Različice igre Hanojski stolp

Sandi Klavžar

Fakulteta za matematiko in fiziko, Univerza v Ljubljani

Fakulteta za naravoslovje in matematiko, Univerza v Mariboru

Moderni izzivi poučevanja matematike
FMF UL, 21. september 2013

Klasični problem

Édouard Lucas: pokrov originalne igre iz leta 1883

N. Claus (de Siam)

Lucas d'Amiens

Legenda

- 64 zlatih diskov.

Legenda

- 64 zlatih diskov.
- 3 diamantne igle.

Legenda

- 64 zlatih diskov.
- 3 diamantne igle.
- Naloga: Prestavi vse diske z ene igle na drugo iglo; upoštevaj božansko pravilo (angl. divine rule).

Legenda

- 64 zlatih diskov.
- 3 diamantne igle.
- Naloga: Prestavi vse diske z ene igle na drugo iglo; upoštevaj božansko pravilo (angl. divine rule).
- Ko bo naloga opravljena, bo konec sveta.

Legenda

- 64 zlatih diskov.
- 3 diamantne igle.
- Naloga: Prestavi vse diske z ene igle na drugo iglo; upoštevaj božansko pravilo (angl. divine rule).
- Ko bo naloga opravljena, bo konec sveta.
- Optimalna rešitev enolična in zahteva $2^{64} - 1$ potez.

Legenda

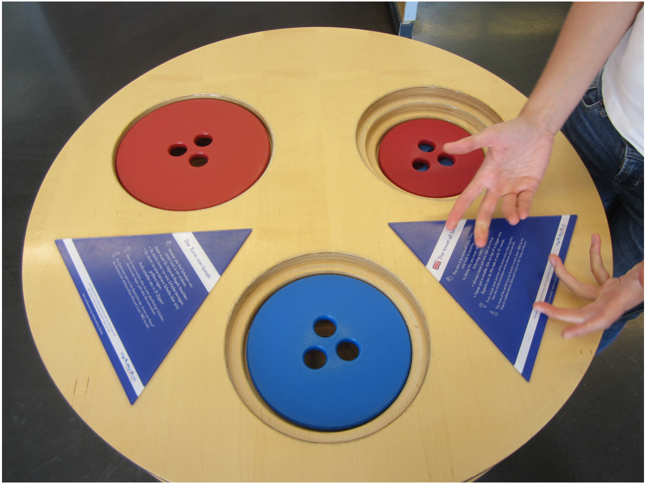
- 64 zlatih diskov.
- 3 diamantne igle.
- Naloga: Prestavi vse diske z ene igle na drugo iglo; upoštevaj božansko pravilo (angl. divine rule).
- Ko bo naloga opravljena, bo konec sveta.
- Optimalna rešitev enolična in zahteva $2^{64} - 1$ potez.
- $2^{64} - 1 = 18446744073709551615$.

Legenda

- 64 zlatih diskov.
- 3 diamantne igle.
- Naloga: Prestavi vse diske z ene igle na drugo iglo; upoštevaj božansko pravilo (angl. divine rule).
- Ko bo naloga opravljena, bo konec sveta.
- Optimalna rešitev enolična in zahteva $2^{64} - 1$ potez.
- $2^{64} - 1 = 18446744073709551615$.
- Poteza/sekundo: $5.849424173 \cdot 10^{11}$ let $=$ 585 milijard let.

Mathematikum Gießen - “riž na šahovnici”

Mathematikum Gießen - “The Tower of Ionah”



Rekurzivna rešitev

Procedure ToH(n, i, j)
Parameter n: število diskov
Parameter i: izvorni nosilec
Parameter j: ciljni nosilec

Rekurzivna rešitev

Procedure ToH(n, i, j)
Parameter n: število diskov
Parameter i: izvorni nosilec
Parameter j: ciljni nosilec
če $n \neq 0$ in $i \neq j$ **potem**

Rekurzivna rešitev

Procedure ToH(n, i, j)
Parameter n: število diskov
Parameter i: izvorni nosilec
Parameter j: ciljni nosilec
  če $n \neq 0$ in $i \neq j$ **potem**
    $k \leftarrow 3 - i - j$

Rekurzivna rešitev

Procedure ToH(n, i, j)
Parameter n: število diskov
Parameter i: izvorni nosilec
Parameter j: ciljni nosilec
  če $n \neq 0$ in $i \neq j$ **potem**
    $k \leftarrow 3 - i - j$
    ToH$(n-1, i, k)$

Rekurzivna rešitev

Procedure ToH(n, i, j)
Parameter n: število diskov
Parameter i: izvorni nosilec
Parameter j: ciljni nosilec
    če $n \neq 0$ in $i \neq j$ **potem**
        $k \leftarrow 3 - i - j$
        ToH$(n - 1, i, k)$
        premakni disk n z nosilca i na nosilec j

Rekurzivna rešitev

Procedure ToH(n, i, j)
Parameter n: število diskov
Parameter i: izvorni nosilec
Parameter j: ciljni nosilec
  če $n \neq 0$ in $i \neq j$ **potem**
    $k \leftarrow 3 - i - j$
    ToH$(n-1, i, k)$
    premakni disk n z nosilca i na nosilec j
    ToH$(n-1, k, j)$

Oliveova rešitev

Procedure Olive(n, i, j)
Parameter n, i, j: število diskov, izvorni nosilec, ciljni nosilec

Oliveova rešitev

Procedure Olive(n, i, j)
Parameter n, i, j: število diskov, izvorni nosilec, ciljni nosilec
če n lih **potem**
  premakni disk 1 z nosilca i na nosilec j
sicer
  premakni disk 1 z nosilca i na nosilec $3 - i - j$
konec če

Oliveova rešitev

Procedure Olive(n, i, j)
Parameter n, i, j: število diskov, izvorni nosilec, ciljni nosilec
če n lih **potem**
  premakni disk 1 z nosilca i na nosilec j
sicer
  premakni disk 1 z nosilca i na nosilec $3 - i - j$
konec če
zapomni si smer gibanja diska 1

Oliveova rešitev

Procedure Olive(n, i, j)
Parameter n, i, j: število diskov, izvorni nosilec, ciljni nosilec
če n lih **potem**
  premakni disk 1 z nosilca i na nosilec j
sicer
  premakni disk 1 z nosilca i na nosilec $3 - i - j$
konec če
zapomni si smer gibanja diska 1
dokler niso vsi diski na nosilcu j

Oliveova rešitev

Procedure Olive(n, i, j)
Parameter n, i, j: število diskov, izvorni nosilec, ciljni nosilec
če n lih **potem**
  premakni disk 1 z nosilca i na nosilec j
sicer
  premakni disk 1 z nosilca i na nosilec $3 - i - j$
konec če
zapomni si smer gibanja diska 1
dokler niso vsi diski na nosilcu j
  naredi legalen premik diska različnega od 1

Oliveova rešitev

Procedure Olive(n, i, j)
Parameter n, i, j: število diskov, izvorni nosilec, ciljni nosilec
če n lih **potem**
  premakni disk 1 z nosilca i na nosilec j
sicer
  premakni disk 1 z nosilca i na nosilec $3 - i - j$
konec če
zapomni si smer gibanja diska 1
dokler niso vsi diski na nosilcu j
  naredi legalen premik diska različnega od 1
  naredi premik diska 1 ciklično v pravi smeri
konec dokler

Hanojski grafi

Problem lahko na naraven način modeliramo z grafi. Formalno:

Hanojski grafi

Problem lahko na naraven način modeliramo z grafi. Formalno:

- Regularno stanje: $s = s_n \ldots s_1 \in \{0, 1, 2\}^n$, s_i je nosilec, na katerem je disk i

Hanojski grafi

Problem lahko na naraven način modeliramo z grafi. Formalno:

- Regularno stanje: $s = s_n \ldots s_1 \in \{0, 1, 2\}^n$, s_i je nosilec, na katerem je disk i
- H_3^n: vozlišča regularna stanja: $V(H_3^n) = \{0, 1, 2\}^n$.

Hanojski grafi

Problem lahko na naraven način modeliramo z grafi. Formalno:

- Regularno stanje: $s = s_n \dots s_1 \in \{0,1,2\}^n$, s_i je nosilec, na katerem je disk i
- H_3^n: vozlišča regularna stanja: $V(H_3^n) = \{0,1,2\}^n$.
- Povezave: iz stanja v stanje s premikom enega diska:

$$E(H_3^n) = \left\{ \{\underline{s}i(3-i-j)^{d-1}, \underline{s}j(3-i-j)^{d-1}\} \mid \underline{s} \in \{0,1,2\}^{n-d} \right\} .$$

Hanojski grafi

Problem lahko na naraven način modeliramo z grafi. Formalno:

- Regularno stanje: $s = s_n \ldots s_1 \in \{0, 1, 2\}^n$, s_i je nosilec, na katerem je disk i
- H_3^n: vozlišča regularna stanja: $V(H_3^n) = \{0, 1, 2\}^n$.
- Povezave: iz stanja v stanje s premikom enega diska:

$$E(H_3^n) = \left\{ \{\underline{s} i (3-i-j)^{d-1}, \underline{s} j (3-i-j)^{d-1}\} \mid \underline{s} \in \{0, 1, 2\}^{n-d} \right\} .$$

- $|V(H_3^n)| = 3^n$.

Hanojski grafi

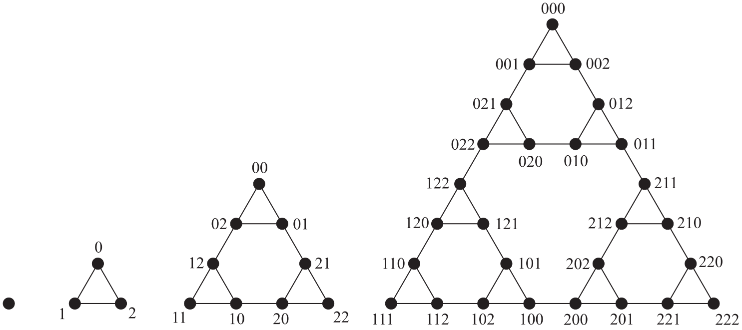
Problem lahko na naraven način modeliramo z grafi. Formalno:

- Regularno stanje: $s = s_n \ldots s_1 \in \{0,1,2\}^n$, s_i je nosilec, na katerem je disk i
- H_3^n: vozlišča regularna stanja: $V(H_3^n) = \{0,1,2\}^n$.
- Povezave: iz stanja v stanje s premikom enega diska:

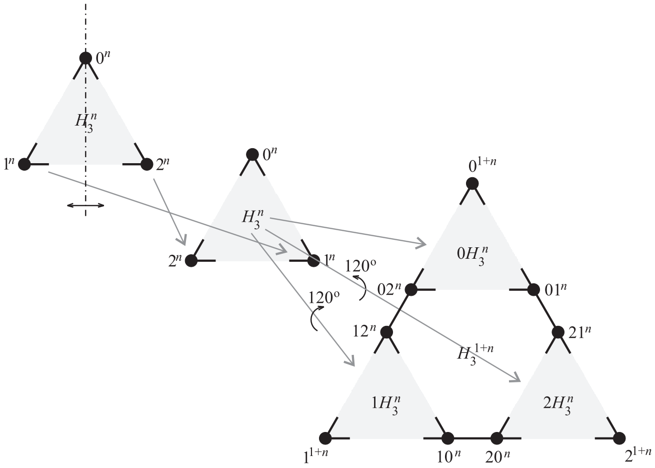
$$E(H_3^n) = \left\{ \{\underline{s}i(3-i-j)^{d-1}, \underline{s}j(3-i-j)^{d-1}\} \mid \underline{s} \in \{0,1,2\}^{n-d} \right\}.$$

- $|V(H_3^n)| = 3^n$.
- $|E(H_3^n)| = \frac{3}{2}(3^n - 1)$.

Hanojski grafi - nadaljevanje



Hanojski grafi - nadaljevanje



Inačice problema

Kaj je inačica igre Hanojskega stolpa?

- Nosilce razlikujemo med seboj.

Kaj je inačica igre Hanojskega stolpa?

- Nosilce razlikujemo med seboj.
- Diske razlikujemo ned seboj.

Kaj je inačica igre Hanojskega stolpa?

- Nosilce razlikujemo med seboj.
- Diske razlikujemo ned seboj.
- Diski so ves čas na nosilcih, razen ko jih premikamo.

Kaj je inačica igre Hanojskega stolpa?

- Nosilce razlikujemo med seboj.
- Diske razlikujemo ned seboj.
- Diski so ves čas na nosilcih, razen ko jih premikamo.
- Lahko premaknemo le disk (diske) z vrha kupa.

Kaj je inačica igre Hanojskega stolpa?

- Nosilce razlikujemo med seboj.
- Diske razlikujemo ned seboj.
- Diski so ves čas na nosilcih, razen ko jih premikamo.
- Lahko premaknemo le disk (diske) z vrha kupa.
- Naloga: Za podano začetno porazdelitev diskov in podano končno porazdelitev diskov poišči najkrajše zaporedje premikov, ki ob danih pravilih preoblikuje začetno porazdelitev v končno porazdelitev.

Še vedno veliko ("neskončno") možnosti

- Lahko imamo poljubno število nosilcev.

Še vedno veliko (“neskončno”) možnosti

- Lahko imamo poljubno število nosilcev.
- Nosilce lahko razlikujemo tudi po njihovi višini, torej glede na število diskov, ki jih lahko nosijo.

Še vedno veliko ("neskončno") možnosti

- Lahko imamo poljubno število nosilcev.
- Nosilce lahko razlikujemo tudi po njihovi višini, torej glede na število diskov, ki jih lahko nosijo.
- Diske lahko razlikujemo po velikosti in/ali barvi.

Še vedno veliko (“neskončno”) možnosti

- Lahko imamo poljubno število nosilcev.
- Nosilce lahko razlikujemo tudi po njihovi višini, torej glede na število diskov, ki jih lahko nosijo.
- Diske lahko razlikujemo po velikosti in/ali barvi.
- Lahko dopuščamo določena neregularna stanja (glede na klasična pravila).

Še vedno veliko ("neskončno") možnosti

- Lahko imamo poljubno število nosilcev.
- Nosilce lahko razlikujemo tudi po njihovi višini, torej glede na število diskov, ki jih lahko nosijo.
- Diske lahko razlikujemo po velikosti in/ali barvi.
- Lahko dopuščamo določena neregularna stanja (glede na klasična pravila).
- Lahko premaknemo več kot en disk hkrati.

Še vedno veliko (“neskončno”) možnosti

- Lahko imamo poljubno število nosilcev.
- Nosilce lahko razlikujemo tudi po njihovi višini, torej glede na število diskov, ki jih lahko nosijo.
- Diske lahko razlikujemo po velikosti in/ali barvi.
- Lahko dopuščamo določena neregularna stanja (glede na klasična pravila).
- Lahko premaknemo več kot en disk hkrati.
- Lahko imamo dodatne omejitve za premikanje, ali pa tudi ohlapnejša pravila premikanja; slednja lahko tudi kršijo božansko pravilo.

Še vedno veliko (“neskončno”) možnosti

- Lahko imamo poljubno število nosilcev.
- Nosilce lahko razlikujemo tudi po njihovi višini, torej glede na število diskov, ki jih lahko nosijo.
- Diske lahko razlikujemo po velikosti in/ali barvi.
- Lahko dopuščamo določena neregularna stanja (glede na klasična pravila).
- Lahko premaknemo več kot en disk hkrati.
- Lahko imamo dodatne omejitve za premikanje, ali pa tudi ohlapnejša pravila premikanja; slednja lahko tudi kršijo božansko pravilo.
- In, seveda, poljubne kombinacije naštetega.

Več kot trije nosilci

- Povsem ista naloga, le da imamo štiri nosilce (ali več).

Več kot trije nosilci

- Povsem ista naloga, le da imamo štiri nosilce (ali več).
- Določi optimalno število potez! Problem iz American Mathematical Monthy, 1939.

Več kot trije nosilci

- Povsem ista naloga, le da imamo štiri nosilce (ali več).
- Določi optimalno število potez! Problem iz American Mathematical Monthy, 1939.
- Frame-Stewartov algoritem, 1941.

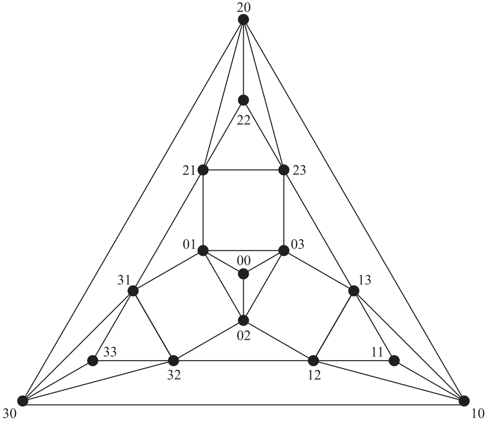
Več kot trije nosilci

- Povsem ista naloga, le da imamo štiri nosilce (ali več).
- Določi optimalno število potez! Problem iz American Mathematical Monthy, 1939.
- Frame-Stewartov algoritem, 1941.
- Preverjeno do $n = 30$ (Korf, 2008).

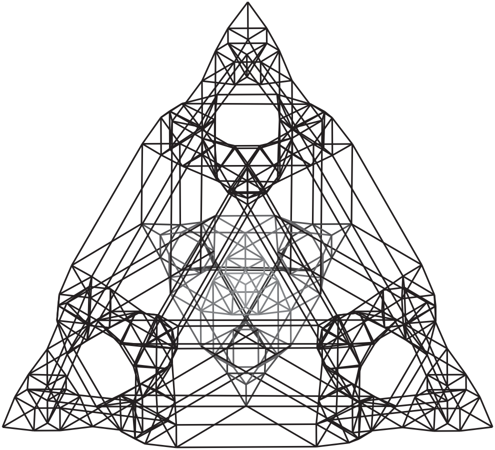
Več kot trije nosilci

- Povsem ista naloga, le da imamo štiri nosilce (ali več).
- Določi optimalno število potez! Problem iz American Mathematical Monthy, 1939.
- Frame-Stewartov algoritem, 1941.
- Preverjeno do $n = 30$ (Korf, 2008).
- Katera naloga je najbolj zahtevna? Fenomen $n = 15, 20$!

Več kot trije nosilci - H_4^2

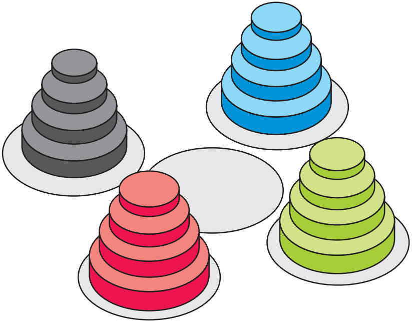


Več kot trije nosilci - H_4^4



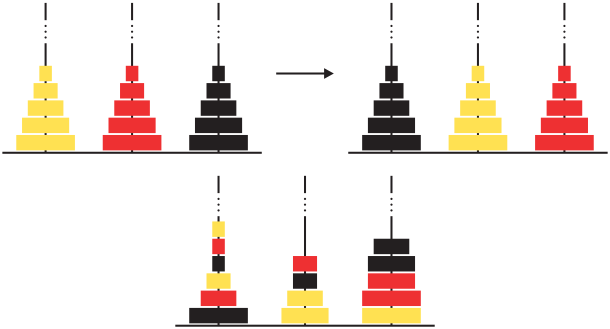
Lucasova inačica iz leta 1889

- 16 diskov paroma različnih velikosti.
- Naloga: spraviti diske na prazni nosilec.
- Optimalna rešitev 63 premikov (računalnik).
- 2. inačica: štirje stolpi s po štirimi diski urejeni po velikosti.
- Optimalna rešitev sedaj 54 premikov (računalnik).



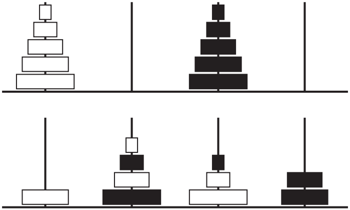
Antwerpenski stolp

- 3 stolpi, $3 \times n$ identičnih diskov—razen v barvi.
- Vsak nosilec lahko nosi poljubno število diskov.
- Običajno božansko pravilo, enaka velikost dovoljena.
- Naloga: stanje s permutiranimi stolpi.
- Izrek: optimalna rešitev zahteva $3 \cdot 2^{n+2} - 8n - 10$ premikov.



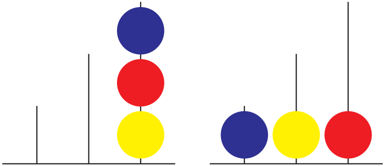
Črno-beli stolp

- 4 stolpi, $2 \times n$ identičnih diskov—razen v barvi.
- Vsak nosilec lahko nosi poljubno število diskov.
- Običajno božansko pravilo z dodatkom, da ne smemo dati diska na disk iste velikosti.
- Naloga: zamenjati beli in črni stolp.
- Izrek: optimalna rešitev zahteva $3 \cdot 2^{n+1} - 1$ premikov.

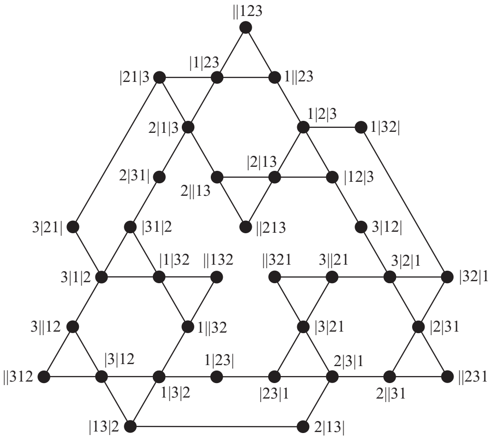


Londonski stolp

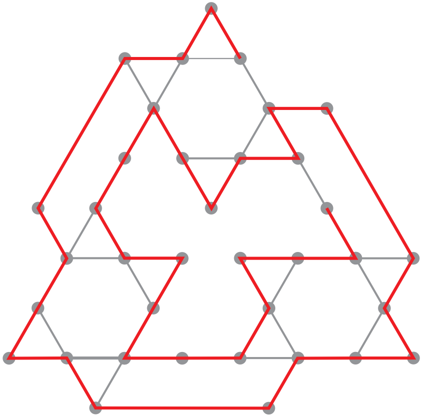
- 3 stolpi, ki lahko nosijo 1, 2, ali 3 "diske".
- 3 "diski", ki se razlikujejo v barvi.
- Naloga: iz podanega stanja doseči ciljno stanje z minimalnim številom potez.
- Graf stanj 36 vozlišč, po 12 stopnje 2, 3 in 4.



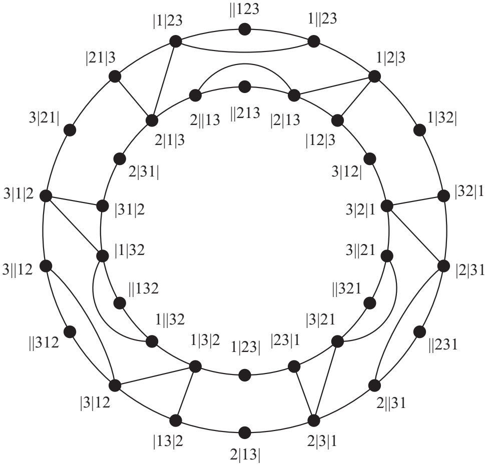
Londonski stolp - graf stanj L



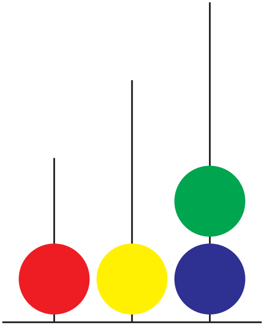
Londonski stolp - hamiltonova pot v L



Londonski stolp - ponovno L



Londonski stolp - posplošitev L_h^n (L_{234}^4)



Hanojski stolp z usmerjenimi premiki diskov

- Klasični problem: 3 stolpi, n diskov.

Hanojski stolp z usmerjenimi premiki diskov

- Klasični problem: 3 stolpi, n diskov.
- Dodatek: prepovemo premike med urejenimi pari izbranih nosilcev.

Hanojski stolp z usmerjenimi premiki diskov

- Klasični problem: 3 stolpi, n diskov.
- Dodatek: prepovemo premike med urejenimi pari izbranih nosilcev.
- Naloga: iz popolnega stanja v drugo polno stanje z minimalnim številom potez.

Hanojski stolp z usmerjenimi premiki diskov

- Klasični problem: 3 stolpi, n diskov.
- Dodatek: prepovemo premike med urejenimi pari izbranih nosilcev.
- Naloga: iz popolnega stanja v drugo polno stanje z minimalnim številom potez.
- Niso vse naloge enake, niso vse rešljive.

Hanojski stolp z usmerjenimi premiki diskov

- Klasični problem: 3 stolpi, n diskov.
- Dodatek: prepovemo premike med urejenimi pari izbranih nosilcev.
- Naloga: iz popolnega stanja v drugo polno stanje z minimalnim številom potez.
- Niso vse naloge enake, niso vse rešljive.
- Problem je natančno določen z ustreznim digrafom D.

Hanojski stolp z usmerjenimi premiki diskov

- Klasični problem: 3 stolpi, n diskov.
- Dodatek: prepovemo premike med urejenimi pari izbranih nosilcev.
- Naloga: iz popolnega stanja v drugo polno stanje z minimalnim številom potez.
- Niso vse naloge enake, niso vse rešljive.
- Problem je natančno določen z ustreznim digrafom D.
- Kratka oznaka problema: $TH(D)$.

Hanojski stolp z usmerjenimi premiki diskov - nadaljevanje

$TH(D)$ je rešljiv, če za vsako izbiro začetnega nosilca in končnega nosilca in za poljubno število diskov obstaja zaporedje legalnih potez, ki reši problem.

Hanojski stolp z usmerjenimi premiki diskov - nadaljevanje

$TH(D)$ je rešljiv, če za vsako izbiro začetnega nosilca in končnega nosilca in za poljubno število diskov obstaja zaporedje legalnih potez, ki reši problem.

Trditev

Linearni TH je rešljiv. Graf stanj je pot na 3^n vozliščih med popolnim stanjem na nosilcu 0 in popolnim stanjem na nosilcu 2.

Hanojski stolp z usmerjenimi premiki diskov - nadaljevanje

$TH(D)$ je rešljiv, če za vsako izbiro začetnega nosilca in končnega nosilca in za poljubno število diskov obstaja zaporedje legalnih potez, ki reši problem.

Trditev

Linearni TH je rešljiv. Graf stanj je pot na 3^n vozliščih med popolnim stanjem na nosilcu 0 in popolnim stanjem na nosilcu 2.

Digraf D je krepko povezan, če za vsaki vozlišči $u, v \in V(D)$ obstaja usmerjena pot iz u v v in usmerjena pot iz v v u.

Hanojski stolp z usmerjenimi premiki diskov - nadaljevanje

$TH(D)$ je rešljiv, če za vsako izbiro začetnega nosilca in končnega nosilca in za poljubno število diskov obstaja zaporedje legalnih potez, ki reši problem.

Trditev

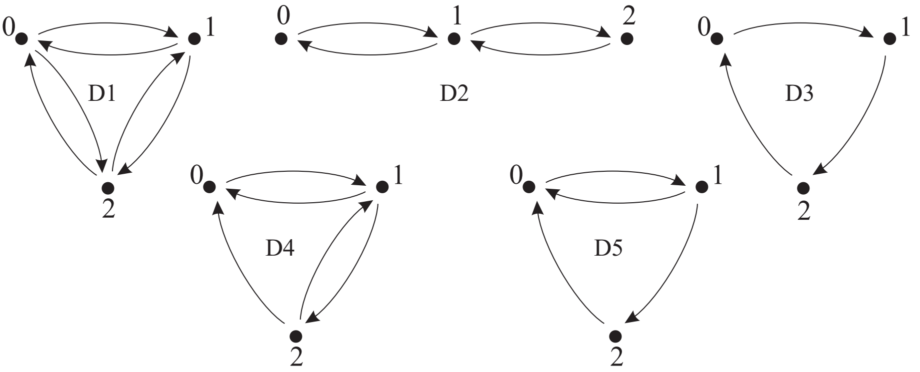
Linearni TH je rešljiv. Graf stanj je pot na 3^n vozliščih med popolnim stanjem na nosilcu 0 in popolnim stanjem na nosilcu 2.

Digraf D je krepko povezan, če za vsaki vozlišči $u, v \in V(D)$ obstaja usmerjena pot iz u v v in usmerjena pot iz v v u.

Izrek

Naj bo D digraf z vsaj tremi vozlišči. Tedaj je $TH(D)$ rešljiv natanko tedaj, ko je D krepko povezan.

Krepko povezani digrafi na treh vozliščih



Več o klasični nalogi

Dodatne naloge

- Kako nadaljevati iz zapuščenega stanja?

Dodatne naloge

- Kako nadaljevati iz zapuščenega stanja?
- Kako ugotoviti, ali smo na optimalni poti?

Dodatne naloge

- Kako nadaljevati iz zapuščenega stanja?
- Kako ugotoviti, ali smo na optimalni poti?
- Kako priti iz poljubnega stanja v popolno stanje?

Dodatne naloge

- Kako nadaljevati iz zapuščenega stanja?
- Kako ugotoviti, ali smo na optimalni poti?
- Kako priti iz poljubnega stanja v popolno stanje?
- Kako priti iz poljubnega stanja v drugo poljubno stanje?

Dodatne naloge

- Kako nadaljevati iz zapuščenega stanja?
- Kako ugotoviti, ali smo na optimalni poti?
- Kako priti iz poljubnega stanja v popolno stanje?
- Kako priti iz poljubnega stanja v drugo poljubno stanje?
- Kako ugotoviti, ali se disk n premakne 1x ali 2x?

Dodatne naloge

- Kako nadaljevati iz zapuščenega stanja?
- Kako ugotoviti, ali smo na optimalni poti?
- Kako priti iz poljubnega stanja v popolno stanje?
- Kako priti iz poljubnega stanja v drugo poljubno stanje?
- Kako ugotoviti, ali se disk n premakne 1x ali 2x?
- Kako priti iz neregularnega stanja v popolno stanje?

Zaporedja brez ponavljanj

Zaporedje $a = a_1, a_2, \ldots, a_n$ je brez ponavljanj, če ne vsebuje podzaporedja

$$a_{i+1}, a_{i+2}, \ldots, a_{i+2m}$$

tako da je

Zaporedja brez ponavljanj

Zaporedje $a = a_1, a_2, \ldots, a_n$ je brez ponavljanj, če ne vsebuje podzaporedja

$$a_{i+1}, a_{i+2}, \ldots, a_{i+2m}$$

tako da je

$$a_{i+j} = a_{i+j+m}, \quad j = 1, \ldots, m\,.$$

Zaporedja brez ponavljanj

Zaporedje $a = a_1, a_2, \ldots, a_n$ je brez ponavljanj, če ne vsebuje podzaporedja

$$a_{i+1}, a_{i+2}, \ldots, a_{i+2m}$$

tako da je

$$a_{i+j} = a_{i+j+m}, \quad j = 1, \ldots, m\,.$$

Tako podzaporedje se imenuje kvadrat, saj ga lahko zapišemo v obliki xx. Zaporedje a se zato imenuje tudi brez kvadrata.

Primeri

- $1, 2, 1, 3, 1, 2, 3, 1, 2, 1$

Primeri

- $1, 2, 1, 3, 1, 2, 3, 1, 2, 1$
- $1, 2, 1, \boxed{3,\ 1,\ 2,}\boxed{3,\ 1,\ 2,}1$

Primeri

- $1, 2, 1, 3, 1, 2, 3, 1, 2, 1$
- $1, 2, 1, \boxed{3,\ 1,\ 2,}\boxed{3,\ 1,\ 2,}1$
- $1,$

Primeri

- $1, 2, 1, 3, 1, 2, 3, 1, 2, 1$
- $1, 2, 1, \boxed{3,\ 1,\ 2,}\boxed{3,\ 1,\ 2,}1$
- $1,\ 2,$

Primeri

- $1, 2, 1, 3, 1, 2, 3, 1, 2, 1$
- $1, 2, 1, \boxed{3,\ 1,\ 2,}\boxed{3,\ 1,\ 2,}1$
- $1,\ 2,\ 3,$

Primeri

- $1, 2, 1, 3, 1, 2, 3, 1, 2, 1$
- $1, 2, 1, \boxed{3,\ 1,\ 2,}\boxed{3,\ 1,\ 2,}1$
- 1, 2, 3, 4,

Primeri

- $1, 2, 1, 3, 1, 2, 3, 1, 2, 1$
- $1, 2, 1, \boxed{3,\ 1,\ 2,}\boxed{3,\ 1,\ 2,}1$
- $1,\ 2,\ 3,\ 4,\ 5,$

Primeri

- $1, 2, 1, 3, 1, 2, 3, 1, 2, 1$
- $1, 2, 1, \boxed{3,\ 1,\ 2,}\boxed{3,\ 1,\ 2,}1$
- $1,\ 2,\ 3,\ 4,\ 5,\ 6,\ \ldots$

Primeri

- $1, 2, 1, 3, 1, 2, 3, 1, 2, 1$
- $1, 2, 1, \boxed{3,\ 1,\ 2,}\boxed{3,\ 1,\ 2,}1$
- $1,\ 2,\ 3,\ 4,\ 5,\ 6,\ \ldots$
- Poiščimo zaporedje brez ponavljanj s požrešnim algoritmom:

Primeri

- $1, 2, 1, 3, 1, 2, 3, 1, 2, 1$
- $1, 2, 1, \boxed{3,\ 1,\ 2,}\boxed{3,\ 1,\ 2,}1$
- $1,\ 2,\ 3,\ 4,\ 5,\ 6,\ \ldots$
- Poiščimo zaporedje brez ponavljanj s požrešnim algoritmom:
- 1,

Primeri

- $1, 2, 1, 3, 1, 2, 3, 1, 2, 1$
- $1, 2, 1, \boxed{3,\ 1,\ 2,}\boxed{3,\ 1,\ 2,}1$
- 1, 2, 3, 4, 5, 6, . . .
- Poiščimo zaporedje brez ponavljanj s požrešnim algoritmom:
- 1, 2,

Primeri

- $1, 2, 1, 3, 1, 2, 3, 1, 2, 1$
- $1, 2, 1, \boxed{3,\ 1,\ 2,}\boxed{3,\ 1,\ 2,}1$
- 1, 2, 3, 4, 5, 6, ...
- Poiščimo zaporedje brez ponavljanj s požrešnim algoritmom:
- 1, 2, 1,

Primeri

- $1, 2, 1, 3, 1, 2, 3, 1, 2, 1$
- $1, 2, 1, \boxed{3,\ 1,\ 2,}\boxed{3,\ 1,\ 2,}1$
- 1, 2, 3, 4, 5, 6, . . .
- Poiščimo zaporedje brez ponavljanj s požrešnim algoritmom:
- 1, 2, 1, 3,

Primeri

- $1, 2, 1, 3, 1, 2, 3, 1, 2, 1$
- $1, 2, 1, \boxed{3,\ 1,\ 2,}\boxed{3,\ 1,\ 2,}1$
- 1, 2, 3, 4, 5, 6, . . .
- Poiščimo zaporedje brez ponavljanj s požrešnim algoritmom:
- 1, 2, 1, 3, 1,

Primeri

- $1, 2, 1, 3, 1, 2, 3, 1, 2, 1$
- $1, 2, 1, \boxed{3,\ 1,\ 2,}\boxed{3,\ 1,\ 2,}1$
- 1, 2, 3, 4, 5, 6, . . .
- Poiščimo zaporedje brez ponavljanj s požrešnim algoritmom:
- 1, 2, 1, 3, 1, 2,

Primeri

- $1, 2, 1, 3, 1, 2, 3, 1, 2, 1$
- $1, 2, 1, \boxed{3,\ 1,\ 2,}\boxed{3,\ 1,\ 2,}1$
- $1,\ 2,\ 3,\ 4,\ 5,\ 6,\ \ldots$
- Poiščimo zaporedje brez ponavljanj s požrešnim algoritmom:
- $1,\ 2,\ 1,\ 3,\ 1,\ 2,\ 1,$

Primeri

- $1, 2, 1, 3, 1, 2, 3, 1, 2, 1$
- $1, 2, 1, \boxed{3,\ 1,\ 2,}\boxed{3,\ 1,\ 2,}1$
- 1, 2, 3, 4, 5, 6, . . .
- Poiščimo zaporedje brez ponavljanj s požrešnim algoritmom:
- 1, 2, 1, 3, 1, 2, 1, 4,

Primeri

- $1, 2, 1, 3, 1, 2, 3, 1, 2, 1$
- $1, 2, 1, \boxed{3,\ 1,\ 2,}\boxed{3,\ 1,\ 2,}1$
- 1, 2, 3, 4, 5, 6, ...
- Poiščimo zaporedje brez ponavljanj s požrešnim algoritmom:
- 1, 2, 1, 3, 1, 2, 1, 4, 1,

Primeri

- $1, 2, 1, 3, 1, 2, 3, 1, 2, 1$
- $1, 2, 1, \boxed{3,\ 1,\ 2,}\boxed{3,\ 1,\ 2,}1$
- 1, 2, 3, 4, 5, 6, . . .
- Poiščimo zaporedje brez ponavljanj s požrešnim algoritmom:
- 1, 2, 1, 3, 1, 2, 1, 4, 1, 2,

Primeri

- $1, 2, 1, 3, 1, 2, 3, 1, 2, 1$
- $1, 2, 1, \boxed{3,\ 1,\ 2,}\boxed{3,\ 1,\ 2,}1$
- $1,\ 2,\ 3,\ 4,\ 5,\ 6,\ \ldots$
- Poiščimo zaporedje brez ponavljanj s požrešnim algoritmom:
- $1,\ 2,\ 1,\ 3,\ 1,\ 2,\ 1,\ 4,\ 1,\ 2,\ 1,$

Primeri

- $1, 2, 1, 3, 1, 2, 3, 1, 2, 1$
- $1, 2, 1, \boxed{3,\ 1,\ 2,}\boxed{3,\ 1,\ 2,}1$
- $1,\ 2,\ 3,\ 4,\ 5,\ 6,\ \ldots$
- Poiščimo zaporedje brez ponavljanj s požrešnim algoritmom:
- $1,\ 2,\ 1,\ 3,\ 1,\ 2,\ 1,\ 4,\ 1,\ 2,\ 1,\ 3,$

Primeri

- $1, 2, 1, 3, 1, 2, 3, 1, 2, 1$
- $1, 2, 1, \boxed{3,\ 1,\ 2,}\boxed{3,\ 1,\ 2,}1$
- 1, 2, 3, 4, 5, 6, . . .
- Poiščimo zaporedje brez ponavljanj s požrešnim algoritmom:
- 1, 2, 1, 3, 1, 2, 1, 4, 1, 2, 1, 3, 1,

Primeri

- $1, 2, 1, 3, 1, 2, 3, 1, 2, 1$
- $1, 2, 1, \boxed{3,\ 1,\ 2,}\boxed{3,\ 1,\ 2,}1$
- 1, 2, 3, 4, 5, 6, . . .
- Poiščimo zaporedje brez ponavljanj s požrešnim algoritmom:
- 1, 2, 1, 3, 1, 2, 1, 4, 1, 2, 1, 3, 1, 2,

Primeri

- $1, 2, 1, 3, 1, 2, 3, 1, 2, 1$
- $1, 2, 1, \boxed{3,\ 1,\ 2,}\boxed{3,\ 1,\ 2,}1$
- 1, 2, 3, 4, 5, 6, . . .
- Poiščimo zaporedje brez ponavljanj s požrešnim algoritmom:
- 1, 2, 1, 3, 1, 2, 1, 4, 1, 2, 1, 3, 1, 2, 1,

Primeri

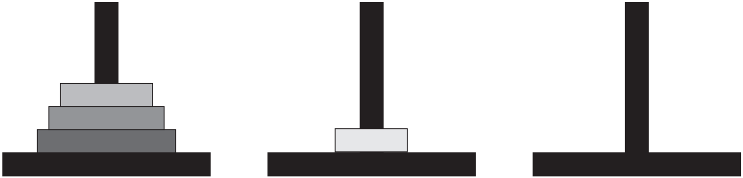
- $1, 2, 1, 3, 1, 2, 3, 1, 2, 1$
- $1, 2, 1, \boxed{3,\ 1,\ 2,}\boxed{3,\ 1,\ 2,}1$
- 1, 2, 3, 4, 5, 6, . . .
- Poiščimo zaporedje brez ponavljanj s požrešnim algoritmom:
- 1, 2, 1, 3, 1, 2, 1, 4, 1, 2, 1, 3, 1, 2, 1, 5,

Primeri

- $1, 2, 1, 3, 1, 2, 3, 1, 2, 1$
- $1, 2, 1, \boxed{3,\ 1,\ 2,}\boxed{3,\ 1,\ 2,}1$
- 1, 2, 3, 4, 5, 6, . . .
- Poiščimo zaporedje brez ponavljanj s požrešnim algoritmom:
- 1, 2, 1, 3, 1, 2, 1, 4, 1, 2, 1, 3, 1, 2, 1, 5, 1, . . .

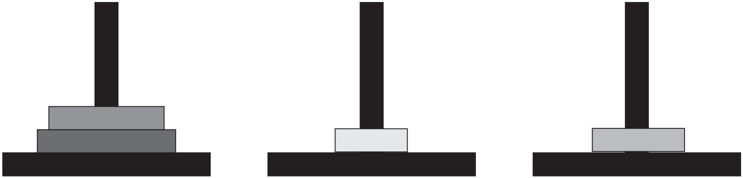
Premiki 4 diskov

Premiki 4 diskov



1

Premiki 4 diskov

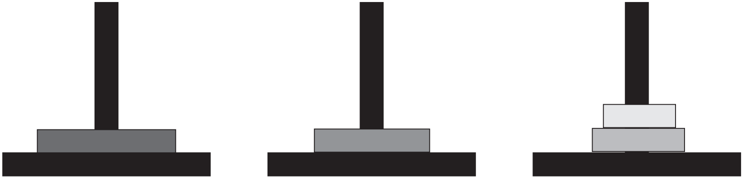


1 2

Premiki 4 diskov

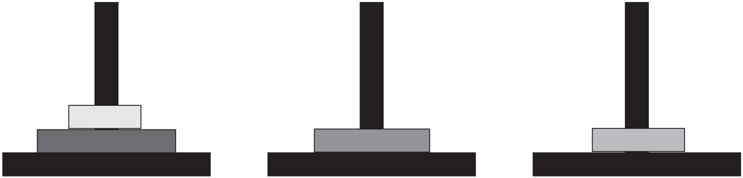
1 2 1

Premiki 4 diskov



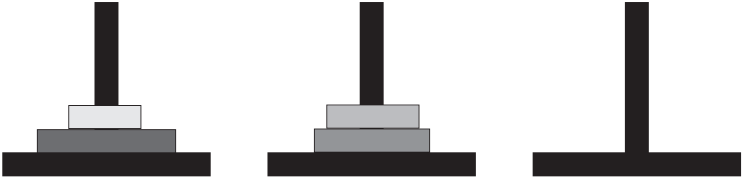
1 2 1 3

Premiki 4 diskov



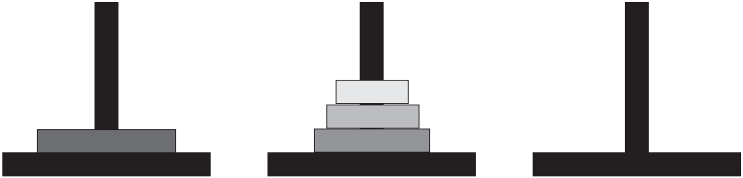
1 2 1 3 1

Premiki 4 diskov



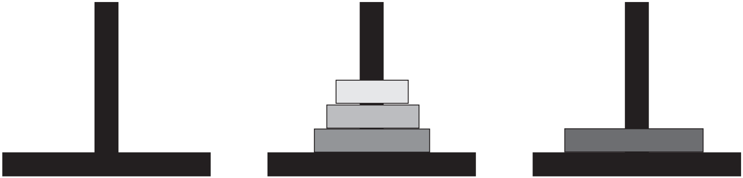
1 2 1 3 1 2

Premiki 4 diskov



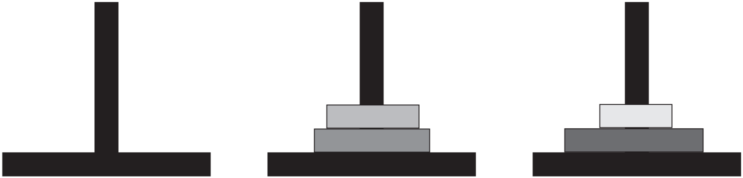
1 2 1 3 1 2 1

Premiki 4 diskov



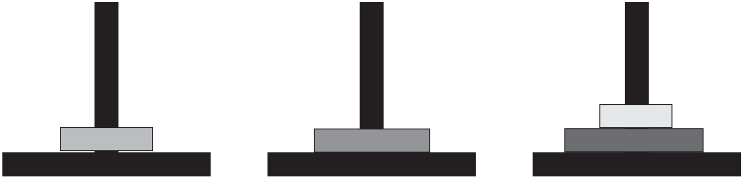
1 2 1 3 1 2 1 4

Premiki 4 diskov



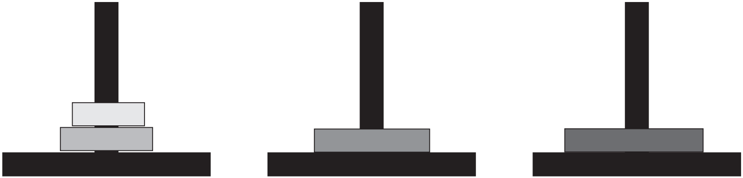
1 2 1 3 1 2 1 4 1

Premiki 4 diskov



1 2 1 3 1 2 1 4 1 2

Premiki 4 diskov

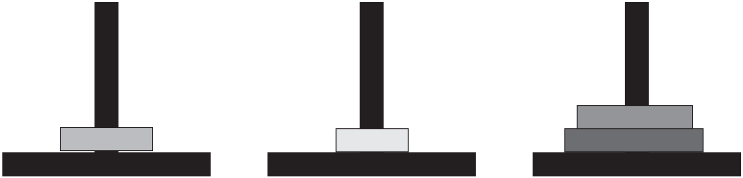


1 2 1 3 1 2 1 4 1 2 1

Premiki 4 diskov

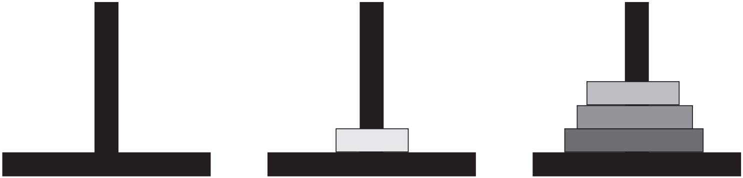
1 2 1 3 1 2 1 4 1 2 1 3

Premiki 4 diskov



1 2 1 3 1 2 1 4 1 2 1 3 1

Premiki 4 diskov



1 2 1 3 1 2 1 4 1 2 1 3 1 2

Premiki 4 diskov

1 2 1 3 1 2 1 4 1 2 1 3 1 2 1

Premiki 4 diskov - ponovno

Zakodirajmo premike takole:

- Premik z nosilca 1 na nosilec 2: a

Premiki 4 diskov - ponovno

Zakodirajmo premike takole:

- Premik z nosilca 1 na nosilec 2: a
- Premik z nosilca 2 na nosilec 3: b

Premiki 4 diskov - ponovno

Zakodirajmo premike takole:

- Premik z nosilca 1 na nosilec 2: a
- Premik z nosilca 2 na nosilec 3: b
- Premik z nosilca 3 na nosilec 1: c

Premiki 4 diskov - ponovno

Zakodirajmo premike takole:

- Premik z nosilca 1 na nosilec 2: a
- Premik z nosilca 2 na nosilec 3: b
- Premik z nosilca 3 na nosilec 1: c
- Premik z nosilca 2 na nosilec 1: $\bar{a}$

Premiki 4 diskov - ponovno

Zakodirajmo premike takole:

- Premik z nosilca 1 na nosilec 2: a
- Premik z nosilca 2 na nosilec 3: b
- Premik z nosilca 3 na nosilec 1: c
- Premik z nosilca 2 na nosilec 1: $\bar{a}$
- Premik z nosilca 3 na nosilec 2: $\bar{b}$

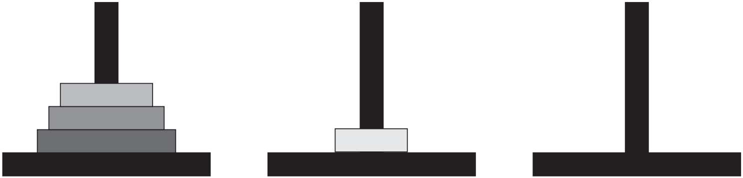
Premiki 4 diskov - ponovno

Zakodirajmo premike takole:

- Premik z nosilca 1 na nosilec 2: a
- Premik z nosilca 2 na nosilec 3: b
- Premik z nosilca 3 na nosilec 1: c
- Premik z nosilca 2 na nosilec 1: $\bar{a}$
- Premik z nosilca 3 na nosilec 2: $\bar{b}$
- Premik z nosilca 1 na nosilec 3: $\bar{c}$

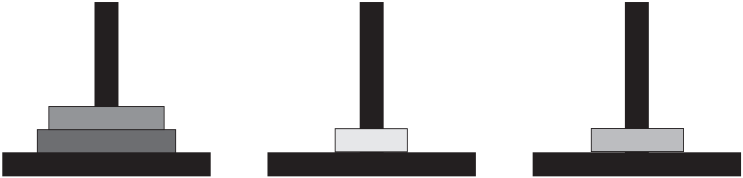
Premiki 4 diskov - ponovno

Premiki 4 diskov - ponovno



a

Premiki 4 diskov - ponovno

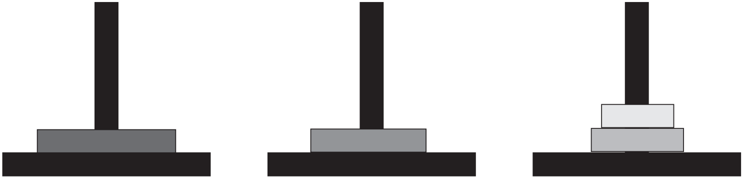


$$a \ \bar{c}$$

Premiki 4 diskov - ponovno

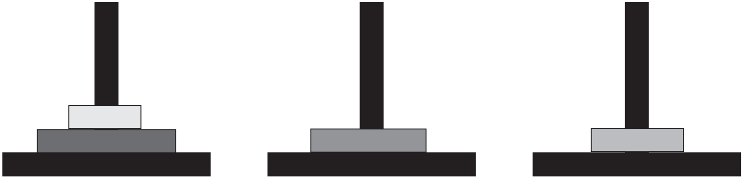
$a\ \bar{c}\ b$

Premiki 4 diskov - ponovno



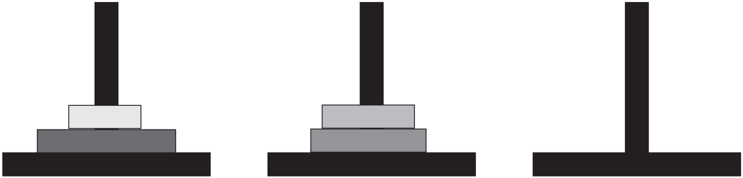
$$a\ \bar{c}\ b\ a$$

Premiki 4 diskov - ponovno



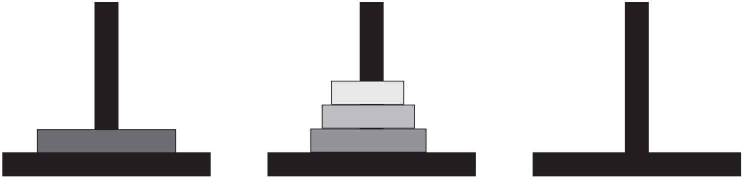
$a\ \bar{c}\ b\ a\ c$

Premiki 4 diskov - ponovno



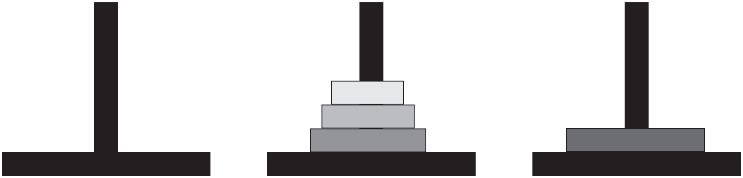
$$a\ \bar{c}\ b\ a\ c\ \bar{b}$$

Premiki 4 diskov - ponovno



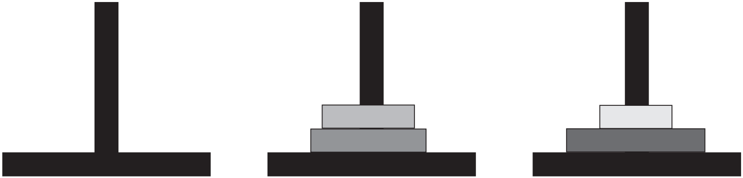
$$a\ \bar{c}\ b\ a\ c\ \bar{b}\ a$$

Premiki 4 diskov - ponovno



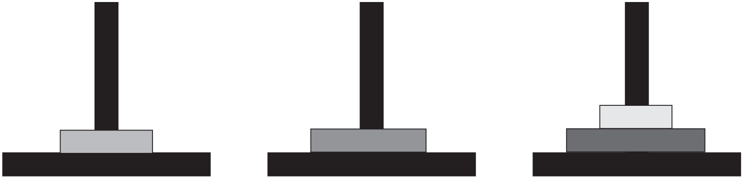
$$a\ \bar{c}\ b\ a\ c\ \bar{b}\ a\ \bar{c}$$

Premiki 4 diskov - ponovno



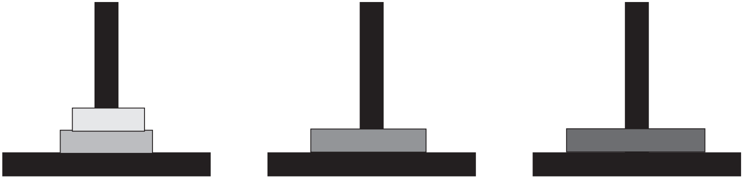
$$a\ \bar{c}\ b\ a\ c\ \bar{b}\ a\ \bar{c}\ b$$

Premiki 4 diskov - ponovno



$$a\ \bar{c}\ b\ a\ c\ \bar{b}\ a\ \bar{c}\ b\ \bar{a}$$

Premiki 4 diskov - ponovno

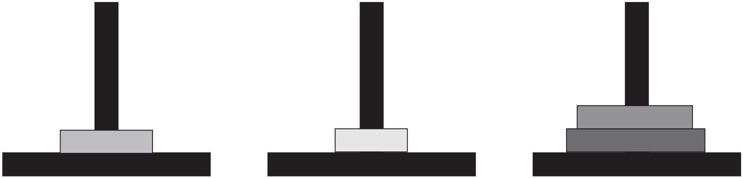


$$a\ \bar{c}\ b\ a\ c\ \bar{b}\ a\ \bar{c}\ b\ \bar{a}\ c$$

Premiki 4 diskov - ponovno

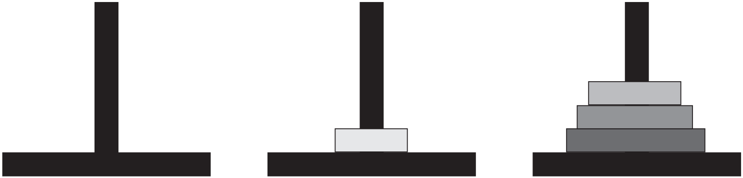
$$a\ \bar{c}\ b\ a\ c\ \bar{b}\ a\ \bar{c}\ b\ \bar{a}\ c\ b$$

Premiki 4 diskov - ponovno



$$a\ \bar{c}\ b\ a\ c\ \bar{b}\ a\ \bar{c}\ b\ \bar{a}\ c\ b\ a$$

Premiki 4 diskov - ponovno



$$a\ \bar{c}\ b\ a\ c\ \bar{b}\ a\ \bar{c}\ b\ \bar{a}\ c\ b\ a\ \bar{c}$$

Premiki 4 diskov - ponovno

$$a\,\bar{c}\,b\,a\,c\,\bar{b}\,a\,\bar{c}\,b\,\bar{a}\,c\,b\,a\,\bar{c}\,b$$

6 simbolov zadošča

6 simbolov zadošča

Izrek (Allouche, Astoorian, Randall, Shallit, 1994)

ToH zaporedje

$$a, \bar{c}, b, a, c, \bar{b}, a, \bar{c}, b, \bar{a}, c, b, a, \bar{c}, b, \ldots$$

je brez ponavljanj.

Imamo pa še več

Opazujmo

$$(a, \bar{c}, b), (a, c, \bar{b}), (a, \bar{c}, b), (\bar{a}, c, b), (a, \bar{c}, b), \ldots$$

Imamo pa še več

Opazujmo

$$(a, \bar{c}, b), (a, c, \bar{b}), (a, \bar{c}, b), (\bar{a}, c, b), (a, \bar{c}, b), \ldots$$

Obstaja samo 5 tipov takih trojic:

$$(a, \bar{c}, b) \quad (a, c, \bar{b}) \quad (\bar{a}, c, b) \quad (a, c, b) \quad (\bar{a}, c, \bar{b})\,.$$

Imamo pa še več

Opazujmo

$$(a, \bar{c}, b), (a, c, \bar{b}), (a, \bar{c}, b), (\bar{a}, c, b), (a, \bar{c}, b), \ldots$$

Obstaja samo 5 tipov takih trojic:

$$(a, \bar{c}, b) \quad (a, c, \bar{b}) \quad (\bar{a}, c, b) \quad (a, c, b) \quad (\bar{a}, c, \bar{b})\,.$$

Odtod dobimo:

Izrek (Hinz, 1996)

ToH porodi zaporedje brez ponavljanj s petimi simboli.

Hvala za pozornost!