod{p}$,
2. $g^{\frac{p-1}{r}} \not\equiv 1 \pmod{p}$ minden r prímszámra, melyre $r|p-1$.

Gyors hatványozás: a^m alakú hatvány legfeljebb $2 \log_2 m$ szorzással kiszámítható.

Gyors hatványozás: a^m alakú hatvány legfeljebb $2 \log_2 m$ szorzással kiszámítható.

$$m = e_0 + e_1 2^1 + e_2 2^2 + \dots + e_k 2^k, \quad k \leq \log_2 m \text{ és } e_j \in \{0, 1\}.$$

Gyors hatványozás: a^m alakú hatvány legfeljebb $2 \log_2 m$ szorzással kiszámítható.

$$m = e_0 + e_1 2^1 + e_2 2^2 + \dots + e_k 2^k, \quad k \leq \log_2 m \text{ és } e_j \in \{0, 1\}.$$

Ismételt négyzetre emelésekkel $\implies a^{2^j} \implies$ ez k szorzás

Gyors hatványozás: a^m alakú hatvány legfeljebb $2 \log_2 m$ szorzással kiszámítható.

$$m = e_0 + e_1 2^1 + e_2 2^2 + \dots + e_k 2^k, \quad k \leq \log_2 m \text{ és } e_j \in \{0, 1\}.$$

Ismételt négyzetre emelésekkel $\implies a^{2^j} \implies$ ez k szorzás

szorozzuk össze az a^{2^j} hatványokat azokra a j értékekre, melyekre $e_j = 1$

$$\implies a^m = a^{e_0 + e_1 2^1 + e_2 2^2 + \dots + e_k 2^k} = a^{e_0} a^{e_1 2^1} a^{e_2 2^2} \dots a^{e_k 2^k}$$

$\implies k$ szorzás

Gyors hatványozás: a^m alakú hatvány legfeljebb $2 \log_2 m$ szorzással kiszámítható.

$$m = e_0 + e_1 2^1 + e_2 2^2 + \dots + e_k 2^k, \quad k \leq \log_2 m \text{ és } e_j \in \{0, 1\}.$$

Ismételt négyzetre emelésekkel $\implies a^{2^j} \implies$ ez k szorzás

szorozzuk össze az a^{2^j} hatványokat azokra a j értékekre, melyekre $e_j = 1$

$$\implies a^m = a^{e_0 + e_1 2^1 + e_2 2^2 + \dots + e_k 2^k} = a^{e_0} a^{e_1 2^1} a^{e_2 2^2} \dots a^{e_k 2^k}$$

$\implies k$ szorzás

a^m mérete $m + a$ méretében exponenciális \implies exponenciális alg.

Gyors hatványozás: a^m alakú hatvány legfeljebb $2 \log_2 m$ szorzással kiszámítható.

$$m = e_0 + e_1 2^1 + e_2 2^2 + \dots + e_k 2^k, \quad k \leq \log_2 m \text{ és } e_j \in \{0, 1\}.$$

Ismételt négyzetre emelésekkel $\implies a^{2^j} \implies$ ez k szorzás

szorozzuk össze az a^{2^j} hatványokat azokra a j értékekre, melyekre $e_j = 1$

$$\implies a^m = a^{e_0 + e_1 2^1 + e_2 2^2 + \dots + e_k 2^k} = a^{e_0} a^{e_1 2^1} a^{e_2 2^2} \dots a^{e_k 2^k}$$

$\implies k$ szorzás

a^m mérete $m + a$ méretében exponenciális \implies exponenciális alg.

$a^m \pmod n$ mérete legfeljebb $\log_2 n \implies$ ha mindig a maradékot vesszük

\implies polinomiális alg.

Gyors hatványozás: a^m alakú hatvány legfeljebb $2 \log_2 m$ szorzással kiszámítható.

$$m = e_0 + e_1 2^1 + e_2 2^2 + \dots + e_k 2^k, \quad k \leq \log_2 m \text{ és } e_j \in \{0, 1\}.$$

Ismételt négyzetre emelésekkel $\implies a^{2^j} \implies$ ez k szorzás

szorozzuk össze az a^{2^j} hatványokat azokra a j értékekre, melyekre $e_j = 1$

$$\implies a^m = a^{e_0 + e_1 2^1 + e_2 2^2 + \dots + e_k 2^k} = a^{e_0} a^{e_1 2^1} a^{e_2 2^2} \dots a^{e_k 2^k}$$

$\implies k$ szorzás

a^m mérete $m + a$ méretében exponenciális \implies exponenciális alg.

$a^m \pmod n$ mérete legfeljebb $\log_2 n \implies$ ha mindig a maradékot vesszük

\implies polinomiális alg.

A p prím állításra tanú: g és $p - 1$ egész r_1, \dots, r_k prímosztói

Gyors hatványozás: a^m alakú hatvány legfeljebb $2 \log_2 m$ szorzással kiszámítható.

$$m = e_0 + e_1 2^1 + e_2 2^2 + \dots + e_k 2^k, \quad k \leq \log_2 m \text{ és } e_j \in \{0, 1\}.$$

Ismételt négyzetre emelésekkel $\implies a^{2^j} \implies$ ez k szorzás

szorozzuk össze az a^{2^j} hatványokat azokra a j értékekre, melyekre $e_j = 1$

$$\implies a^m = a^{e_0 + e_1 2^1 + e_2 2^2 + \dots + e_k 2^k} = a^{e_0} a^{e_1 2^1} a^{e_2 2^2} \dots a^{e_k 2^k}$$

$\implies k$ szorzás

a^m mérete $m + a$ méretében exponenciális \implies exponenciális alg.

$a^m \pmod n$ mérete legfeljebb $\log_2 n \implies$ ha mindig a maradékot vesszük

\implies polinomiális alg.

A p prím állításra tanú: g és $p - 1$ egész r_1, \dots, r_k prímosztói

a bíró gyors hatványozással ellenőrizheti, hogy a Lemma feltételei teljesülnek.

Gyors hatványozás: a^m alakú hatvány legfeljebb $2 \log_2 m$ szorzással kiszámítható.

$$m = e_0 + e_1 2^1 + e_2 2^2 + \dots + e_k 2^k, \quad k \leq \log_2 m \text{ és } e_j \in \{0, 1\}.$$

Ismételt négyzetre emelésekkel $\implies a^{2^j} \implies$ ez k szorzás

szorozzuk össze az a^{2^j} hatványokat azokra a j értékekre, melyekre $e_j = 1$

$$\implies a^m = a^{e_0 + e_1 2^1 + e_2 2^2 + \dots + e_k 2^k} = a^{e_0} a^{e_1 2^1} a^{e_2 2^2} \dots a^{e_k 2^k}$$

$\implies k$ szorzás

a^m mérete $m + a$ méretében exponenciális \implies exponenciális alg.

$a^m \pmod{n}$ mérete legfeljebb $\log_2 n \implies$ ha mindig a maradékot vesszük

\implies polinomiális alg.

A p prím állításra tanú: g és $p - 1$ egész r_1, \dots, r_k prímosztói

a bíró gyors hatványozással ellenőrizheti, hogy a Lemma feltételei teljesülnek.

Azt is tanúsítani kell, hogy r_1, \dots, r_k éppen a $p - 1$ prímosztói (más nincs)

\implies prímtényezős felbontás 

Gyors hatványozás: a^m alakú hatvány legfeljebb $2 \log_2 m$ szorzással kiszámítható.

$$m = e_0 + e_1 2^1 + e_2 2^2 + \dots + e_k 2^k, \quad k \leq \log_2 m \text{ és } e_j \in \{0, 1\}.$$

Ismételt négyzetre emelésekkel $\implies a^{2^j} \implies$ ez k szorzás

szorozzuk össze az a^{2^j} hatványokat azokra a j értékekre, melyekre $e_j = 1$

$$\implies a^m = a^{e_0 + e_1 2^1 + e_2 2^2 + \dots + e_k 2^k} = a^{e_0} a^{e_1 2^1} a^{e_2 2^2} \dots a^{e_k 2^k}$$

$\implies k$ szorzás

a^m mérete $m + a$ méretében exponenciális \implies exponenciális alg.

$a^m \pmod n$ mérete legfeljebb $\log_2 n \implies$ ha mindig a maradékot vesszük

\implies polinomiális alg.

A p prím állításra tanú: g és $p - 1$ egész r_1, \dots, r_k prímosztói

a bíró gyors hatványozással ellenőrizheti, hogy a Lemma feltételei teljesülnek.

Azt is tanúsítani kell, hogy r_1, \dots, r_k éppen a $p - 1$ prímosztói (más nincs)

\implies prímtényezős felbontás 

és azt is, hogy r_1, \dots, r_k prímek

Gyors hatványozás: a^m alakú hatvány legfeljebb $2 \log_2 m$ szorzással kiszámítható.

$$m = e_0 + e_1 2^1 + e_2 2^2 + \dots + e_k 2^k, \quad k \leq \log_2 m \text{ és } e_j \in \{0, 1\}.$$

Ismételt négyzetre emelésekkel $\implies a^{2^j} \implies$ ez k szorzás

szorozzuk össze az a^{2^j} hatványokat azokra a j értékekre, melyekre $e_j = 1$

$$\implies a^m = a^{e_0 + e_1 2^1 + e_2 2^2 + \dots + e_k 2^k} = a^{e_0} a^{e_1 2^1} a^{e_2 2^2} \dots a^{e_k 2^k}$$

$\implies k$ szorzás

a^m mérete $m + a$ méretében exponenciális \implies exponenciális alg.

$a^m \pmod n$ mérete legfeljebb $\log_2 n \implies$ ha mindig a maradékot vesszük

\implies polinomiális alg.

A p prím állításra tanú: g és $p - 1$ egész r_1, \dots, r_k prímosztói

a bíró gyors hatványozással ellenőrizheti, hogy a Lemma feltételei teljesülnek.

Azt is tanúsítani kell, hogy r_1, \dots, r_k éppen a $p - 1$ prímosztói (más nincs)

\implies prímtényezős felbontás ✓

és azt is, hogy r_1, \dots, r_k prímek \implies rekurzívan ✓

Gyors hatványozás: a^m alakú hatvány legfeljebb $2 \log_2 m$ szorzással kiszámítható.

$$m = e_0 + e_1 2^1 + e_2 2^2 + \dots + e_k 2^k, \quad k \leq \log_2 m \text{ és } e_j \in \{0, 1\}.$$

Ismételt négyzetre emelésekkel $\implies a^{2^j} \implies$ ez k szorzás

szorozzuk össze az a^{2^j} hatványokat azokra a j értékekre, melyekre $e_j = 1$

$$\implies a^m = a^{e_0 + e_1 2^1 + e_2 2^2 + \dots + e_k 2^k} = a^{e_0} a^{e_1 2^1} a^{e_2 2^2} \dots a^{e_k 2^k}$$

$\implies k$ szorzás

a^m mérete $m + a$ méretében exponenciális \implies exponenciális alg.

$a^m \pmod n$ mérete legfeljebb $\log_2 n \implies$ ha mindig a maradékot vesszük

\implies polinomiális alg.

A p prím állításra tanú: g és $p - 1$ egész r_1, \dots, r_k prímosztói

a bíró gyors hatványozással ellenőrizheti, hogy a Lemma feltételei teljesülnek.

Azt is tanúsítani kell, hogy r_1, \dots, r_k éppen a $p - 1$ prímosztói (más nincs)

\implies prímtényezős felbontás ✓

és azt is, hogy r_1, \dots, r_k prímek \implies rekurzívan ✓

Belátható, hogy a tanú össz mérete $O(n^2)$.

Gyors hatványozás: a^m alakú hatvány legfeljebb $2 \log_2 m$ szorzással kiszámítható.

$$m = e_0 + e_1 2^1 + e_2 2^2 + \dots + e_k 2^k, \quad k \leq \log_2 m \text{ és } e_j \in \{0, 1\}.$$

Ismételt négyzetre emelésekkel $\implies a^{2^j} \implies$ ez k szorzás

szorozzuk össze az a^{2^j} hatványokat azokra a j értékekre, melyekre $e_j = 1$

$$\implies a^m = a^{e_0 + e_1 2^1 + e_2 2^2 + \dots + e_k 2^k} = a^{e_0} a^{e_1 2^1} a^{e_2 2^2} \dots a^{e_k 2^k}$$

$\implies k$ szorzás

a^m mérete $m + a$ méretében exponenciális \implies exponenciális alg.

$a^m \pmod n$ mérete legfeljebb $\log_2 n \implies$ ha mindig a maradékot vesszük

\implies polinomiális alg.

A p prím állításra tanú: g és $p - 1$ egész r_1, \dots, r_k prímosztói

a bíró gyors hatványozással ellenőrizheti, hogy a Lemma feltételei teljesülnek.

Azt is tanúsítani kell, hogy r_1, \dots, r_k éppen a $p - 1$ prímosztói (más nincs)

\implies prímtényezős felbontás ✓

és azt is, hogy r_1, \dots, r_k prímek \implies rekurzívan ✓

Belátható, hogy a tanú össz mérete $O(n^2)$.

Sejtés: $\Pi \notin P$

Felismerés és keresés

Prím-e \leftrightarrow legkisebb prím osztója

Felismerés és keresés

Prím-e \leftrightarrow legkisebb prím osztója

$$F = \left\{ (a, c) \mid \begin{array}{l} 1 < c \leq a \text{ egészek és van olyan } 1 < b \leq c \text{ egész,} \\ \text{melyre } b \text{ osztója } a\text{-nak} \end{array} \right\}.$$

Tétel. $F \in \text{NP} \cap \text{coNP}$.

Felismerés és keresés

Prím-e \leftrightarrow legkisebb prím osztója

$$F = \left\{ (a, c) \mid \begin{array}{l} 1 < c \leq a \text{ egészek és van olyan } 1 < b \leq c \text{ egész,} \\ \text{melyre } b \text{ osztója } a\text{-nak} \end{array} \right\}.$$

Tétel. $F \in \text{NP} \cap \text{coNP}$.

Bizonyítás: $(a, c) \in F \implies$ tanú egy jó b érték

Felismerés és keresés

Prím-e \leftrightarrow legkisebb prím osztója

$$F = \left\{ (a, c) \mid \begin{array}{l} 1 < c \leq a \text{ egészek és van olyan } 1 < b \leq c \text{ egész,} \\ \text{melyre } b \text{ osztója } a\text{-nak} \end{array} \right\}.$$

Tétel. $F \in \text{NP} \cap \text{coNP}$.

Bizonyítás: $(a, c) \in F \implies$ tanú egy jó b érték

$(a, c) \notin F \implies$ tanú \rightarrow az $a = p_1^{\alpha_1} p_2^{\alpha_2} \cdots p_k^{\alpha_k}$ és a p_i számok prím-tulajdonságának tanúi

Felismerés és keresés

Prím-e \leftrightarrow legkisebb prím osztója

$$F = \left\{ (a, c) \mid \begin{array}{l} 1 < c \leq a \text{ egészek és van olyan } 1 < b \leq c \text{ egész,} \\ \text{melyre } b \text{ osztója } a\text{-nak} \end{array} \right\}.$$

Tétel. $F \in \text{NP} \cap \text{coNP}$.

Bizonyítás: $(a, c) \in F \implies$ tanú egy jó b érték

$(a, c) \notin F \implies$ tanú \rightarrow az $a = p_1^{\alpha_1} p_2^{\alpha_2} \cdots p_k^{\alpha_k}$ és a p_i számok prím-tulajdonságának tanúi

Legjobb ismert algoritmus a prím-felbontásra: D. Shanks $\implies n$ bites inputon $c2^{n/4}$.

Tétel. *Ha $F \in P$ igaz lenne, akkor $\{\text{prímtényezős felbontás}\} \in FP$ is igaz lenne.*

Bizonyítás: Legyen E egy gyors alg. F felismerésére

Tétel. Ha $F \in P$ igaz lenne, akkor $\{\text{prímtényezős felbontás}\} \in FP$ is igaz lenne.

Bizonyítás: Legyen E egy gyors alg. F felismerésére
 $(a, a - 1) \in F$?

Tétel. Ha $F \in P$ igaz lenne, akkor $\{\text{prímtényezős felbontás}\} \in \text{FP}$ is igaz lenne.

Bizonyítás: Legyen E egy gyors alg. F felismerésére

$(a, a - 1) \in F?$

ha nem $\rightarrow a$ prím ✓

Tétel. Ha $F \in P$ igaz lenne, akkor $\{\text{prímtényezős felbontás}\} \in FP$ is igaz lenne.

Bizonyítás: Legyen E egy gyors alg. F felismerésére

$(a, a - 1) \in F?$

ha nem $\rightarrow a$ prím ✓

ha igen \implies bináris kereséssel megkeressük a legkisebb olyan c -t amire

$(a, c) \in F$

Tétel. Ha $F \in P$ igaz lenne, akkor $\{\text{prímtényezős felbontás}\} \in FP$ is igaz lenne.

Bizonyítás: Legyen E egy gyors alg. F felismerésére

$(a, a - 1) \in F?$

ha nem $\rightarrow a$ prím ✓

ha igen \implies bináris kereséssel megkeressük a legkisebb olyan c -t amire

$(a, c) \in F \implies \leq \log_2 a$ hívás

Tétel. Ha $F \in P$ igaz lenne, akkor $\{\text{prímtényezős felbontás}\} \in \text{FP}$ is igaz lenne.

Bizonyítás: Legyen E egy gyors alg. F felismerésére

$(a, a - 1) \in F?$

ha nem $\rightarrow a$ prím ✓

ha igen \implies bináris kereséssel megkeressük a legkisebb olyan c -t amire

$(a, c) \in F \implies \leq \log_2 a$ hívás

Utána a/c -re folytatjuk ...

Tétel. Ha $F \in P$ igaz lenne, akkor $\{\text{prímtényezős felbontás}\} \in \text{FP}$ is igaz lenne.

Bizonyítás: Legyen E egy gyors alg. F felismerésére

$(a, a - 1) \in F?$

ha nem $\rightarrow a$ prím ✓

ha igen \implies bináris kereséssel megkeressük a legkisebb olyan c -t amire

$(a, c) \in F \implies \leq \log_2 a$ hívás

Utána a/c -re folytatjuk ...

\implies egy prím-osztó megtalálása: $O((\log a)^d)$

Tétel. Ha $F \in P$ igaz lenne, akkor $\{\text{prímtényezős felbontás}\} \in \text{FP}$ is igaz lenne.

Bizonyítás: Legyen E egy gyors alg. F felismerésére

$(a, a - 1) \in F?$

ha nem $\rightarrow a$ prím ✓

ha igen \implies bináris kereséssel megkeressük a legkisebb olyan c -t amire

$(a, c) \in F \implies \leq \log_2 a$ hívás

Utána a/c -re folytatjuk ...

\implies egy prím-osztó megtalálása: $O((\log a)^d)$

prím-osztók száma $\leq \log a \implies$ összköltség $O((\log a)^{d+1})$

Tétel. Ha $F \in P$ igaz lenne, akkor $\{\text{prímtényezős felbontás}\} \in \text{FP}$ is igaz lenne.

Bizonyítás: Legyen E egy gyors alg. F felismerésére

$(a, a - 1) \in F?$

ha nem $\rightarrow a$ prím ✓

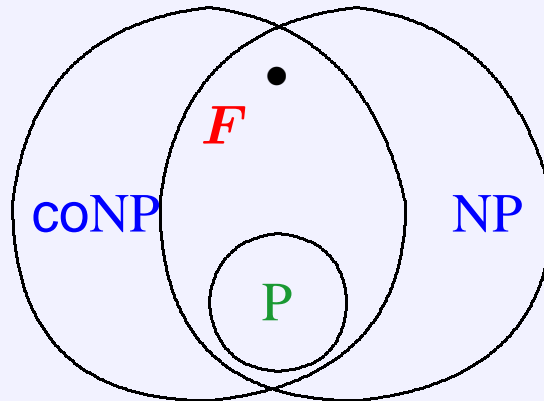
ha igen \implies bináris kereséssel megkeressük a legkisebb olyan c -t amire

$(a, c) \in F \implies \leq \log_2 a$ hívás

Utána a/c -re folytatjuk ...

\implies egy prím-osztó megtalálása: $O((\log a)^d)$

prím-osztók száma $\leq \log a \implies$ összköltség $O((\log a)^{d+1})$



Karp-redukció

Mikor nem lényegesen nehezebb egy L_1 probléma egy L_2 problémánál?

Karp-redukció

Mikor nem lényegesen nehezebb egy L_1 probléma egy L_2 problémánál?

\implies Ha L_2 felhasználásával meg lehet oldani L_1 -et is.

Karp-redukció

Mikor nem lényegesen nehezebb egy L_1 probléma egy L_2 problémánál?

⇒ Ha L_2 felhasználásával meg lehet oldani L_1 -et is.

⇒ L_1 visszavezethető a L_2 problémára.

Karp-redukció

Mikor nem lényegesen nehezebb egy L_1 probléma egy L_2 problémánál?

\implies Ha L_2 felhasználásával meg lehet oldani L_1 -et is.

\implies L_1 visszavezethető a L_2 problémára.

Definíció. Az $f : I^* \rightarrow I^*$ leképezés az $L_1 \subseteq I^*$ nyelv **Karp-redukciója** az $L_2 \subseteq I^*$ nyelvre, ha

1. Tetszőleges $x \in I^*$ szóra $x \in L_1$ pontosan akkor teljesül, ha $f(x) \in L_2$;
2. $f \in FP$, azaz f polinom időben számítható.

Karp-redukció

Mikor nem lényegesen nehezebb egy L_1 probléma egy L_2 problémánál?

\implies Ha L_2 felhasználásával meg lehet oldani L_1 -et is.

\implies L_1 visszavezethető a L_2 problémára.

Definíció. Az $f : I^* \rightarrow I^*$ leképezés az $L_1 \subseteq I^*$ nyelv **Karp-redukciója** az $L_2 \subseteq I^*$ nyelvre, ha

1. Tetszőleges $x \in I^*$ szóra $x \in L_1$ pontosan akkor teljesül, ha $f(x) \in L_2$;
2. $f \in FP$, azaz f polinom időben számítható.

Jelölés: $L_1 \prec L_2$ ha L_1 -nek van Karp-redukciója L_2 -re.

Ha tehát van algoritmusunk L_2 eldöntésére \implies

Karp-redukció

Mikor nem lényegesen nehezebb egy L_1 probléma egy L_2 problémánál?

\implies Ha L_2 felhasználásával meg lehet oldani L_1 -et is.

\implies L_1 visszavezethető a L_2 problémára.

Definíció. Az $f : I^* \rightarrow I^*$ leképezés az $L_1 \subseteq I^*$ nyelv **Karp-redukciója** az $L_2 \subseteq I^*$ nyelvre, ha

1. Tetszőleges $x \in I^*$ szóra $x \in L_1$ pontosan akkor teljesül, ha $f(x) \in L_2$;
2. $f \in FP$, azaz f polinom időben számítható.

Jelölés: $L_1 \prec L_2$ ha L_1 -nek van Karp-redukciója L_2 -re.

Ha tehát van algoritmusunk L_2 eldöntésére $\implies x \in L_1$ -re kiszámítjuk $f(x)$ -et

Karp-redukció

Mikor nem lényegesen nehezebb egy L_1 probléma egy L_2 problémánál?

\implies Ha L_2 felhasználásával meg lehet oldani L_1 -et is.

\implies L_1 visszavezethető a L_2 problémára.

Definíció. Az $f : I^* \rightarrow I^*$ leképezés az $L_1 \subseteq I^*$ nyelv **Karp-redukciója** az $L_2 \subseteq I^*$ nyelvre, ha

1. Tetszőleges $x \in I^*$ szóra $x \in L_1$ pontosan akkor teljesül, ha $f(x) \in L_2$;
2. $f \in FP$, azaz f polinom időben számítható.

Jelölés: $L_1 \prec L_2$ ha L_1 -nek van Karp-redukciója L_2 -re.

Ha tehát van algoritmusunk L_2 eldöntésére $\implies x \in L_1$ -re kiszámítjuk $f(x)$ -et eldöntjük $f(x) \in L_2$?

Karp-redukció

Mikor nem lényegesen nehezebb egy L_1 probléma egy L_2 problémánál?

\implies Ha L_2 felhasználásával meg lehet oldani L_1 -et is.

\implies L_1 visszavezethető a L_2 problémára.

Definíció. Az $f : I^* \rightarrow I^*$ leképezés az $L_1 \subseteq I^*$ nyelv **Karp-redukciója** az $L_2 \subseteq I^*$ nyelvre, ha

1. Tetszőleges $x \in I^*$ szóra $x \in L_1$ pontosan akkor teljesül, ha $f(x) \in L_2$;
2. $f \in FP$, azaz f polinom időben számítható.

Jelölés: $L_1 \preceq L_2$ ha L_1 -nek van Karp-redukciója L_2 -re.

Ha tehát van algoritmusunk L_2 eldöntésére $\implies x \in L_1$ -re kiszámítjuk $f(x)$ -et eldöntjük $f(x) \in L_2$? \implies tudjuk, hogy $x \in L_1$ igaz-e \checkmark

Karp-redukció

Mikor nem lényegesen nehezebb egy L_1 probléma egy L_2 problémánál?

\implies Ha L_2 felhasználásával meg lehet oldani L_1 -et is.

\implies L_1 visszavezethető a L_2 problémára.

Definíció. Az $f : I^* \rightarrow I^*$ leképezés az $L_1 \subseteq I^*$ nyelv **Karp-redukciója** az $L_2 \subseteq I^*$ nyelvre, ha

1. Tetszőleges $x \in I^*$ szóra $x \in L_1$ pontosan akkor teljesül, ha $f(x) \in L_2$;
2. $f \in FP$, azaz f polinom időben számítható.

Jelölés: $L_1 \prec L_2$ ha L_1 -nek van Karp-redukciója L_2 -re.

Ha tehát van algoritmusunk L_2 eldöntésére $\implies x \in L_1$ -re kiszámítjuk $f(x)$ -et eldöntjük $f(x) \in L_2$? \implies tudjuk, hogy $x \in L_1$ igaz-e \checkmark

Ha tudnánk, hogy L nehéz és tudjuk, hogy $L \prec L'$

Karp-redukció

Mikor nem lényegesen nehezebb egy L_1 probléma egy L_2 problémánál?

\implies Ha L_2 felhasználásával meg lehet oldani L_1 -et is.

\implies L_1 visszavezethető a L_2 problémára.

Definíció. Az $f : I^* \rightarrow I^*$ leképezés az $L_1 \subseteq I^*$ nyelv **Karp-redukciója** az $L_2 \subseteq I^*$ nyelvre, ha

1. Tetszőleges $x \in I^*$ szóra $x \in L_1$ pontosan akkor teljesül, ha $f(x) \in L_2$;
2. $f \in FP$, azaz f polinom időben számítható.

Jelölés: $L_1 \prec L_2$ ha L_1 -nek van Karp-redukciója L_2 -re.

Ha tehát van algoritmusunk L_2 eldöntésére $\implies x \in L_1$ -re kiszámítjuk $f(x)$ -et eldöntjük $f(x) \in L_2$? \implies tudjuk, hogy $x \in L_1$ igaz-e \checkmark

Ha tudnánk, hogy L nehéz és tudjuk, hogy $L \prec L' \implies L'$ is nehéz lenne

Karp-redukció

Mikor nem lényegesen nehezebb egy L_1 probléma egy L_2 problémánál?

\implies Ha L_2 felhasználásával meg lehet oldani L_1 -et is.

\implies L_1 visszavezethető a L_2 problémára.

Definíció. Az $f : I^* \rightarrow I^*$ leképezés az $L_1 \subseteq I^*$ nyelv **Karp-redukciója** az $L_2 \subseteq I^*$ nyelvre, ha

1. Tetszőleges $x \in I^*$ szóra $x \in L_1$ pontosan akkor teljesül, ha $f(x) \in L_2$;
2. $f \in FP$, azaz f polinom időben számítható.

Jelölés: $L_1 \prec L_2$ ha L_1 -nek van Karp-redukciója L_2 -re.

Ha tehát van algoritmusunk L_2 eldöntésére $\implies x \in L_1$ -re kiszámítjuk $f(x)$ -et eldöntjük $f(x) \in L_2$? \implies tudjuk, hogy $x \in L_1$ igaz-e \checkmark

Ha tudnánk, hogy L nehéz és tudjuk, hogy $L \prec L' \implies L'$ is nehéz lenne
Ha L' könnyű lenne, és L nem lényegesen nehezebb nála, akkor L is könnyű.

Irányított Hamilton-kör probléma

Tétel. $IH \prec H$.

Bizonyítás: $G = (V, E)$ egy irányított gráf $\rightarrow G' = (V', E')$ irányítatlan gráf

Irányított Hamilton-kör probléma

Tétel. $IH \prec H$.

Bizonyítás: $G = (V, E)$ egy irányított gráf $\rightarrow G' = (V', E')$ irányítatlan gráf
hogy G' gyorsan megépíthető és

Írányított Hamilton-kör probléma

Tétel. $IH \prec H$.

Bizonyítás: $G = (V, E)$ egy irányított gráf $\rightarrow G' = (V', E')$ irányítatlan gráf
hogy G' gyorsan megépíthető és
 G -ben \exists irányított Hamilton-kör $\leftrightarrow G'$ -ben \exists irányítatlan Hamilton-kör.

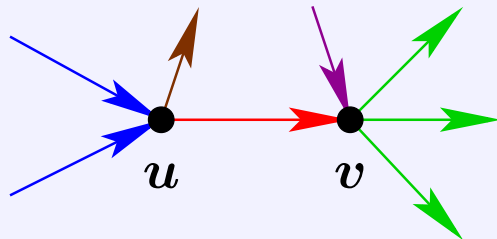
Írányított Hamilton-kör probléma

Tétel. $IH \prec H$.

Bizonyítás: $G = (V, E)$ egy irányított gráf $\rightarrow G' = (V', E')$ irányítatlan gráf
 hogy G' gyorsan megépíthető és
 G -ben \exists irányított Hamilton-kör $\leftrightarrow G'$ -ben \exists irányítatlan Hamilton-kör.

$$V' = \{v_{be}, v_*, v_{ki} \mid v \in V\},$$

$$E' = \{(v_{be}, v_*), (v_*, v_{ki}) \mid v \in V\} \cup \{(u_{ki}, v_{be}) \mid u \rightarrow v \in E\}.$$



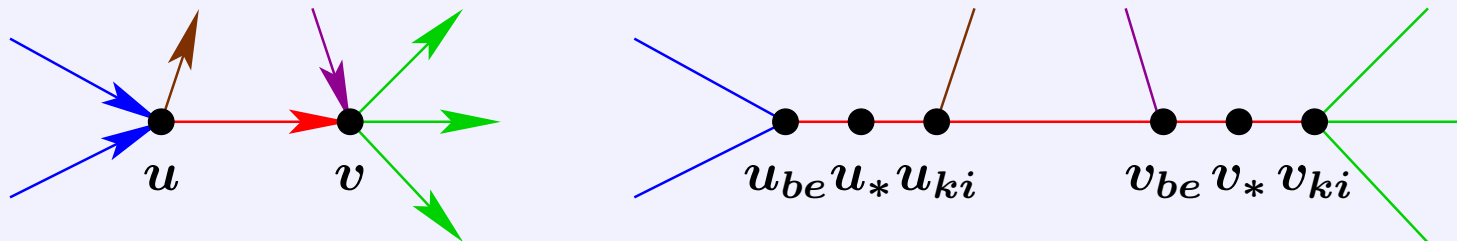
Írányított Hamilton-kör probléma

Tétel. $IH \prec H$.

Bizonyítás: $G = (V, E)$ egy irányított gráf $\rightarrow G' = (V', E')$ irányítatlan gráf
 hogy G' gyorsan megépíthető és
 G -ben \exists irányított Hamilton-kör $\leftrightarrow G'$ -ben \exists irányítatlan Hamilton-kör.

$$V' = \{v_{be}, v_*, v_{ki} \mid v \in V\},$$

$$E' = \{(v_{be}, v_*), (v_*, v_{ki}) \mid v \in V\} \cup \{(u_{ki}, v_{be}) \mid u \rightarrow v \in E\}.$$



$v(G) = n, e(G) = e \implies v(G') = 3n, e(G') = 2n + e \implies (n + e)^c$
 lépésben megkapható.

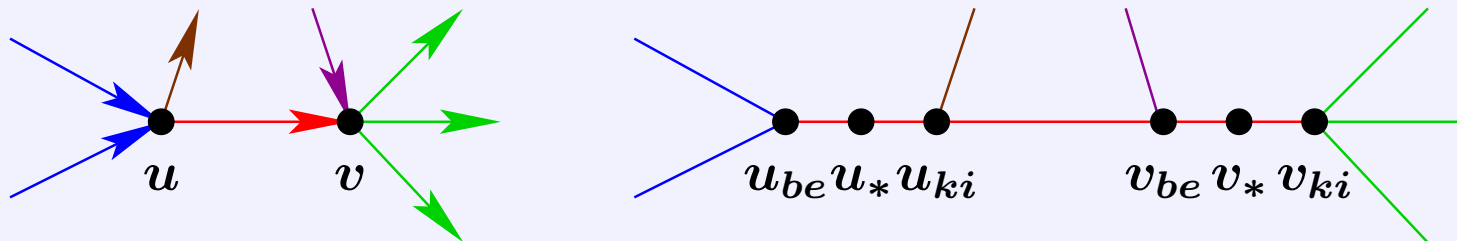
Írányított Hamilton-kör probléma

Tétel. $IH \prec H$.

Bizonyítás: $G = (V, E)$ egy irányított gráf $\rightarrow G' = (V', E')$ irányítatlan gráf
 hogy G' gyorsan megépíthető és
 G -ben \exists irányított Hamilton-kör $\leftrightarrow G'$ -ben \exists irányítatlan Hamilton-kör.

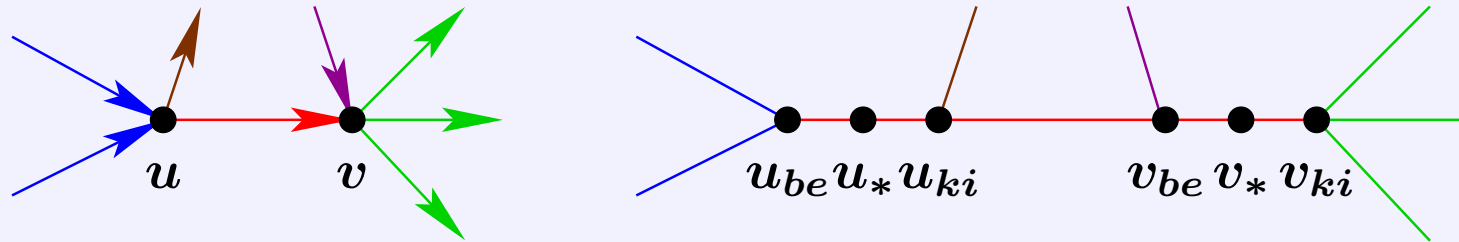
$$V' = \{v_{be}, v_*, v_{ki} \mid v \in V\},$$

$$E' = \{(v_{be}, v_*), (v_*, v_{ki}) \mid v \in V\} \cup \{(u_{ki}, v_{be}) \mid u \rightarrow v \in E\}.$$

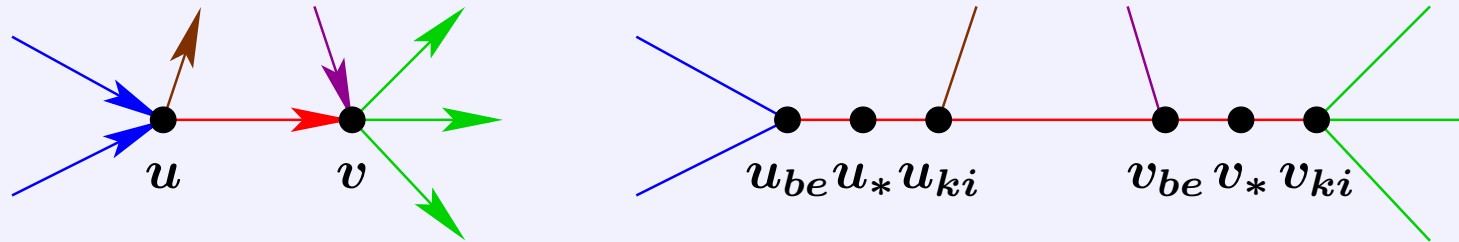


$v(G) = n, e(G) = e \implies v(G') = 3n, e(G') = 2n + e \implies (n + e)^c$
 lépésben megkapható.

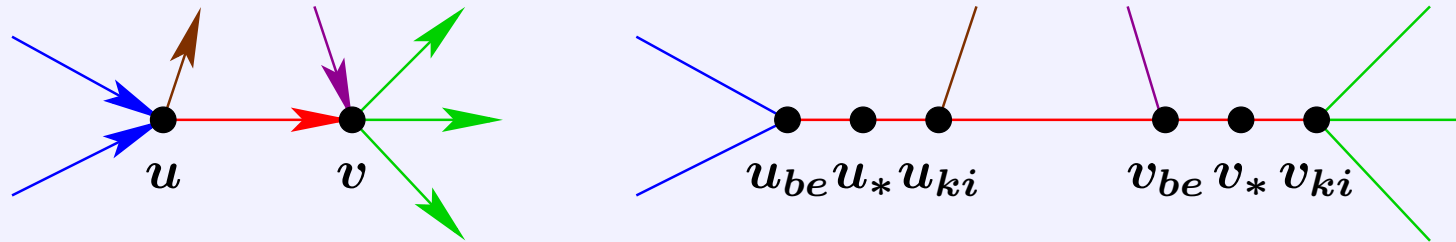
G -beli F irányított Hamilton-körének $\implies G'$ egy F' Hamilton-köre



Az F egy $u \rightarrow v$ éle \implies az F' -ben az $u_* - u_{ki} - v_{be} - v_*$ út

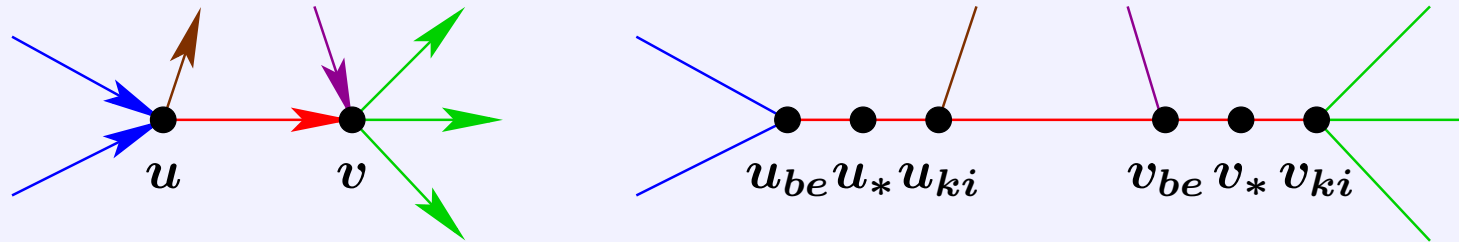


Az F egy $u \rightarrow v$ éle \implies az F' -ben az $u_* - u_{ki} - v_{be} - v_*$ út
 $\implies G \in IH \implies G' \in H$



Az F egy $u \rightarrow v$ éle \implies az F' -ben az $u_* - u_{ki} - v_{be} - v_*$ út
 $\implies G \in IH \implies G' \in H$

Ha G' -ben van egy $F' \subseteq E'$ Hamilton-kör \implies egy u_* -ból indulva egy u_{ki} felé lépünk először



Az F egy $u \rightarrow v$ éle \implies az F' -ben az $u_* - u_{ki} - v_{be} - v_*$ út
 $\implies G \in IH \implies G' \in H$

Ha G' -ben van egy $F' \subseteq E'$ Hamilton-kör \implies egy u_* -ból indulva egy u_{ki} felé lépünk először
 \implies csak $u_* - u_{ki} - v_{be} - v_*$ alakú lehet utána \implies stb. \implies Ha $G' \in H$ akkor $G \in IH$.

A Karp-redukció felhasználása

Tétel.

1. Ha $L_1 \preceq L_2$ és $L_2 \in P$, akkor $L_1 \in P$.
2. Ha $L_1 \preceq L_2$ és $L_2 \in \text{NP}$ akkor $L_1 \in \text{NP}$.
3. Ha $L_1 \preceq L_2$, akkor $\bar{L}_1 \preceq \bar{L}_2$, ahol $\bar{L}_i = I^* \setminus L_i$.
4. Ha $L_1 \preceq L_2$ és $L_2 \in \text{coNP}$, akkor $L_1 \in \text{coNP}$.
5. Ha $L_1 \preceq L_2$ és $L_2 \in \text{NP} \cap \text{coNP}$, akkor $L_1 \in \text{NP} \cap \text{coNP}$.
6. Ha $L_1 \preceq L_2$ és $L_2 \preceq L_3$, akkor $L_1 \preceq L_3$.

Bizonyítás:

Legyen $f : I^* \rightarrow I^*$ az L_1 Karp-redukciója L_2 -re, $f \in \text{FTIME}(n^k)$.

A Karp-redukció felhasználása

Tétel.

1. Ha $L_1 \prec L_2$ és $L_2 \in P$, akkor $L_1 \in P$.
2. Ha $L_1 \prec L_2$ és $L_2 \in NP$ akkor $L_1 \in NP$.
3. Ha $L_1 \prec L_2$, akkor $\bar{L}_1 \prec \bar{L}_2$, ahol $\bar{L}_i = I^* \setminus L_i$.
4. Ha $L_1 \prec L_2$ és $L_2 \in coNP$, akkor $L_1 \in coNP$.
5. Ha $L_1 \prec L_2$ és $L_2 \in NP \cap coNP$, akkor $L_1 \in NP \cap coNP$.
6. Ha $L_1 \prec L_2$ és $L_2 \prec L_3$, akkor $L_1 \prec L_3$.

Bizonyítás:

Legyen $f : I^* \rightarrow I^*$ az L_1 Karp-redukciója L_2 -re, $f \in FTIME(n^k)$.

$x \in I^*$ egy input szót, melyre szeretnénk eldönteni, hogy $x \in L_1$ teljesül-e, n az x hossza.

A Karp-redukció felhasználása

Tétel.

1. Ha $L_1 \prec L_2$ és $L_2 \in P$, akkor $L_1 \in P$.
2. Ha $L_1 \prec L_2$ és $L_2 \in NP$ akkor $L_1 \in NP$.
3. Ha $L_1 \prec L_2$, akkor $\bar{L}_1 \prec \bar{L}_2$, ahol $\bar{L}_i = I^* \setminus L_i$.
4. Ha $L_1 \prec L_2$ és $L_2 \in coNP$, akkor $L_1 \in coNP$.
5. Ha $L_1 \prec L_2$ és $L_2 \in NP \cap coNP$, akkor $L_1 \in NP \cap coNP$.
6. Ha $L_1 \prec L_2$ és $L_2 \prec L_3$, akkor $L_1 \prec L_3$.

Bizonyítás:

Legyen $f : I^* \rightarrow I^*$ az L_1 Karp-redukciója L_2 -re, $f \in FTIME(n^k)$.

$x \in I^*$ egy input szót, melyre szeretnénk eldönteni, hogy $x \in L_1$ teljesül-e, n az x hossza.

1. Kiszámítjuk $f(x)$ -et

A Karp-redukció felhasználása

Tétel.

1. Ha $L_1 \prec L_2$ és $L_2 \in P$, akkor $L_1 \in P$.
2. Ha $L_1 \prec L_2$ és $L_2 \in NP$ akkor $L_1 \in NP$.
3. Ha $L_1 \prec L_2$, akkor $\bar{L}_1 \prec \bar{L}_2$, ahol $\bar{L}_i = I^* \setminus L_i$.
4. Ha $L_1 \prec L_2$ és $L_2 \in coNP$, akkor $L_1 \in coNP$.
5. Ha $L_1 \prec L_2$ és $L_2 \in NP \cap coNP$, akkor $L_1 \in NP \cap coNP$.
6. Ha $L_1 \prec L_2$ és $L_2 \prec L_3$, akkor $L_1 \prec L_3$.

Bizonyítás:

Legyen $f : I^* \rightarrow I^*$ az L_1 Karp-redukciója L_2 -re, $f \in FTIME(n^k)$.

$x \in I^*$ egy input szót, melyre szeretnénk eldönteni, hogy $x \in L_1$ teljesül-e, n az x hossza.

1. Kiszámítjuk $f(x)$ -et \implies időigénye $\leq c_1 n^k \implies |f(x)| \leq c_1 n^k$

A Karp-redukció felhasználása

Tétel.

1. Ha $L_1 \prec L_2$ és $L_2 \in P$, akkor $L_1 \in P$.
2. Ha $L_1 \prec L_2$ és $L_2 \in NP$ akkor $L_1 \in NP$.
3. Ha $L_1 \prec L_2$, akkor $\bar{L}_1 \prec \bar{L}_2$, ahol $\bar{L}_i = I^* \setminus L_i$.
4. Ha $L_1 \prec L_2$ és $L_2 \in coNP$, akkor $L_1 \in coNP$.
5. Ha $L_1 \prec L_2$ és $L_2 \in NP \cap coNP$, akkor $L_1 \in NP \cap coNP$.
6. Ha $L_1 \prec L_2$ és $L_2 \prec L_3$, akkor $L_1 \prec L_3$.

Bizonyítás:

Legyen $f : I^* \rightarrow I^*$ az L_1 Karp-redukciója L_2 -re, $f \in FTIME(n^k)$.

$x \in I^*$ egy input szót, melyre szeretnénk eldönteni, hogy $x \in L_1$ teljesül-e, n az x hossza.

1. Kiszámítjuk $f(x)$ -et \implies időigénye $\leq c_1 n^k \implies |f(x)| \leq c_1 n^k$
 L_2 felismerő algoritmusával eldöntjük, hogy $f(x) \in L_2$ igaz-e

A Karp-redukció felhasználása

Tétel.

1. Ha $L_1 \prec L_2$ és $L_2 \in P$, akkor $L_1 \in P$.
2. Ha $L_1 \prec L_2$ és $L_2 \in NP$ akkor $L_1 \in NP$.
3. Ha $L_1 \prec L_2$, akkor $\bar{L}_1 \prec \bar{L}_2$, ahol $\bar{L}_i = I^* \setminus L_i$.
4. Ha $L_1 \prec L_2$ és $L_2 \in coNP$, akkor $L_1 \in coNP$.
5. Ha $L_1 \prec L_2$ és $L_2 \in NP \cap coNP$, akkor $L_1 \in NP \cap coNP$.
6. Ha $L_1 \prec L_2$ és $L_2 \prec L_3$, akkor $L_1 \prec L_3$.

Bizonyítás:

Legyen $f : I^* \rightarrow I^*$ az L_1 Karp-redukciója L_2 -re, $f \in FTIME(n^k)$.

$x \in I^*$ egy input szót, melyre szeretnénk eldönteni, hogy $x \in L_1$ teljesül-e, n az x hossza.

1. Kiszámítjuk $f(x)$ -et \implies időigénye $\leq c_1 n^k \implies |f(x)| \leq c_1 n^k$
 L_2 felismerő algoritmusával eldöntjük, hogy $f(x) \in L_2$ igaz-e
 \implies időigénye $\leq c_2 (c_1 n^k)^l$

A Karp-redukció felhasználása

Tétel.

1. Ha $L_1 \prec L_2$ és $L_2 \in P$, akkor $L_1 \in P$.
2. Ha $L_1 \prec L_2$ és $L_2 \in NP$ akkor $L_1 \in NP$.
3. Ha $L_1 \prec L_2$, akkor $\bar{L}_1 \prec \bar{L}_2$, ahol $\bar{L}_i = I^* \setminus L_i$.
4. Ha $L_1 \prec L_2$ és $L_2 \in coNP$, akkor $L_1 \in coNP$.
5. Ha $L_1 \prec L_2$ és $L_2 \in NP \cap coNP$, akkor $L_1 \in NP \cap coNP$.
6. Ha $L_1 \prec L_2$ és $L_2 \prec L_3$, akkor $L_1 \prec L_3$.

Bizonyítás:

Legyen $f : I^* \rightarrow I^*$ az L_1 Karp-redukciója L_2 -re, $f \in FTIME(n^k)$.

$x \in I^*$ egy input szót, melyre szeretnénk eldönteni, hogy $x \in L_1$ teljesül-e, n az x hossza.

1. Kiszámítjuk $f(x)$ -et \implies időigénye $\leq c_1 n^k \implies |f(x)| \leq c_1 n^k$
 L_2 felismerő algoritmusával eldöntjük, hogy $f(x) \in L_2$ igaz-e
 \implies időigénye $\leq c_2 (c_1 n^k)^l$
 $x \in L_1 \leftrightarrow f(x) \in L_2 \implies$ összidő $O(n^{kl})$ ✓

A Karp-redukció felhasználása

Tétel.

1. Ha $L_1 \prec L_2$ és $L_2 \in P$, akkor $L_1 \in P$.
2. Ha $L_1 \prec L_2$ és $L_2 \in NP$ akkor $L_1 \in NP$.
3. Ha $L_1 \prec L_2$, akkor $\bar{L}_1 \prec \bar{L}_2$, ahol $\bar{L}_i = I^* \setminus L_i$.
4. Ha $L_1 \prec L_2$ és $L_2 \in coNP$, akkor $L_1 \in coNP$.
5. Ha $L_1 \prec L_2$ és $L_2 \in NP \cap coNP$, akkor $L_1 \in NP \cap coNP$.
6. Ha $L_1 \prec L_2$ és $L_2 \prec L_3$, akkor $L_1 \prec L_3$.

Bizonyítás:

Legyen $f : I^* \rightarrow I^*$ az L_1 Karp-redukciója L_2 -re, $f \in FTIME(n^k)$.

$x \in I^*$ egy input szót, melyre szeretnénk eldönteni, hogy $x \in L_1$ teljesül-e, n az x hossza.

1. Kiszámítjuk $f(x)$ -et \implies időigénye $\leq c_1 n^k \implies |f(x)| \leq c_1 n^k$
 L_2 felismerő algoritmusával eldöntjük, hogy $f(x) \in L_2$ igaz-e
 \implies időigénye $\leq c_2 (c_1 n^k)^l$
 $x \in L_1 \leftrightarrow f(x) \in L_2 \implies$ összidő $O(n^{kl})$ ✓

2.: Ha $L_1 \prec L_2$ és $L_2 \in \text{NP}$ akkor $L_1 \in \text{NP}$: Az $f(x) \in L_2$ tény egy y tanúja jó $x \in L_1$ tanújának is, és az L_2 -höz tartozó bíró egy kis módosítással jó lesz az L_1 bírójának is

2.: Ha $L_1 \prec L_2$ és $L_2 \in \text{NP}$ akkor $L_1 \in \text{NP}$: Az $f(x) \in L_2$ tény egy y tanúja jó $x \in L_1$ tanújának is, és az L_2 -höz tartozó bíró egy kis módosítással jó lesz az L_1 bírójának is

$$|y| \leq |f(x)|^c \implies |y| \leq c_1^c |x|^{kc}$$

2.: Ha $L_1 \prec L_2$ és $L_2 \in \text{NP}$ akkor $L_1 \in \text{NP}$: Az $f(x) \in L_2$ tény egy y tanúja jó $x \in L_1$ tanújának is, és az L_2 -höz tartozó bíró egy kis módosítással jó lesz az L_1 bírójának is

$$|y| \leq |f(x)|^c \implies |y| \leq c_1^c |x|^{kc}$$

Az L_1 bírója az $(x, y) \rightarrow f(x) \implies (f(x), y) \rightarrow L_2$ bírójának

2.: Ha $L_1 \prec L_2$ és $L_2 \in \text{NP}$ akkor $L_1 \in \text{NP}$: Az $f(x) \in L_2$ tény egy y tanúja jó $x \in L_1$ tanújának is, és az L_2 -höz tartozó bíró egy **kis módosítással** jó lesz az L_1 bírójának is

$$|y| \leq |f(x)|^c \implies |y| \leq c_1^c |x|^{kc}$$

Az L_1 bírója az $(x, y) \rightarrow f(x) \implies (f(x), y) \rightarrow L_2$ bírójának

L_1 bírója pontosan akkor fogadja el az (x, y) párt $\leftrightarrow L_2$ bírója elfogadja az $(f(x), y)$ párt

2.: Ha $L_1 \prec L_2$ és $L_2 \in \text{NP}$ akkor $L_1 \in \text{NP}$: Az $f(x) \in L_2$ tény egy y tanúja jó $x \in L_1$ tanújának is, és az L_2 -höz tartozó bíró egy kis módosítással jó lesz az L_1 bírójának is

$$|y| \leq |f(x)|^c \implies |y| \leq c_1^c |x|^{kc}$$

Az L_1 bírója az $(x, y) \rightarrow f(x) \implies (f(x), y) \rightarrow L_2$ bírójának

L_1 bírója pontosan akkor fogadja el az (x, y) párt $\leftrightarrow L_2$ bírója elfogadja az $(f(x), y)$ párt

3.: Ha $L_1 \prec L_2$, akkor $\bar{L}_1 \prec \bar{L}_2$: Mint 1. hiszen $x \in I^*$ szóra $x \notin L_1 \leftrightarrow f(x) \notin L_2$.

2.: Ha $L_1 \prec L_2$ és $L_2 \in \text{NP}$ akkor $L_1 \in \text{NP}$: Az $f(x) \in L_2$ tény egy y tanúja jó $x \in L_1$ tanújának is, és az L_2 -höz tartozó bíró egy kis módosítással jó lesz az L_1 bírójának is

$$|y| \leq |f(x)|^c \implies |y| \leq c_1^c |x|^{kc}$$

Az L_1 bírója az $(x, y) \rightarrow f(x) \implies (f(x), y) \rightarrow L_2$ bírójának

L_1 bírója pontosan akkor fogadja el az (x, y) párt $\leftrightarrow L_2$ bírója elfogadja az $(f(x), y)$ párt

3.: Ha $L_1 \prec L_2$, akkor $\bar{L}_1 \prec \bar{L}_2$: Mint 1. hiszen $x \in I^*$ szóra $x \notin L_1 \leftrightarrow f(x) \notin L_2$.

4.: Ha $L_1 \prec L_2$ és $L_2 \in \text{coNP}$, akkor $L_1 \in \text{coNP}$: $\iff 2., 3.$

2.: Ha $L_1 \prec L_2$ és $L_2 \in \text{NP}$ akkor $L_1 \in \text{NP}$: Az $f(x) \in L_2$ tény egy y tanúja jó $x \in L_1$ tanújának is, és az L_2 -höz tartozó bíró egy kis módosítással jó lesz az L_1 bírójának is

$$|y| \leq |f(x)|^c \implies |y| \leq c_1^c |x|^{kc}$$

Az L_1 bírója az $(x, y) \rightarrow f(x) \implies (f(x), y) \rightarrow L_2$ bírójának

L_1 bírója pontosan akkor fogadja el az (x, y) párt $\leftrightarrow L_2$ bírója elfogadja az $(f(x), y)$ párt

3.: Ha $L_1 \prec L_2$, akkor $\bar{L}_1 \prec \bar{L}_2$: Mint 1. hiszen $x \in I^*$ szóra $x \notin L_1 \leftrightarrow f(x) \notin L_2$.

4.: Ha $L_1 \prec L_2$ és $L_2 \in \text{coNP}$, akkor $L_1 \in \text{coNP}$: $\iff 2., 3.$

5.: Ha $L_1 \prec L_2$ és $L_2 \in \text{NP} \cap \text{coNP}$, akkor $L_1 \in \text{NP} \cap \text{coNP}$.: $\iff 2., 4.$

2.: Ha $L_1 \prec L_2$ és $L_2 \in \text{NP}$ akkor $L_1 \in \text{NP}$: Az $f(x) \in L_2$ tény egy y tanúja jó $x \in L_1$ tanújának is, és az L_2 -höz tartozó bíró egy kis módosítással jó lesz az L_1 bírójának is

$$|y| \leq |f(x)|^c \implies |y| \leq c_1^c |x|^{kc}$$

Az L_1 bírója az $(x, y) \rightarrow f(x) \implies (f(x), y) \rightarrow L_2$ bírójának

L_1 bírója pontosan akkor fogadja el az (x, y) párt $\leftrightarrow L_2$ bírója elfogadja az $(f(x), y)$ párt

3.: Ha $L_1 \prec L_2$, akkor $\bar{L}_1 \prec \bar{L}_2$: Mint 1. hiszen $x \in I^*$ szóra $x \notin L_1 \leftrightarrow f(x) \notin L_2$.

4.: Ha $L_1 \prec L_2$ és $L_2 \in \text{coNP}$, akkor $L_1 \in \text{coNP}$: $\iff 2., 3.$

5.: Ha $L_1 \prec L_2$ és $L_2 \in \text{NP} \cap \text{coNP}$, akkor $L_1 \in \text{NP} \cap \text{coNP}$: $\iff 2., 4.$

6.: Ha $L_1 \prec L_2$ és $L_2 \prec L_3$, akkor $L_1 \prec L_3$: Legyen f az $L_1 \prec L_2$ függvénye, ami $O(x^k)$ időben számolható és g az $L_2 \prec L_3$ függvénye, ami $O(x^l)$ időben számolható

2.: Ha $L_1 \prec L_2$ és $L_2 \in \text{NP}$ akkor $L_1 \in \text{NP}$: Az $f(x) \in L_2$ tény egy y tanúja jó $x \in L_1$ tanújának is, és az L_2 -höz tartozó bíró egy **kis módosítással** jó lesz az L_1 bírójának is

$$|y| \leq |f(x)|^c \implies |y| \leq c_1^c |x|^{kc}$$

Az L_1 bírója az $(x, y) \rightarrow f(x) \implies (f(x), y) \rightarrow L_2$ bírójának

L_1 bírója pontosan akkor fogadja el az (x, y) párt $\leftrightarrow L_2$ bírója elfogadja az $(f(x), y)$ párt

3.: Ha $L_1 \prec L_2$, akkor $\bar{L}_1 \prec \bar{L}_2$: Mint 1. hiszen $x \in I^*$ szóra $x \notin L_1 \leftrightarrow f(x) \notin L_2$.

4.: Ha $L_1 \prec L_2$ és $L_2 \in \text{coNP}$, akkor $L_1 \in \text{coNP}$: $\iff 2., 3.$

5.: Ha $L_1 \prec L_2$ és $L_2 \in \text{NP} \cap \text{coNP}$, akkor $L_1 \in \text{NP} \cap \text{coNP}$.: $\iff 2., 4.$

6.: Ha $L_1 \prec L_2$ és $L_2 \prec L_3$, akkor $L_1 \prec L_3$: Legyen f az $L_1 \prec L_2$ függvénye, ami $O(x^k)$ időben számolható

és g az $L_2 \prec L_3$ függvénye, ami $O(x^l)$ időben számolható

Az $L_1 \prec L_3$ függvénye $g(f(x))$ lesz, ami $O((x^k)^l) = O(x^{kl})$ időben számolható

NP-teljes nyelvek

Definíció. Az $L \subseteq I^*$ nyelv **NP-teljes**, ha

1. $L \in \text{NP}$,
2. tetszőleges (azaz minden) $L' \in \text{NP}$ nyelv esetén létezik $L' \preceq L$ Karp-redukció.

Egy NP-teljes nyelv tehát legalább olyan nehéz, mint bármely más NP-beli nyelv.

NP-teljes nyelvek

Definíció. Az $L \subseteq I^*$ nyelv **NP-teljes**, ha

1. $L \in \text{NP}$,
2. *tetszőleges (azaz minden) $L' \in \text{NP}$ nyelv esetén létezik $L' \prec L$ Karp-redukció.*

Egy NP-teljes nyelv tehát legalább olyan nehéz, mint bármely más NP-beli nyelv.

Ha egy ilyen nyelvről kiderülne, hogy P-beli (coNP-beli), akkor ugyanez igaz lenne minden NP-beli nyelvre.

Van-e NP-teljes nyelv?

Cook–Levin-tétel

Boole-formula:

$$\text{pl. } (x_1 \vee \overline{x_2} \vee x_5) \wedge (\overline{x_3} \vee x_2 \vee x_6 \vee x_1) \wedge \overline{(x_5 \vee x_6)}$$

Cook–Levin-tétel

Boole-formula:

$$\text{pl. } (x_1 \vee \overline{x_2} \vee x_5) \wedge (\overline{x_3} \vee x_2 \vee x_6 \vee x_1) \wedge \overline{(x_5 \vee x_6)}$$

Definíció. *SAT nyelv:* a kielégíthető Boole-formulák nyelve.

Cook–Levin-tétel

Boole-formula:

$$\text{pl. } (x_1 \vee \overline{x_2} \vee x_5) \wedge (\overline{x_3} \vee x_2 \vee x_6 \vee x_1) \wedge \overline{(x_5 \vee x_6)}$$

Definíció. *SAT nyelv:* a kielégíthető Boole-formulák nyelve.

Tétel. [S. A. Cook, L. Levin, 1971]

A SAT nyelv NP-teljes.

Cook–Levin-tétel

Boole-formula:

$$\text{pl. } (x_1 \vee \overline{x_2} \vee x_5) \wedge (\overline{x_3} \vee x_2 \vee x_6 \vee x_1) \wedge \overline{(x_5 \vee x_6)}$$

Definíció. *SAT nyelv:* a kielégíthető Boole-formulák nyelve.

Tétel. [S. A. Cook, L. Levin, 1971]

A SAT nyelv NP-teljes.

Bizonyítás: $\text{SAT} \in \text{NP}$, mert egy kielégítés (értékkadás a változóknak) megfelelő tanú ✓

Cook–Levin-tétel

Boole-formula:

$$\text{pl. } (x_1 \vee \overline{x_2} \vee x_5) \wedge (\overline{x_3} \vee x_2 \vee x_6 \vee x_1) \wedge \overline{(x_5 \vee x_6)}$$

Definíció. *SAT nyelv:* a kielégíthető Boole-formulák nyelve.

Tétel. [S. A. Cook, L. Levin, 1971]

A SAT nyelv NP-teljes.

Bizonyítás: $\text{SAT} \in \text{NP}$, mert egy kielégítés (értékkadás a változóknak) megfelelő tanú ✓

Be kell látni, hogy $\forall L \in \text{NP}$ nyelvre létezik egy $L \preceq \text{SAT}$ Karp-redukció

Cook–Levin-tétel

Boole-formula:

$$\text{pl. } (x_1 \vee \overline{x_2} \vee x_5) \wedge (\overline{x_3} \vee x_2 \vee x_6 \vee x_1) \wedge \overline{(x_5 \vee x_6)}$$

Definíció. *SAT nyelv:* a kielégíthető Boole-formulák nyelve.

Tétel. [S. A. Cook, L. Levin, 1971]

A SAT nyelv NP-teljes.

Bizonyítás: $SAT \in NP$, mert egy kielégítés (értékkadás a változóknak) megfelelő tanú ✓

Be kell látni, hogy $\forall L \in NP$ nyelvre létezik egy $L \prec SAT$ Karp-redukció
 $\implies (x \in L?)$ kérdés tetszőleges $x \in I^*$ inputjához meg kell adnunk egy ϕ Boole-formulát, mely pontosan akkor kielégíthető, ha $x \in L$.

Cook–Levin-tétel

Boole-formula:

$$\text{pl. } (x_1 \vee \overline{x_2} \vee x_5) \wedge (\overline{x_3} \vee x_2 \vee x_6 \vee x_1) \wedge \overline{(x_5 \vee x_6)}$$

Definíció. *SAT nyelv:* a kielégíthető Boole-formulák nyelve.

Tétel. [S. A. Cook, L. Levin, 1971]

A SAT nyelv NP-teljes.

Bizonyítás: $\text{SAT} \in \text{NP}$, mert egy kielégítés (értékkadás a változóknak) megfelelő tanú ✓

Be kell látni, hogy $\forall L \in \text{NP}$ nyelvre létezik egy $L \prec \text{SAT}$ Karp-redukció

$\implies (x \in L?)$ kérdés tetszőleges $x \in I^*$ inputjához meg kell adnunk egy ϕ Boole-formulát, mely pontosan akkor kielégíthető, ha $x \in L$.

\implies tanú-tétel miatt elég leírni Boole-formulával az (x, y) -t felismerő TG-t

$$0x[i, j] = \begin{cases} 1 & \text{ha az } i\text{-edik lépés után a } j\text{-edik cella tartalma } 0 \\ 0 & \text{különben} \end{cases}$$

$$1x[i, j] = \begin{cases} 1 & \text{ha az } i\text{-edik lépés után a } j\text{-edik cella tartalma } 1 \\ 0 & \text{különben} \end{cases}$$

$$\ddot{x}[i, j] = \begin{cases} 1 & \text{ha az } i\text{-edik lépés után a } j\text{-edik cella tartalma } \ddot{u} \\ 0 & \text{különben} \end{cases}$$

$$f[i, j] = \begin{cases} 1 & \text{ha az } i\text{-edik lépés után a fej } j\text{-edik cellán áll} \\ 0 & \text{különben} \end{cases}$$

$$q[i, s] = \begin{cases} 1 & \text{ha az } i\text{-edik lépés után } M \text{ belső állapota } q_s \\ 0 & \text{különben.} \end{cases}$$

Le kell írni \implies a szalag egy mezőjén minden időpontban éppen egy szalagjel van; a fej minden időpontban a szalag egyetlen celláján van; a gép minden időpontban egyetlen állapotban van.

Le kell írni \implies a szalag egy mezőjén minden időpontban éppen egy szalagjel van; a fej minden időpontban a szalag egyetlen celláján van; a gép minden időpontban egyetlen állapotban van.

pl. az első: $(0 \leq i \leq n^c, 1 \leq j \leq n^c)$ párra

$$(0x[i, j] \vee 1x[i, j] \vee \ddot{u}x[i, j]) \wedge \\ \wedge (\overline{0x[i, j]} \vee \overline{1x[i, j]}) \wedge (\overline{0x[i, j]} \vee \overline{\ddot{u}x[i, j]}) \wedge (\overline{1x[i, j]} \vee \overline{\ddot{u}x[i, j]}).$$

Le kell írni \implies a szalag egy mezőjén minden időpontban éppen egy szalagjel van; a fej minden időpontban a szalag egyetlen celláján van; a gép minden időpontban egyetlen állapotban van.

pl. az első: $(0 \leq i \leq n^c, 1 \leq j \leq n^c)$ párra

$$(0x[i, j] \vee 1x[i, j] \vee \ddot{u}x[i, j]) \wedge$$

$$\wedge (\overline{0x[i, j]} \vee \overline{1x[i, j]}) \wedge (\overline{0x[i, j]} \vee \overline{\ddot{u}x[i, j]}) \wedge (\overline{1x[i, j]} \vee \overline{\ddot{u}x[i, j]}).$$

Összesen $(n^c + 1)n^c$ ilyen formula

Le kell írni \implies a szalag egy mezőjén minden időpontban éppen egy szalagjel van; a fej minden időpontban a szalag egyetlen celláján van; a gép minden időpontban egyetlen állapotban van.

pl. az első: $(0 \leq i \leq n^c, 1 \leq j \leq n^c)$ párra

$$(0x[i, j] \vee 1x[i, j] \vee \ddot{u}x[i, j]) \wedge$$

$$\wedge (\overline{0x[i, j]} \vee \overline{1x[i, j]}) \wedge (\overline{0x[i, j]} \vee \overline{\ddot{u}x[i, j]}) \wedge (\overline{1x[i, j]} \vee \overline{\ddot{u}x[i, j]}).$$

Összesen $(n^c + 1)n^c$ ilyen formula

Átmenet függvény leírása: pl. $\delta(q_s, 1) = (q_k, 0, bal) \implies$

Le kell írni \implies a szalag egy mezőjén minden időpontban éppen egy szalagjel van; a fej minden időpontban a szalag egyetlen celláján van; a gép minden időpontban egyetlen állapotban van.

pl. az első: $(0 \leq i \leq n^c, 1 \leq j \leq n^c)$ párra

$$(0x[i, j] \vee 1x[i, j] \vee \ddot{u}x[i, j]) \wedge$$

$$\wedge (\overline{0x[i, j]} \vee \overline{1x[i, j]}) \wedge (\overline{0x[i, j]} \vee \overline{\ddot{u}x[i, j]}) \wedge (\overline{1x[i, j]} \vee \overline{\ddot{u}x[i, j]}).$$

Összesen $(n^c + 1)n^c$ ilyen formula

Átmenet függvény leírása: pl. $\delta(q_s, 1) = (q_k, 0, bal) \implies$

$$\left((q[i, s] \wedge 1x[i, l] \wedge f[i, l]) \longrightarrow (q[i + 1, k] \wedge 0x[i + 1, l] \wedge f[i + 1, l - 1]) \right)$$

Le kell írni \implies a szalag egy mezőjén minden időpontban éppen egy szalagjel van; a fej minden időpontban a szalag egyetlen celláján van; a gép minden időpontban egyetlen állapotban van.

pl. az első: $(0 \leq i \leq n^c, 1 \leq j \leq n^c)$ párra

$$(0x[i, j] \vee 1x[i, j] \vee \ddot{u}x[i, j]) \wedge$$

$$\wedge (\overline{0x[i, j]} \vee \overline{1x[i, j]}) \wedge (\overline{0x[i, j]} \vee \overline{\ddot{u}x[i, j]}) \wedge (\overline{1x[i, j]} \vee \overline{\ddot{u}x[i, j]}).$$

Összesen $(n^c + 1)n^c$ ilyen formula

Átmenet függvény leírása: pl. $\delta(q_s, 1) = (q_k, 0, bal) \implies$

$$\left((q[i, s] \wedge 1x[i, l] \wedge f[i, l]) \longrightarrow (q[i + 1, k] \wedge 0x[i + 1, l] \wedge f[i + 1, l - 1]) \right)$$

$O(n^{2c})$ ilyen formula

Le kell írni \implies a szalag egy mezőjén minden időpontban éppen egy szalagjel van; a fej minden időpontban a szalag egyetlen celláján van; a gép minden időpontban egyetlen állapotban van.

pl. az első: $(0 \leq i \leq n^c, 1 \leq j \leq n^c)$ párra

$$(0x[i, j] \vee 1x[i, j] \vee \ddot{u}x[i, j]) \wedge \\ \wedge (\overline{0x[i, j]} \vee \overline{1x[i, j]}) \wedge (\overline{0x[i, j]} \vee \overline{\ddot{u}x[i, j]}) \wedge (\overline{1x[i, j]} \vee \overline{\ddot{u}x[i, j]}).$$

Összesen $(n^c + 1)n^c$ ilyen formula

Átmenet függvény leírása: pl. $\delta(q_s, 1) = (q_k, 0, bal) \implies$

$$\left((q[i, s] \wedge 1x[i, l] \wedge f[i, l]) \longrightarrow (q[i + 1, k] \wedge 0x[i + 1, l] \wedge f[i + 1, l - 1]) \right)$$

$O(n^{2c})$ ilyen formula

Ahol **nincs fej**, nincs változás:

$$\overline{f[i, l]} \longrightarrow (0x[i, l] \leftrightarrow 0x[i + 1, l])$$

Le kell írni \implies a szalag egy mezőjén minden időpontban éppen egy szalagjel van; a fej minden időpontban a szalag egyetlen celláján van; a gép minden időpontban egyetlen állapotban van.

pl. az első: $(0 \leq i \leq n^c, 1 \leq j \leq n^c)$ párra

$$(0x[i, j] \vee 1x[i, j] \vee \ddot{u}x[i, j]) \wedge \\ \wedge (\overline{0x[i, j]} \vee \overline{1x[i, j]}) \wedge (\overline{0x[i, j]} \vee \overline{\ddot{u}x[i, j]}) \wedge (\overline{1x[i, j]} \vee \overline{\ddot{u}x[i, j]}).$$

Összesen $(n^c + 1)n^c$ ilyen formula

Átmenet függvény leírása: pl. $\delta(q_s, 1) = (q_k, 0, bal) \implies$

$$\left((q[i, s] \wedge 1x[i, l] \wedge f[i, l]) \longrightarrow (q[i + 1, k] \wedge 0x[i + 1, l] \wedge f[i + 1, l - 1]) \right)$$

$O(n^{2c})$ ilyen formula

Ahol **nincs fej**, nincs változás:

$$\overline{f[i, l]} \longrightarrow (0x[i, l] \leftrightarrow 0x[i + 1, l])$$

$O(n^{2c})$ ilyen formula (1-re és ü-re is)

Kezdő helyzet:

$$q[0, 0] \wedge f[0, 1]$$

Kezdő helyzet:

$$q[0, 0] \wedge f[0, 1]$$

Input leírása:

$$0x[0, 1] \wedge 1x[0, 2] \wedge 0x[0, 3] \dots$$

Kezdő helyzet:

$$q[0, 0] \wedge f[0, 1]$$

Input leírása:

$$0x[0, 1] \wedge 1x[0, 2] \wedge 0x[0, 3] \dots$$

Elfogadó állapot:

$$1x[n^c, 1]$$

Kezdő helyzet:

$$q[0, 0] \wedge f[0, 1]$$

Input leírása:

$$0x[0, 1] \wedge 1x[0, 2] \wedge 0x[0, 3] \dots$$

Elfogadó állapot:

$$1x[n^c, 1]$$

Minden ilyen $i = 0, 1, 2, \dots, n^c$ -re ÉS-sel

Kezdő helyzet:

$$q[0, 0] \wedge f[0, 1]$$

Input leírása:

$$0x[0, 1] \wedge 1x[0, 2] \wedge 0x[0, 3] \dots$$

Elfogadó állapot:

$$1x[n^c, 1]$$

Minden ilyen $i = 0, 1, 2, \dots, n^c$ -re ÉS-sel

Szabadsági fok: a súgás helyén nincs megkötve semmi

Kezdő helyzet:

$$q[0, 0] \wedge f[0, 1]$$

Input leírása:

$$0x[0, 1] \wedge 1x[0, 2] \wedge 0x[0, 3] \dots$$

Elfogadó állapot:

$$1x[n^c, 1]$$

Minden ilyen $i = 0, 1, 2, \dots, n^c$ -re ÉS-sel

Szabadsági fok: a sűgás helyén nincs megkötve semmi

Ez a Boole formula akkor és csak akkor kielégíthető, ha van megfelelő sűgás x -hez. ✓

További NP-teljes feladatok

Tétel. *Ha az L_1 nyelv NP-teljes, $L_2 \in \text{NP}$ és $L_1 \prec L_2$, akkor L_2 is NP-teljes.*

Bizonyítás: Láttuk, hogy a Karp-redukció tranzitív

További NP-teljes feladatok

Tétel. Ha az L_1 nyelv NP-teljes, $L_2 \in \text{NP}$ és $L_1 \prec L_2$, akkor L_2 is NP-teljes.

Bizonyítás: Láttuk, hogy a Karp-redukció tranzitív

Ha $L_1 \prec L_2$ és $L' \prec L_1 \forall L' \in \text{NP-teljes}$

További NP-teljes feladatok

Tétel. Ha az L_1 nyelv NP-teljes, $L_2 \in \text{NP}$ és $L_1 \prec L_2$, akkor L_2 is NP-teljes.

Bizonyítás: Láttuk, hogy a Karp-redukció tranzitív

Ha $L_1 \prec L_2$ és $L' \prec L_1 \forall L' \in \text{NP-teljes}$

$\implies L' \prec L_2 \forall L' \in \text{NP-teljes}$ ✓

További NP-teljes feladatok

Tétel. Ha az L_1 nyelv NP-teljes, $L_2 \in \text{NP}$ és $L_1 \prec L_2$, akkor L_2 is NP-teljes.

Bizonyítás: Láttuk, hogy a Karp-redukció tranzitív

Ha $L_1 \prec L_2$ és $L' \prec L_1 \forall L' \in \text{NP-teljes}$

$\implies L' \prec L_2 \forall L' \in \text{NP-teljes}$ ✓

Nem kell már *minden* NP-beli nyelvet az L_2 -re redukálni; elég ezt megtenni *egyetlen* NP-teljes L_1 nyelvvel.

További NP-teljes feladatok

Tétel. Ha az L_1 nyelv NP-teljes, $L_2 \in \text{NP}$ és $L_1 \prec L_2$, akkor L_2 is NP-teljes.

Bizonyítás: Láttuk, hogy a Karp-redukció tranzitív

Ha $L_1 \prec L_2$ és $L' \prec L_1 \forall L' \in \text{NP-teljes}$

$\implies L' \prec L_2 \forall L' \in \text{NP-teljes}$ ✓

Nem kell már *minden* NP-beli nyelvet az L_2 -re redukálni; elég ezt megtenni *egyetlen* NP-teljes L_1 nyelvvel.

Definíció. Ha az L_2 nyelvről csak azt tudjuk, hogy van olyan NP-teljes L_1 nyelv, melyre $L_1 \prec L_2$, akkor L_2 -t **NP-nehéz** nyelvnek nevezzük. Az előző állítás szerint L_2 pontosan akkor NP-teljes, ha NP-beli és ugyanakkor NP-nehéz is.