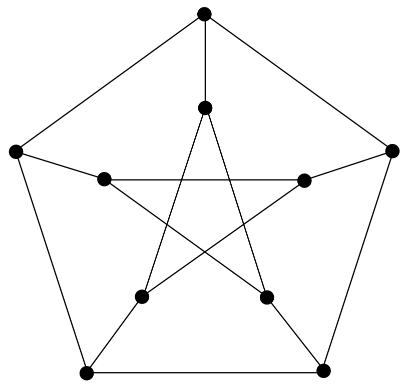
Laskuharjoitus 1

17.1.2002

1. Muodosta allaolevalle *Petersenin graafille* naapuruus- ja insidenssimatriisit.



2. Leksikografinen järjestys k-osajoukoille.

   (a) Määritä leksikografisessa järjestyksessä 100. joukon $\{1, 2, \ldots, 10\}$ 3-osajoukko.

   (b) Määritä osajoukon $\{3, 5, 7\}$ leksikografinen rank joukon $\{1, 2, \ldots, 10\}$ 3-osajoukkojen joukossa.

3. Minimimuutosjärjestys osajoukoille.

   (a) Määritä koodisanan 01010110 rank peilatussa Gray-koodissa (binary reflected Gray code) G^8.

   (b) Määritä järjestyksessä 100. peilatun Gray-koodin G^8 koodisana.

4. Määrittele leksikografiset (= "puhelinluettelojärjestys") rank- ja unrank-funktiot muotoa

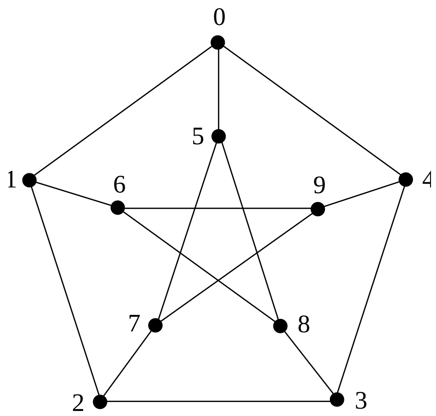
$$\mathrm{X}_1\mathrm{X}_2\mathrm{X}_3 \text{-} \mathrm{Y}_1\mathrm{Y}_2\mathrm{Y}_3$$

   oleville auton rekisterikilville, missä $\mathrm{X}_1, \mathrm{X}_2, \mathrm{X}_3 \in \{\mathrm{A}, \mathrm{B}, \mathrm{C}, \mathrm{D}, \ldots, \mathrm{Z}\}$ ja $\mathrm{Y}_1, \mathrm{Y}_2, \mathrm{Y}_3 \in \{0, 1, 2, \ldots, 9\}$. Mikä on rekisterikilven IOI-010 rank?

Laskuharjoituksen 1 vastaukset

Tehtävä 1

Nimetään aluksi Petersenin graafin solmut kuten alla.



Tällöin naapuruusmatriisiksi saadaan

	0	1	2	3	4	5	6	7	8	9
0	0	1	0	0	1	1	0	0	0	0
1	1	0	1	0	0	0	1	0	0	0
2	0	1	0	1	0	0	0	1	0	0
3	0	0	1	0	1	0	0	0	1	0
4	1	0	0	1	0	0	0	0	0	1
5	1	0	0	0	0	0	0	1	1	0
6	0	1	0	0	0	0	0	0	1	1
7	0	0	1	0	0	1	0	0	0	1
8	0	0	0	1	0	1	1	0	0	0
9	0	0	0	0	1	0	1	1	0	0

(Naapuruusmatriisissa solmua i vastaavalla rivillä on solmua j vastaavassa sarakkeessa alkio 1 jos solmujen i ja j välillä on kaari graafissa; muussa tapauksessa kyseinen alkio on 0.)

Vastaavasti insidenssimatriisiksi saadaan

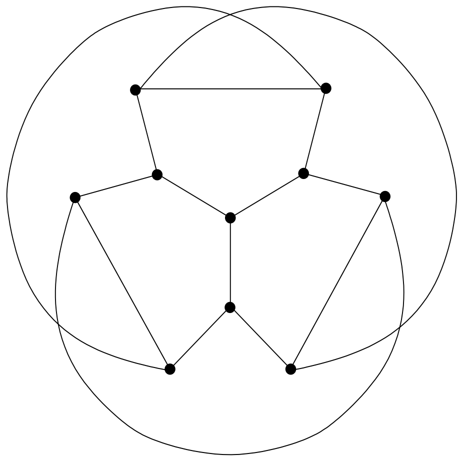
	01	12	23	34	40	05	16	27	38	49	57	58	68	69	79
0	1	0	0	0	1	1	0	0	0	0	0	0	0	0	0
1	1	1	0	0	0	0	1	0	0	0	0	0	0	0	0
2	0	1	1	0	0	0	0	1	0	0	0	0	0	0	0
3	0	0	1	1	0	0	0	0	1	0	0	0	0	0	0
4	0	0	0	1	1	0	0	0	0	1	0	0	0	0	0
5	0	0	0	0	0	1	0	0	0	0	1	1	0	0	0
6	0	0	0	0	0	0	1	0	0	0	0	0	1	1	0
7	0	0	0	0	0	0	0	1	0	0	1	0	0	0	1
8	0	0	0	0	0	0	0	0	1	0	0	1	1	0	0
9	0	0	0	0	0	0	0	0	0	1	0	0	0	1	1

.

(Insidenssimatriisissa solmua i vastaavalla rivillä on kaarta $\{j,k\}$ vastaavassa sarakkeessa alkio 1 jos $i \in \{j,k\}$; muussa tapauksessa kyseinen alkio on 0.)

Huomaa, että sekä naapuruus- että insidenssimatriisi riippuvat solmujen nimeämistavasta.

Lisätehtävä. Alla on toinen esitystapa Petersenin graafille. Etsi tapa nimetä graafin solmut siten, että tulokseksi saadaan sama naapuruus/insidenssimatriisi kuin edellä. (Ratkaisuja on useampia kuin yksi.)



Tehtävä 2 (a)

Käytetään hyväksi k-osajoukkojen koleksikografisen ja leksikografisen järjestyksen välistä yhteyttä (kirjan lause 2.4): Olkoon $T = \{t_1, t_2, \ldots, t_k\} \subseteq \{1, 2, \ldots, n\}$ tutkittava k-osajoukko ja $T' \stackrel{\text{def}}{=} \{n+1-i \;:\; i \in T\}$. Tällöin pätee

$$\mathsf{rank}_{\text{lex}}(T) + \mathsf{rank}_{\text{colex}}(T') = \binom{n}{k} - 1.$$

Tehtävässä on annettu $n = 10$, $k = 3$ ja $\mathsf{rank}(T) = 99$, joten

$$\mathsf{rank}_{\mathrm{colex}}(T') = \binom{10}{3} - 1 - 99 = 120 - 1 - 99 = 20.$$

Käytetään kirjan algoritmia 2.10 laskemaan $T' = \{t'_1, t'_2, t'_3\} = \mathsf{unrank}_{\mathrm{colex}}(20)$ kun $n = 10$ ja $k = 3$. Tulokseksi saadaan

$$20 = 20 + 0 + 0 = \binom{6}{3} + \binom{1}{2} + \binom{0}{1},$$

joten

$$t'_1 = 6 + 1 = 7, \qquad t'_2 = 1 + 1 = 2, \qquad t'_3 = 0 + 1 = 1.$$

Nyt siis $T' = \{7, 2, 1\}$, joten $T = \{4, 9, 10\}$.

Tehtävä 2 (b)

Ratkaistaan tehtävä vastaavalla tempulla kuten edellä: Nyt $T = \{3, 5, 7\}$, joten $T' = \{8, 6, 4\}$. Koleksikografisen rank-funktion määritelmästä

$$\mathsf{rank}_{\mathrm{colex}}(T') \stackrel{\mathrm{def}}{=} \sum_{i=1}^{k} \binom{t_i - 1}{k + 1 - i}$$

saadaan tapauksessa $t'_1 = 8$, $t'_2 = 6$, $t'_3 = 4$

$$\mathsf{rank}_{\mathrm{colex}}(\{8, 6, 4\}) = \binom{7}{3} + \binom{5}{2} + \binom{3}{1} = 35 + 10 + 3 = 48,$$

joten

$$\mathsf{rank}_{\mathrm{lex}}(\{3, 5, 7\}) = \binom{10}{3} - 1 - \mathsf{rank}_{\mathrm{colex}}(\{8, 6, 4\}) = 120 - 1 - 48 = 71.$$

Tehtävä 3 (a)

Johdetaan aluksi rekursiivinen versio rank-funktiosta peilatulle Gray-koodille G^n. (Ei-rekursiivinen versio on kirjan algoritmi 2.4.) Perustapauksessa $n = 1$ pätee selvästi

$$\mathsf{rank}(0) = 0, \qquad \mathsf{rank}(1) = 1.$$

Tapauksissa $n \geq 2$ pätee koodin G^n rekursiivisen määritelmän perusteella

$$\mathsf{rank}(b_n b_{n-1} \cdots b_1) = \begin{cases} \mathsf{rank}(b_{n-1} \cdots b_1) & \text{jos } b_n = 0; \\ 2^n - 1 - \mathsf{rank}(b_{n-1} \cdots b_1) & \text{jos } b_n = 1. \end{cases}$$

Koodisanan 01010110 rank saadaan nyt rekursion avulla:

$$\begin{aligned} \mathsf{rank}(01010110) &= \mathsf{rank}(1010110) \\ &= 127 - \mathsf{rank}(010110) \\ &= 127 - \mathsf{rank}(10110) \\ &= 127 - (31 - \mathsf{rank}(0110)) \\ &= 127 - (31 - \mathsf{rank}(110)) \\ &= 127 - (31 - (7 - \mathsf{rank}(10))) \\ &= 127 - (31 - (7 - (3 - \mathsf{rank}(0)))) \\ &= 127 - (31 - (7 - (3 - 0))) \\ &= 100. \end{aligned}$$

Tehtävä 3 (b)

Olkoon koodiin G^n kuuluvan koodisanan $b_n b_{n-1} \cdots b_1$ rank r_n. Nyt unrank voidaan tehdä tarkastelemalla rekursiivisesti kumpaan G^n:n puoliskoon (0-alkuiset koodisanat / 1-alkuiset koodisanat) koodisana kuuluu:

$$b_n = \begin{cases} 0 & \text{jos } r_n < 2^{n-1}; \\ 1 & \text{jos } r_n \geq 2^{n-1}, \end{cases} \qquad r_{n-1} = \begin{cases} r_n & \text{jos } r_n < 2^{n-1}; \\ 2^n - 1 - r_n & \text{jos } r_n \geq 2^{n-1}. \end{cases}$$

Tehtävässä on annettu $r_8 = 99$. Edellistä soveltamalla saadaan

n	r_n	2^{n-1}	b_n
8	99	128	0
7	99	64	1
6	28	32	0
5	28	16	1
4	3	8	0
3	3	4	0
2	3	2	1
1	0	1	0

.

Joten $\mathsf{unrank}(99) = 01010010$. (Vaihtoehtoisesti tehtävä voidaan ratkaista kirjan algoritmilla 2.5.)

Tehtävä 4

Samaistetaan aluksi aakkosten kirjaimet A, B, C, ... , Z lukujen 0, 1, ... , 25 kanssa:

$$\mathrm{A} \triangleq 0, \quad \mathrm{B} \triangleq 1, \quad \mathrm{C} \triangleq 2, \quad \cdots \quad \mathrm{Y} \triangleq 24, \quad \mathrm{Z} \triangleq 25.$$

Tarkastellaan rekisterikilpiä muotoa $[z_1, z_2, z_3, z_4, z_5, z_6]$ olevina listoina, missä $z_1, z_2, z_3 \in \{0, 1, \dots, 25\}$ ja $z_4, z_5, z_6 \in \{0, 1, \dots, 10\}$. Tällöin esimerkiksi kilpeä IOI-010 vastaa lista $[8, 14, 8, 0, 1, 0]$.

Tavanomainen leksikografinen järjestys rekisterikilpien välillä määräytyy nyt ehdosta $[z'_1, z'_2, z'_3, z'_4, z'_5, z'_6] < [z_1, z_2, z_3, z_4, z_5, z_6]$ jos ja vain jos on olemassa $j \in \{1, 2, 3, 4, 5, 6\}$ s.e. $z'_j < z_j$ ja kaikilla $i \in \{1, \dots, j-1\}$ pätee $z'_i = z_i$.

Rakennetaan aluksi rank-funktio. Kilven $[z_1, z_2, z_3, z_4, z_5, z_6]$ leksikografinen rank on niiden kilpien $[z'_1, z'_2, z'_3, z'_4, z'_5, z'_6]$ lukumäärä, joille pätee $[z'_1, z'_2, z'_3, z'_4, z'_5, z'_6] < [z_1, z_2, z_3, z_4, z_5, z_6]$. Tällaiset kilvet voidaan jakaa alitapauksiin j:n arvon (ks. leksikografisen järjestyksen määritelmä yllä) perusteella. Erityisesti z'_j voidaan valita z_j tavalla siten, että $z'_j < z_j$. Tämän jälkeen arvot z'_k, missä $k > j$, voidaan valita mielivaltaisesti. Näin ollen saadaan

$$\begin{aligned}\mathsf{rank}([z_1, z_2, z_3, z_4, z_5, z_6]) = {} & z_1 \cdot 26 \cdot 26 \cdot 10 \cdot 10 \cdot 10 + \\ & + z_2 \cdot 26 \cdot 10 \cdot 10 \cdot 10 + \\ & + z_3 \cdot 10 \cdot 10 \cdot 10 + \\ & + z_4 \cdot 10 \cdot 10 + \\ & + z_5 \cdot 10 + \\ & + z_6.\end{aligned}$$

Eli lyhyemmin

$$\mathsf{rank}([z_1, z_2, z_3, z_4, z_5, z_6]) = \sum_{j=1}^{6} \big(z_j \prod_{k=j+1}^{6} B_k\big)$$

jos asetetaan $B_1 = B_2 = B_3 = 26$ ja $B_4 = B_5 = B_6 = 10$. (Tapauksessa $j = 6$ määritellään lisäksi $\prod_{k=j+1}^{6} B_k = 1$.)

Kilvelle IOI-010 saadaan nyt

$$\begin{aligned}
\mathsf{rank}([8,14,8,1,0,1]) &= 8 \cdot 26 \cdot 26 \cdot 10 \cdot 10 \cdot 10 + \\
&+ 14 \cdot 26 \cdot 10 \cdot 10 \cdot 10 + \\
&+ 8 \cdot 10 \cdot 10 \cdot 10 + \\
&+ 0 \cdot 10 \cdot 10 + \\
&+ 1 \cdot 10 + \\
&+ 0 \\
&= 5780010.
\end{aligned}$$

Unrank voidaan tehdä seuraavalla algoritmilla:

1. Asetetaan $n_6 \stackrel{\text{def}}{=} \mathsf{rank}([z_1, z_2, z_3, z_4, z_5, z_6])$.
2. Toistetaan kaikille $i = 6, 5, \ldots, 1$.
   (a) Määritetään jakojäännös $z_i = n_i \mod B_i$.
   (b) Asetetaan $n_{i-1} = (n_i - z_i)/B_i$.
3. Palautetaan tulos $\mathsf{unrank}(n_6) = [z_1, z_2, z_3, z_4, z_5, z_6]$.

Esimerkiksi

i	n_i	B_i	z_i
6	5780010	10	0
5	578001	10	1
4	57800	10	0
3	5780	26	8
2	222	26	14
1	8	26	8

,

joten $\mathsf{unrank}(5780010) = [8, 14, 8, 0, 1, 0]$ kuten pitääkin.

Kotitehtävä 1

n-permutaatioiden Gray-koodi on joukon $\{1, 2, \ldots, n\}$ alkioiden permutaatioiden lista $[\pi_0, \pi_1, \ldots, \pi_{n!-1}]$, joka täyttää seuraavat ehdot:

(a) kukin joukon $\{1, 2, \ldots, n\}$ permutaatio esiintyy koodissa täsmälleen kerran; ja

(b) $\pi_0 = [1, 2, \ldots, n]$; ja

(c) kullakin $0 \leq i < n!$ permutaation $\pi_{(i+1) \bmod n!}$ listaesitys saadaan permutaation π_i listaesityksestä kahden vierekkäisen alkion paikkaa vaihtamalla.

Koska permutaation listaesityksen positioissa δ_i ja $\delta_i + 1$ olevien alkioiden vaihto voidaan esittää permutaation kertomisena oikealta transpositiolla $(\delta_i,\ \delta_i{+}1)$, ehto (c) voidaan esittää seuraavasti: kaikilla $0 \leq i < n!$

$$\pi_{(i+1) \bmod n!} = \pi_i(\delta_i,\ \delta_i{+}1),$$

missä $1 \leq \delta_i < n$. Esimerkiksi $[2, 4, 1, 3] = [2, 1, 4, 3](2, 3)$.

1. Määritä 4-permutaatioiden Gray-koodien lukumäärä.

2. Sovelletaan permutaatioihin leksikografista järjestystä. Mikä on leksikografisessa järjestyksessä ensimmäinen koodi?

3. Määritä 4-permutaatioiden Gray-koodien lukumäärä, kun edellisestä määritelmästä poiketen *ei* vaadita, että π_0:n listaesitys voitaisiin saada $\pi_{n!-1}$:n listaesityksestä kahden vierekkäisen alkion paikkaa vaihtamalla.

Bonustehtävä. Gray-koodin *deltajono* on lista $[\delta_0, \delta_1, \ldots, \delta_{n!-1}]$. Kahta koodia pidetään ekvivalentteina, jos toisen deltajono saadaan toisesta syklisellä siirrolla $[\delta_k, \ldots, \delta_{n!-1}, \delta_0, \ldots, \delta_k]$, kääntämällä $[\delta_{n!-1}, \ldots, \delta_1, \delta_0]$, peilaamalla $[n - \delta_0, n - \delta_1, \ldots, n - \delta_{n!-1}]$ tai näitä operaatioita yhdistelemällä. Kuinka moneen ekvivalenssiluokkaan 4-permutaatioiden Gray-koodit jakautuvat, ja montako koodia kussakin luokassa on?

Palautus. Kotitehtävä on palautettava sähköpostitse assistentille (Petteri Kaski, `Petteri.Kaski@hut.fi`) viimeistään torstaina 28.2.2002. Palautettavassa selostuksessa tulee vastausten lisäksi kuvata käytetty ratkaisutapa. Selostuksen voi palauttaa esimerkiksi hyvälaatuisena PostScriptinä. Selostuksen lisäksi palautuksessa on oltava liitteenä ongelman ratkaisemiseen mahdollisesti käytetyn ohjelman lähdekoodi.

Laskuharjoitus 2

24.1.2002

1. Permutaation sykli- ja listaesitykset.

   (a) Esitä permutaatio $\pi_1 = [2, 4, 6, 7, 5, 3, 1]$ pistevieraiden syklien tulona.

   (b) Esitä permutaatio $\pi_2 = (1, 5, 6)(2, 4, 3)(7)$ listana.

   (c) Määritä tulo $\pi_1\pi_2$ ja käänteispermutaatiot π_1^{-1}, π_2^{-1}.

2. Oletetaan, että $p, q, r_1, \dots, r_k, s_1, \dots, s_l$, missä $k, l \geq 0$, ovat toisistaan poikkeavia alkioita. Sievennä seuraavat syklien tuloina annetut permutaatiot pistevieraiden syklien tuloiksi:

   (a) $(p, q)(p, r_1, \dots, r_k, q, s_1, \dots, s_l)$

   (b) $(p, q)(p, r_1, \dots, r_k)(q, s_1, \dots, s_l)$

3. Permutaatio π on *parillinen* jos se voidaan esittää parillisen määrän transpositioita tulona. Vastaavasti π on *pariton* jos se voidaan esittää parittoman määrän transpositioita tulona. Jokainen permutaatio on joko pariton tai parillinen.

   (a) Olkoon $r \geq 1$. Esitä r-sykli $(1, 2, \dots, r)$ transpositioiden tulona.

   (b) Edellisen perusteella esitä permutaatio $[2, 4, 6, 7, 5, 3, 1]$ transpositioiden tulona. Onko permutaatio parillinen vai pariton?

4. Kirjan tehtävä 2.11: Määritä permutaation $[2, 4, 6, 7, 5, 3, 1]$ rank ja seuraaja sekä leksikografisessa että Trotter-Johnson -järjestyksessä.

Laskuharjoituksen 2 vastaukset

Kerrataan aluksi muutamia permutaatioihin liittyviä määritelmiä. Olkoon X äärellinen ei-tyhjä joukko.

- Joukon X permutaatio σ on *r-sykli* jos on olemassa alkiot $a_1, \ldots, a_r \in X$ siten, että

$$\sigma(a_1) = a_2, \quad \sigma(a_2) = a_3, \quad \ldots, \sigma(a_{r-1}) = a_r, \quad \sigma(a_r) = a_1,$$

  ja $\sigma(a) = a$ kaikilla $a \in X \setminus \{a_1, \ldots, a_r\}$. Edellinen r-sykli on tapana kirjoittaa $(a_1, a_2, \ldots, a_r)$. Sama sykli voidaan tällä tavoin esittää useammalla tavalla, koska syklin alkupiste on mielivaltainen; esim. $(a_2, a_3, \ldots, a_r, a_1) = (a_1, a_2, \ldots, a_r)$.

- *Transpositio* on mielivaltainen joukon X 2-sykli.

- Joukon X permutaatiot σ_1, σ_2 ovat *pistevieraat* jos millään $x \in X$ ei päde sekä $\sigma_1(x) \neq x$ että $\sigma_2(x) \neq x$.

- Joukon X permutaatioiden π_1, π_2 *tulo* $\pi_1\pi_2$ määritellään kaikille $x \in X$ säännöstä $\pi_1\pi_2(x) = \pi_1 \circ \pi_2(x) = \pi_1(\pi_2(x))$.

- Jokainen joukon X permutaatio voidaan esittää (syklien järjestystä vaille) yksikäsitteisesti pistevieraiden syklien tulona, jossa jokainen $x \in X$ esiintyy täsmälleen yhdellä syklillä.

Tehtävä 1 (a)

Permutaation π_1 listaesityksestä $[2, 4, 6, 7, 5, 3, 1]$ nähdään, että

$$\pi_1(1) = 2, \quad \pi_1(2) = 4, \quad \pi_1(4) = 7, \quad \pi_1(7) = 1,$$

joten alkion 1 sisältävä sykli on $(1, 2, 4, 7)$. Pienin joukon $\{1, 2, 3, 4, 5, 6, 7\}$ alkio joka ei ole tällä syklillä on 3. Alkion 3 sisältävä sykli on $(3, 6)$, koska

$$\pi_1(3) = 6, \quad \pi_1(6) = 3.$$

Pienin alkio joka ei ole edellisillä sykleillä on 5, jolle pätee $\pi_1(5) = 5$. Nyt kaikki joukon $\{1, 2, 3, 4, 5, 6, 7\}$ alkioiden syklit on käyty läpi, joten saadaan

$$\pi_1 = (1, 2, 4, 7)(3, 6)(5).$$

Tehtävä 1 (b)

Sykliesityksestä $\pi_2 = (1,5,6)(2,4,3)(7)$ nähdään

$$\begin{aligned} &\pi_2(1) = 5, \quad \pi_2(2) = 4, \quad \pi_2(3) = 2, \quad \pi_2(4) = 3, \\ &\pi_2(5) = 6, \quad \pi_2(6) = 1, \quad \pi_2(7) = 7, \end{aligned}$$

joten $\pi_2 = [5,4,2,3,6,1,7]$.

Tehtävä 1 (c)

Tulo $\pi_1\pi_2$ on sykliesityksien avulla

$$\pi_1\pi_2 = (1,2,4,7)(3,6)(5)(1,5,6)(2,4,3)(7),$$

joka voidaan sieventää muotoon

$$\pi_1\pi_2 = (1,5,3,4,6,2,7)$$

määrittämällä jokaisen alkion kuva. Määritettäessä alkion kuvaa syklien tulossa syklejä luetaan järjestyksessä *oikealta vasemmalle*. Esim. edellisessä tulossa sykli (7) pitää alkion 1 paikallaan, kuten myös (2,4,3). Sykli $(1,5,6)$ kuvaa alkion 1 alkiolle 5. Sykli (5) pitää alkion 5 paikallaan, kuten myös syklit $(3,6)$ ja $(1,2,4,7)$. Näin ollen $\pi_1\pi_2(1) = 5$.

Permutaation käänteispermutaatio muodostetaan helposti sen sykliesityksestä kääntämällä sekä syklien, että syklien alkioiden järjestys päinvastaiseksi. Esimerkiksi

$$\pi_1^{-1} = (5)(6,3)(7,4,2,1), \qquad \pi_2^{-1} = (7)(3,4,2)(6,5,1).$$

Tehtävä 2

Tehtävän tulot ovat sievennetyssä muodossa

$$(p,q)(p,r_1,\ldots,r_k,q,s_1,\ldots,s_l) = (p,r_1,\ldots,r_k)(q,s_1,\ldots,s_l)$$

ja

$$(p,q)(p,r_1,\ldots,r_k)(q,s_1,\ldots,s_l) = (p,r_1,\ldots,r_k,q,s_1,\ldots,s_l).$$

Näin ollen transpositiolla vasemmalta kertominen joko "leikkaa" syklin kahtia, tai "yhdistää" kaksi pistevierasta sykliä.

Edellisestä huomataan että jos mielivaltainen permutaatio π kerrotaan vasemmalta transpositiolla (p,q), niin tällöin permutaatiossa $(p,q)\pi$ on joko yksi sykli enemmän (a-kohta) kuin permutaatiossa π, tai yksi sykli vähemmän (b-kohta).

Tehtävä 3

Tehtävän (a)-kohdan r-sykli voidaan esittää monella tavalla transpositioiden tulona. Esimerkiksi

$$(1, 2, \dots, r) = (1, 2)(2, 3)(4, 5) \cdots (r-2, r-1)(r-1, r)$$

tai

$$(1, 2, \dots, r) = (1, r)(1, r-1)(1, r-2) \cdots (1, 3)(1, 2)$$

ovat molemmat käypiä esitystapoja, joihin tarvitaan $r - 1$ transpositiota.

Tehtävän (b)-kohdassa transpositioesitys voidaan muodostaa esimerkiksi määrittämällä ensin permutaatiolle $[2, 4, 6, 7, 5, 3, 1]$ esitys pistevieraiden syklien avulla (ks. tehtävä 1a) ja tämän jälkeen sovelletaan jokaiselle syklille (a)-kohtaa. Tällöin saadaan

$$\begin{aligned}[2, 4, 6, 7, 5, 3, 1] &= (1, 2, 4, 7)(3, 6)(5) \\ &= (1, 2)(2, 4)(4, 7)(3, 6).\end{aligned}$$

Permutaatio $[2, 4, 6, 7, 5, 3, 1]$ on siis parillinen.

Yleisesti jos joukon X permutaation π esityksessä pistevieraiden syklien avulla on c sykliä, joiden pituudet ovat $r_1, \dots, r_c$, saadaan, että permutaatio voidaan esittää $N_\pi = \sum_{i=1}^{c}(r_i - 1)$ transposition avulla. Koska $\sum_{i=1}^{c} r_i = |X|$, voidaan edellinen sieventää muotoon $N_\pi = |X| - c$. Permutaatio π on nyt parillinen (vastaavasti pariton) täsmälleen jos N_π on parillinen (pariton).

Tehtävä 4

Tarkastellaan ensin leksikografista järjestystä. Permutaation

$$\pi = [\pi(1), \pi(2), \dots, \pi(n)]$$

leksikografinen seuraaja määritetään seuraavasti:

1. Etsitään suurin $j \in \{1, \dots, n-1\}$, jolle pätee $\pi(j) < \pi(j+1)$. Jos tällaista ei ole olemassa, kyseessä on leksikografisessa järjestyksessä viimeinen permutaatio $[n, n-1, \dots, 1]$, jolla ei ole seuraajaa.

   Permutaatiolle $[2, 4, 6, 7, 5, 3, 1]$ on $j = 3$.

2. Etsitään alkioista $\pi(j+1), \dots, \pi(n)$ pienin alkio, joka on suurempi kuin $\pi(j)$.

   Permutaatiolle $[2, 4, 6, 7, 5, 3, 1]$ tämä alkio on 7.

3. Vaihdetaan edellisessä kohdassa valittu alkio ja $\pi(j)$ keskenään. Tulokseksi saadaan permutaatio π'.

   Permutaation $[2,4,6,7,5,3,1]$ tapauksessa pätee $\pi' = [2,4,7,6,5,3,1]$.

4. Järjestetään alkiot $\pi'(j+1),\ldots,\pi'(n)$ kasvavaan järjestykseen (koska alkiot ovat valmiiksi laskevassa järjestyksessä, riittää kääntää niiden järjestys). Tuloksena saadaan π:n leksikografinen seuraajapermutaatio.

   Permutaation $[2,4,6,7,5,3,1]$ tapauksessa seuraaja on $[2,4,7,1,3,5,6]$.

Permutaation $\pi = [\pi(1),\pi(2),\ldots,\pi(n)]$ leksikografinen rank määritetään rekursiivisesti:

1. Perustapaus $n = 1$: $\mathsf{rank}(\pi) = 0$.
2. Yleinen tapaus $n \geq 2$:

$$\mathsf{rank}(\pi) = (\pi(1)-1)(n-1)! + \mathsf{rank}(\pi'),$$

   missä

$$\pi'(j) = \begin{cases} \pi(j+1) & \text{jos } \pi(j+1) < \pi(1); \\ \pi(j+1)-1 & \text{jos } \pi(j+1) > \pi(1). \end{cases}$$

Permutaation $[2,4,6,7,5,3,1]$ rank on edellisen perusteella

$$\begin{aligned}
&\mathsf{rank}([2,4,6,7,5,3,1]) = \\
&= (2-1)6! + \mathsf{rank}([3,5,6,4,2,1]) \\
&= (2-1)6! + (3-1)5! + \mathsf{rank}([4,5,3,2,1]) \\
&= (2-1)6! + (3-1)5! + (4-1)4! + \mathsf{rank}([4,3,2,1]) \\
&= (2-1)6! + (3-1)5! + (4-1)4! + (4-1)3! + \mathsf{rank}([3,2,1]) \\
&= (2-1)6! + (3-1)5! + (4-1)4! + (4-1)3! + (3-1)2! + \mathsf{rank}([2,1]) \\
&= (2-1)6! + (3-1)5! + (4-1)4! + (4-1)3! + (3-1)2! + (2-1)1! \\
&= 1\cdot 6! + 2\cdot 5! + 3\cdot 4! + 3\cdot 3! + 2\cdot 2! + 1\cdot 1! \\
&= 1055.
\end{aligned}$$

Siirrytään nyt Trotter-Johnson järjestykseen. Trotter-Johnson järjestyksessä permutaation

$$\pi = [\pi(1),\pi(2),\ldots,\pi(n)]$$

seuraaja määritetään:

1. Etsitään $j \in \{1, \dots, n\}$ siten, että $\pi(j) = n$.

   Permutaation $[2, 4, 6, 7, 5, 3, 1]$ tapauksessa $\pi(4) = 7$.

2. Määritetään, onko permutaatio $\pi' = [\pi(1), \dots, \pi(j-1), \pi(j+1), \dots, \pi(n)]$ parillinen vai pariton.

   Permutaation $[2, 4, 6, 7, 5, 3, 1]$ tapauksessa

   $$\pi' = [2, 4, 6, 5, 3, 1] = (1, 2, 4, 5, 3, 6)$$

   on pariton.

3. Jos π' on parillinen ja $j > 1$: permutaation π seuraaja saadaan vaihtamalla alkiot $\pi(j-1)$ ja $\pi(j)$ keskenään.

4. Jos π' on parillinen ja $j = 1$: permutaation π seuraaja on

   $$[n, \pi''(1), \pi''(2), \dots, \pi''(n-1)],$$

   missä π'' on permutaation π' Trotter-Johnson seuraaja.

5. Jos π' on pariton ja $j < n$: permutaation π seuraaja saadaan vaihtamalla alkiot $\pi(j)$ ja $\pi(j+1)$ keskenään.

   Permutaation $[2, 4, 6, 7, 5, 3, 1]$ tapauksessa Trotter-Johnson seuraaja on näin ollen $[2, 4, 6, 5, 7, 3, 1]$.

6. Jos π' on pariton ja $j = n$: permutaation π seuraaja on

   $$[\pi''(1), \pi''(2), \dots, \pi''(n-1), n],$$

   missä π'' on permutaation π' Trotter-Johnson seuraaja.

Permutaation π Trotter-Johnson rank lasketaan rekursiivisesti:

1. Perustapaus $n = 1$: $\mathsf{rank}(\pi) = 0$.

2. Etsitään $j \in \{1, \dots, n\}$ siten, että $\pi(j) = n$.

3. Määritetään rekursiivisesti permutaation

   $$\pi' = [\pi(1), \dots, \pi(j-1), \pi(j+1), \dots, \pi(n)]$$

   rank.

4. Jos $\mathsf{rank}(\pi')$ on parillinen, asetetaan

   $$\mathsf{rank}(\pi) = n\mathsf{rank}(\pi') + n - j.$$

5. Jos $\mathsf{rank}(\pi')$ on pariton, asetetaan

$$\mathsf{rank}(\pi) = n\mathsf{rank}(\pi') + j - 1.$$

Permutaation $[2, 4, 6, 7, 5, 3, 1]$ tapauksessa:

n	π	j	$\mathsf{rank}(\pi)$
1	$[1]$	1	0
2	$[2, 1]$	1	$2 \cdot 0 + (2 - 1) = 1$
3	$[2, 3, 1]$	2	$3 \cdot 1 + (2 - 1) = 4$
4	$[2, 4, 3, 1]$	2	$4 \cdot 4 + (4 - 2) = 18$
5	$[2, 4, 5, 3, 1]$	3	$5 \cdot 18 + (5 - 3) = 92$
6	$[2, 4, 6, 5, 3, 1]$	3	$6 \cdot 92 + (6 - 3) = 555$
7	$[2, 4, 6, 7, 5, 3, 1]$	4	$7 \cdot 555 + (4 - 1) = 3888$

Laskuharjoitus 3

31.1.2002

1. Määritä osituksen $1 + 3 + 3 + 3 + 7$ rsf-lex rank.

2. Kuinka monella tavalla voidaan 30 samanlaista palloa asettaa kuuteen samanlaiseen laatikkoon, jos vaaditaan, että missään kahdessa laatikossa ei ole samaa määrää palloja.

3. Minkä nimetyn puun Prüferin listaesitys on $[3, 1, 4, 1, 5]$?

4. Neljän termin tulo $abcd$ voidaan ryhmitellä kahden termin tuloiksi viidellä erilaisella tavalla:

$$((ab)c)d, \quad (a(bc))d, \quad a((bc)d), \quad a(b(cd)), \quad (ab)(cd).$$

   Yleisessä tapauksessa tulossa on $n + 1$ termiä. Esitä bijektio erilaisten kahden termin tuloiksi ryhmittelyjen ja Catalanin perheen C_n välille. (Vihje: jokainen ryhmittely voidaan esittää binääripuuna, jonka lehdet koostuvat tulon termeistä.)

Laskuharjoituksen 3 vastaukset

Tehtävä 1

Ositus $1 + 3 + 3 + 3 + 7$ on käänteisessä standardimuodossa (reverse standard form); standardimuoto on $7+3+3+3+1$. Rsf-lex rank on leksikografinen rank osituksien listaesityksille käänteisessä standardimuodossa. Osituksen rsf-rank voidaan laskea seuraavan rekursion (kirjan s. 76) avulla: Oletetaan, että rankattava ositus on standardimuodossa $[a_1, \ldots, a_n]$, missä $a_1 \geq \cdots \geq a_n$. Rekursion perustapauksessa $n = 1$ on aina $\mathsf{rank}([a_1]) = 0$. Tapauksessa $n > 1$ rekursioaskel on

$$\mathsf{rank}([a_1, \ldots, a_n]) = \begin{cases} \mathsf{rank}([a_1, \ldots, a_{n-1}]) & \text{jos } a_n = 1 \\ \mathsf{rank}([a'_1, \ldots, a'_n]) + P(m-1, n-1) & \text{jos } a_n > 1, \end{cases}$$

missä $a'_j = a_j - 1$ kaikilla $1 \leq j \leq n$ ja $m = \sum_{i=1}^{n} a_n$. Soveltamalla rekursioyhtälöä saadaan

$$\mathsf{rank}([7,3,3,3,1]) = \mathsf{rank}([7,3,3,3]) = \mathsf{rank}([6,2,2,2]) + P(16-1, 4-1),$$

josta edelleen

$$\mathsf{rank}([6,2,2,2]) = \mathsf{rank}([5,1,1,1]) + P(12-1, 4-1).$$

Nyt $\mathsf{rank}([5,1,1,1]) = \cdots = \mathsf{rank}([5]) = 0$, joten yhdistämällä saadaan

$$\mathsf{rank}([7,3,3,3,1]) = P(15,3) + P(11,3) = 19 + 10 = 29.$$

Tehtävä 2

Oletetaan aluksi, että pallot saa asetella laatikoihin mielivaltaisesti. Jos kuudesta laatikosta $0 \leq k \leq 5$ jätetään tyhjäksi, pallot asetellaan jäljellä oleviin $6-k$ laatikkoon. Numeroidaan ei-tyhjät laatikot $1, \ldots, 6-k$, ja merkitään a_j:llä laatikossa j olevien pallojen lukumäärää. Nyt on selvästi $a_j \geq 1$ kaikilla $j = 1, \ldots, 6-k$, ja toisaalta $\sum_{j=1}^{6-k} a_j = 30$. Näin ollen joukko $\{(a_1, \ldots, a_{6-k}) : a_j \geq 1 \text{ ja } \sum_{j=1}^{6-k} a_j = 30\}$ käsittää kaikki mahdolliset tavat asetella 30 samanlaista palloa *numeroituihin* laatikoihin. Koska laatikot ovat samanlaisia, lasketaan tällöin kuitenkin tietty osuus asetelmia useampaan kertaan; esimerkiksi asetelmat $(1,1,1,1,1,25)$ ja $(25,1,1,1,1,1)$ tapauksessa $k = 0$ ovat samanlaisia. Samanlaisten asetelmien kertautuminen voidaan välttää vaatimalla lisäksi $a_1 \leq \cdots \leq a_{6-k}$. Tällöin huomataan, että joukon

$$\mathcal{P}(30, 6-k) := \{(a_1, \ldots, a_{6-k}) : 1 \leq a_1 \leq \cdots \leq a_{6-k} \text{ ja } \sum_{j=1}^{6-k} a_j = 30\}$$

alkiot vastaavat itse asiassa luvun 30 erilaisia osituksia $6-k$ osaan, joten joukossa on $P(30, 6-k)$ erilaista asetelmaa. Näin ollen erilaisia asetelmia on kaiken kaikkiaan

$$\sum_{k=0}^{5} P(30, 6-k) = P(30, 6) + P(30, 5) + \cdots + P(30, 1) = 1206.$$

Kielletään nyt sellaiset asetelmat, joissa jossakin kahdessa laatikossa on sama määrä palloja. Tässä tapauksessa selvästi enintään yksi laatikko voidaan jättää tyhjäksi. Olkoon $0 \leq k \leq 1$ tyhjien laatikkojen lukumäärä. Jäljellä olevat $6-k$ laatikkoa tulisi nyt täyttää siten, että missään kahdessa laatikossa ei ole samaa lukumäärää palloja. Tämä voidaan tehdä vaatimalla $1 \leq a_1 < a_2 < \cdots < a_{6-k}$ ja $\sum_{j=1}^{6-k} a_j = 30$. Näin ollen erilaisia asetelmia on yhtä monta kuin joukossa

$$\mathcal{P}'(30, 6-k) = \{(a_1, \ldots, a_{6-k}) \ : \ 1 \leq a_1 < \cdots < a_{6-k} \text{ ja } \sum_{j=1}^{6-k} a_j = 30\}$$

on alkioita. Nyt huomataan, että kuvaus

$$(a_1, \ldots, a_{6-k}) \mapsto (a_1, a_2 - 1, a_3 - 2, \ldots, a_{6-k} - (6-k-1))$$

joukolta $\mathcal{P}'(30, 6-k)$ osituksien joukolle $\mathcal{P}(30 - 1 - 2 - \cdots - (6-k-1), 6-k)$ on bijektio, joten

$$|\mathcal{P}'(30, 6-k)| = P(30 - \frac{(6-k-1)(6-k)}{2}, 6-k),$$

ja näin ollen erilaisia tapoja asettaa pallot laatikoihin tehtävän vaatimalla tavalla on

$$P(30-15, 6) + P(30-10, 5) = P(15, 6) + P(20, 5) = 26 + 84 = 110$$

kappaletta.

Tehtävä 3

Seurataan kirjan algoritmia 3.11.

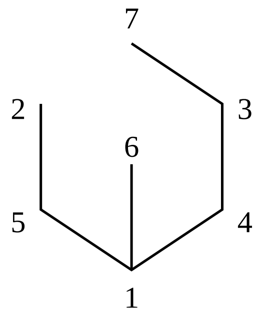
Syöte: Lista $[L[1], L[2], \ldots, L[n-2]]$.

- Puun solmujen lkm n = listan pituus + 2.
- Solmun asteluku = solmun esiintymiskerrat listassa + 1.
- Lisätään $L[n-1] \leftarrow 1$.
- Toistetaan kierroksille $i = 1, \ldots, n-1$.
  1. Etsitään suurin solmu x, jonka asteluku on 1.
  2. Lisätään kaari $\{x, L[i]\}$ puuhun.
  3. Vähennetään solmujen x ja $L[i]$ astelukua yhdellä.

Annetun listan $[3, 1, 4, 1, 5]$ pituus on 5, joten puussa on $n = 7$ solmua. Edetään algoritmin mukaisesti.

		asteluvut							
i	$L[i]$	1	2	3	4	5	6	7	$x, L[i]$
1	3	3	1	2	2	2	1	$\underline{1}$	$7, 3$
2	1	3	1	1	2	2	$\underline{1}$	0	$6, 1$
3	4	2	1	$\underline{1}$	2	2	0	0	$3, 4$
4	1	2	1	0	$\underline{1}$	2	0	0	$4, 1$
5	5	1	$\underline{1}$	0	0	2	0	0	$2, 5$
6	1	1	0	0	0	$\underline{1}$	0	0	$5, 1$

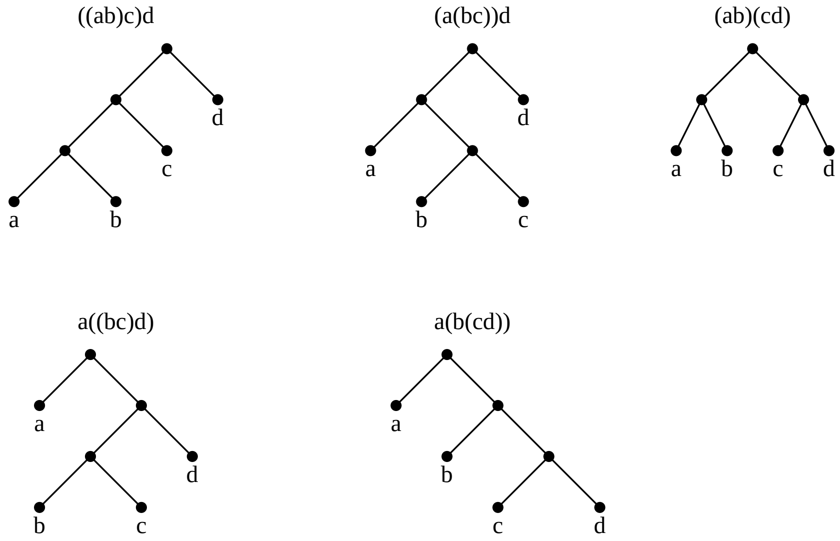
Tuloksena saadaan Kuvan 1 graafi.



Kuva 1: Prüferin listaesitystä $[3, 1, 4, 1, 5]$ vastaava graafi.

Tehtävä 4

Jokaista $n + 1$ termin tulon ryhmittelyä kahden alkion tuloiksi vastaa yksikäsitteisesti juurellinen järjestetty binääripuu, jonka haaroissa on kertolaskuoperaattori ja lehdissä tulon termit. Kuvassa 2 on esitetty kaikki neljän termin tulon eri ryhmittelytavat ja näitä vastaavat jäsennyspuut.



Kuva 2: Neljän termin tulon eri ryhmittelytavat ja näitä vastaavat jäsennyspuut.

Täysi binääripuu (full binary tree) on juurellinen puu, jossa jokainen solmu on joko lehtisolmu, tai sillä on kaksi lapsisolmua. Oletetaan lisäksi, että jokaisen solmun lapsisolmut on järjestetty, jotta voidaan puhua käsitteistä “vasen” ja “oikea” lapsi. Tällöin on selvää, että jokainen $n + 1$ termin tulon jäsennyspuu on täysi binääripuu, jossa on $n + 1$ lehtisolmua, ja kääntäen. Induktiolla voidaan osoittaa seuraava aputulos:

Aputulos. Täydessä binääripuussa, jossa on $n + 1$ lehtisolmua, on täsmälleen n solmua, joilla on kaksi lapsisolmua.

Todistus. Perustapauksessa $n = 0$ väite on ilmeinen. Oletetaan nyt, että väite pätee tapauksissa $0 \leq k \leq n$, ja tarkastellaan täyttä binääripuuta, jossa on $(n+1)+1 \geq 2$ lehtisolmua. On selvää, että tällöin puun juurella tulee olla kaksi lapsisolmua. Olkoon l_1 juuren vasemman lapsen virittämän alipuun lehtisolmujen lukumäärä, ja l_2 vastaavasti juuren oikean lapsen virittämän alipuun lehtisolmujen lukumäärä. Koska juuri ei ole lehtisolmu, on oltava $n + 2 = l_1 + l_2$ ja $l_1, l_2 \geq 1$, joten $l_1 \leq n+1$ ja $l_2 \leq n+1$. Näin ollen induktio-oletusta voidaan soveltaa juuren vasemman ja oikean lapsen virittämiin alipuihin, jotka selvästi ovat täysiä binääripuita. Koko puussa on siten $1+(l_1-1)+(l_2-1) = l_1+l_2-1 = n+1$ solmua (juuri mukaanluettuna), joilla on kaksi lapsisolmua. $\square$

Aputuloksen välitön seuraus on, että täydessä binääripuussa, jossa on $n+1$ lehtisolmua, on täsmälleen $2n$ kaarta.

Catalanin perhe C_n koostuu kaikista $2n$-pituisista binäärijonoista $a_1a_2\cdots a_{2n}$, joille pätee

(a) jonossa on täsmälleen n nollaa ja n ykköstä, ja

(b) alijonossa $a_1a_2\cdots a_i$ on vähintään yhtä monta nollaa kuin siinä on ykköstä kaikilla $1 \le i \le 2n$.

Nyt on helppoa johtaa vastaavuus $(n+1)$-lehtisolmuisten täysien binääripuiden ja Catalanin perheen välille:

Olkoon annettu täysi binääripuu, jolla on $n+1$ lehtisolmua. Merkitään kaikki vasempaan lapseen johtavat kaaret nollalla, ja oikeaan lapseen johtavat kaaret ykkösellä. Käydään puu nyt läpi esijärjestyksessä (ensin solmu itse, sitten rekursiivisesti vasen lapsi, ja lopuksi rekursiivisesti oikea lapsi) siten, että tiettyyn solmuun saavuttaessa tulostetaan solmun isään johtavan kaaren numero.

On selvää, että tällöin jokainen $2n$ kaaresta tulostetaan kerran. Lisäksi tulostejono täyttää ehdon (a), koska jokaisella solmulla, jolla on vasen lapsi, on myös oikea lapsi; toisaalta ehto (b) täyttyy, koska jokaisen kaksilapsisen solmun vasempaan lapseen vievän kaaren numero tulostetaan aina ennen oikeaan lapseen vievän kaaren numeroa.

Toisaalta jos on annettu Catalanin perheeseen kuuluva binäärijono $a_1a_2\cdots a_{2n}$, voidaan sitä vastaava täysi binääripuu konstruoida seuraavasti:

1. Alussa puu koostuu vain sen juuresta, joka on lehtisolmu. Asetetaan juuri työsolmuksi.

2. Toistetaan järjestyksessä kaikille $i = 1, 2, \ldots, 2n$:

   (i) Jos $a_i = 0$, luodaan työsolmulle vasen lapsi, ja asetetaan lapsi työsolmuksi.

   (ii) Jos $a_i = 1$, edetään työsolmusta puun juurta kohti ensimmäiseen sellaiseen solmuun, jolla ei ole oikeata lapsisolmua. Luodaan tälle solmulle oikea lapsisolmu, ja asetetaan se työsolmuksi.

Konstruktio on hyvin määritelty, koska tasapainoehdon (b) mukaan kohdassa (ii) solmu, jolle voidaan lisätä oikea lapsi, on aina olemassa. Toisaalta tasapainoehdosta (a) saadaan, että jokaiselle solmulle, jolle on lisätty vasen lapsi kohdan (i) mukaisesti, lisätään myös oikea lapsi kohdan (ii) mukaisesti. Konstruktion tuloksena syntyy täysi binääripuu, jossa on $2n+1$ solmua ja $2n$ kaarta. Nyt voidaan jälleen osoittaa induktiolla, että tällaisessa puussa on oltava $n+1$ lehtisolmua.

Näin ollen konstruktio tuottaa aina tulokseksi $(n+1)$-lehtisolmuisen täyden binääripuun.

Konstruktioiden määräämät kuvaukset puilta binäärijonoille (ja kääntäen binäärijonoilta puille) on nyt helppo todeta toistensa käänteiskuvauksiksi.

Laskuharjoitus 4

7.2.2002

1. Kuvaile peräytymishakualgoritmit seuraaville ongelmille:

   (a) Etsi kaikki tavat asettaa n kuningatarta $n \times n$-shakkilaudalle siten, etteivät kuningattaret uhkaa toisiaan.

   (b) Etsi $n \times n$ miinaharavapelin (minesweeper) annetusta peliasemasta ruudut

      i. joissa ei varmasti ole miinaa; ja

      ii. joissa on varmasti miina.

   (Miinojen lukumäärä oletetaan tuntemattomaksi.)

2. Määritellään graafin $\mathcal{G}$ kromaattinen luku

$$\chi(\mathcal{G}) = \min\{k \ : \ \mathcal{G}\text{:lla on solmujen } k\text{-väritys}\}$$

   ja klikkiluku

$$\omega(\mathcal{G}) = \max\{k \ : \ \mathcal{G}\text{:ssä on } k\text{:n kokoinen klikki}\}.$$

   Kirjan lemman 4.4 mukaan $\omega(\mathcal{G}) \leq \chi(\mathcal{G})$. Osoita, että epäyhtälö voi olla myös aito: etsi graafi, jolle $\omega(\mathcal{G}) < \chi(\mathcal{G})$.

3. Olkoon annettuna graafi $\mathcal{G}$ ja sen solmuille jokin järjestys. Ahne graafinväritysalgoritmi värittää graafin solmu kerrallaan pienimmästä suurimpaan siten, että kulloinkin väritettävän solmun väriksi valitaan pienin väri (oletetaan värit järjestetyiksi), joka ei esiinny väritettävän solmun jo väritetyissä naapureissa.

   (a) Esitä ääretön perhe graafeja, joiden solmut on järjestetty siten, että ahne väritysalgoritmi ei tuota optimaalista väritystä. (Väritys on optimaalinen jos värien lukumäärä on $\chi(\mathcal{G})$.)

   (b) Onko kaikille graafeille olemassa sellainen solmujen järjestys, että ahne algoritmi tuottaa optimaalisen värityksen?

Laskuharjoituksen 4 vastaukset

Tehtävä 1 (a)

Oletetaan shakkipelin säännöt tunnetuiksi. Tarkastellaan kutakin laudan riviä $1, \ldots, n$ vuorollaan. Valintajoukkona on tällöin niiden tällä rivillä olevien ruutujen joukko, joita ei vielä uhkaa laudalla oleva kuningatar. Kokeillaan kutakin näistä vuorollaan ja siirrytään seuraavalle riville.

Rivejä ei välttämättä kannata käydä numerojärjestyksessä läpi. Tehokkaampi vaihtoehto on järjestää haku niin, että kulloinkin tarkasteltavaksi riviksi valitaan se, jolla sillä hetkellä on vähiten uhkaamattomia ruutuja. Tällöin haun haarautumiskerroin lähellä hakupuun juurta on pieni, jolloin hakusolmujen määrä pienenee, vaikka kaikki vaihtoehdot käydäänkin yhtä kattavasti läpi.

Tehtävä 1 (b)

Miinaharavapelissä pelaajan tehtävänä on löytää annetulta $n \times n$ pelilaudalta kaikki ruudut, jotka sisältävät miinan. Pelaajan käytettävissä on miinaharava, joka ilmoittaa pelilaudan ruudussa miinojen lukumäärän kyseistä ruutua ympäröivissä ruuduissa. Pelin peliasema koostuu joukosta tunnettuja ruutuja, joissa tiedetään miinaharavan lukema. Peli etenee asemasta toiseen siten, että pelaaja osoittaa laudalta tuntemattoman ruudun. Jos ruudussa oli miina, peli päättyy. Jos ruudussa ei ollut miinaa, miinaharavan uusi lukema liitetään peliasemaan, ja peli jatkuu. Peli päättyy pelaajan voittoon, kun kaikissa tuntemattomissa ruuduissa on miina. (Tässä pelin versiossa miinojen lukumäärä oletetaan tuntemattomaksi.)

Olkoon pelilaudan ruutujen joukko $D_n = \{1, 2, \ldots, n\} \times \{1, 2, \ldots, n\}$. Ruudun (x, y) *ympäristö* on joukko

$$N(x, y) = \{(x, y) + (\delta_1, \delta_2) \ : \ -1 \leq \delta_1, \delta_2 \leq 1, \ (\delta_1, \delta_2) \neq (0, 0)\} \cap D_n.$$

Huomaa, että $(x, y) \notin N(x, y)$.

Olkoon $\mathsf{mines} \subseteq D_n$ niiden ruutujen joukko, joissa on miina. (Tämä joukko on luonnollisesti pelaajalle tuntematon.) Miinaharavan lukema ruudussa (x, y) on tällöin

$$d(x, y) = |\mathsf{mines} \cap N(x, y)|.$$

Olkoon $\mathsf{known} \subseteq D_n$ pelaajan osoittamien ruutujen joukko. Jos peli ei ole päättynyt, pätee $\mathsf{known} \cap \mathsf{mines} = \emptyset$. Oletetaan tämä todeksi.

Koska miinojen kokonaismäärä on tuntematon, pelaaja voi saada tietoa miinojen lukumäärästä ja sijainnista vain miinaharavan tunnetussa alueessa antamien lukemien avulla. Määritelmän mukaan miinaharavan lukema ruudussa

(x, y) antaa tietoa vain ruutujen $N(x, y)$ sisältämistä miinoista, joten pelaaja voi päätellä miinojen sijainnista jotain vain tunnetun alueen *reuna-alueella*

$$\mathsf{boundary} = \Big(\bigcup_{(x,y)\in\mathsf{known}} N(x, y)\Big) \setminus \mathsf{known}.$$

Reuna-alueen ulkopuolisista tuntemattomista ruuduista ei voida päätellä mitään. (Huomaa, että edellinen väite ei pidä paikkaansa jos miinojen lukumäärä on tunnettu!)

Rajoitetaan tarkastelu reuna-alueeseen $\mathsf{boundary}$. Sanotaan joukkoa

$$E \subseteq \mathsf{boundary}$$

konsistentiksi, jos kaikilla $(x, y) \in \mathsf{known}$ pätee $d(x, y) = |E \cap N(x, y)|$. Tällöin joukko on konsistentti jos ja vain jos on mahdollista, että miinat todellisuudessa sijaitsevat joukon E ruuduissa. Erityisesti, joukko $\mathsf{mines} \cap \mathsf{boundary}$ on konsistentti.

Olkoon $\mathscr{E}$ kaikkien konsistenttien joukkojen joukko. Määritellään joukot

$$\mathsf{safe} = \mathsf{boundary} \setminus \bigcup_{E\in\mathscr{E}} E, \quad \text{ja} \quad \mathsf{flag} = \bigcap_{E\in\mathscr{E}} E.$$

Tarkastellaan ensin joukkoa safe. Nyt $\mathsf{safe} \cap \mathsf{mines} = \emptyset$, koska $\mathsf{mines} \cap \mathsf{boundary} \in \mathscr{E}$. Joukko safe on siis joukko turvallisia ruutuja, joissa ei varmasti ole miinaa. Toisaalta safe on myös suurin mahdollinen joukko jolle edellinen väite voidaan esittää, koska kaikki pisteet $x \in \mathsf{boundary} \setminus \mathsf{safe}$ sisältyvät johonkin konsistenttiin joukkoon E, ja miinaharavan lukemien perusteella ei voida sulkea vaihtoehtoa $\mathsf{mines} \cap \mathsf{boundary} = E$ pois.

Tarkastellaan seuraavaksi joukkoa flag. Koska $\mathsf{flag} \subseteq E$ kaikilla $E \in \mathscr{E}$, pätee tämä erityisesti joukolle $E = \mathsf{mines} \cap \mathsf{boundary}$. Joten, $\mathsf{flag} \subseteq \mathsf{mines}$. Joukko flag sisältää siis vain ruutuja, joissa on miina. Toisaalta kaikille pisteille $x \in \mathsf{boundary} \setminus \mathsf{flag}$ on olemassa konsistentti joukko E, jolle $x \notin E$. Näin ollen flag on suurin joukko, jonka ruuduissa on varmasti miina.

Joukot safe ja flag ovat siis täsmälleen tehtävän (i) ja (ii)-kohdissa kysytyt joukot.

Peräytyvää hakua voidaan käyttää joukkojen safe ja flag määrittämiseen esimerkiksi seuraavasti:

- Laaditaan peräytyvä hakualgoritmi, joka tuottaa ratkaisuina kaikki konsistentit joukot.
- Ennen hakua alustetaan $\mathsf{safe} \leftarrow \mathsf{boundary}$ ja $\mathsf{flag} \leftarrow \mathsf{boundary}$.

- Käydään peräytyvällä haulla läpi kaikki konsistentit joukot E. Aina konsistentin joukon löydyttyä haun aikana asetetaan

$$\mathsf{safe} \leftarrow \mathsf{safe} \setminus E \quad \text{ja} \quad \mathsf{flag} \leftarrow \mathsf{flag} \cap E.$$

- Kun kaikki konsistentit joukot on käyty läpi, joukot safe ja flag ovat halutut joukot. (Huomaa, että molemmat joukot voivat olla tyhjiä. Tämä tapahtuu esimerkiksi pelin alkutilanteessa, jolloin $\mathsf{known} = \emptyset$.)

Konsistentit joukot voidaan tuottaa peräytyvällä haulla esimerkiksi seuraavasti.

- Olkoon joukossa boundary N alkiota $P_1, \ldots, P_N$, $N \geq 1$. (Erikoistapaus $N = 0$ jätetään huomiotta.)
- Mallinnetaan joukon boundary osajoukkoja $E \subseteq \mathsf{boundary}$ binäärilistoina $\vec{E} = [x_1, \ldots, x_N]$, joille $x_i = 1 \Leftrightarrow P_i \in E$ ja $x_i = 0 \Leftrightarrow P_i \notin E$.
- Osittaisratkaisu on lista $[x_1, \ldots, x_{k-1}]$; alussa tyhjä lista.
- Hakupuussa syvyydellä k lisätään listaan vuorotellen alkiot $x_k = 0$ ja $x_k = 1$, ja jatketaan hakua eteenpäin.
- Kun listassa on N alkiota, tarkistetaan, onko sitä vastaava joukko konsistentti. (Jos on, päivitetään joukkoja safe ja flag kuten edellä.) Tämän jälkeen palataan aiemmalle tasolle hakupuussa.
- Hakua voidaan *huomattavasti* tehostaa karsimalla sellaisia osittaisratkaisuja, joita ei selvästi voida täydentää konsistenteiksi joukoiksi. Palataan tähän tuonnenpana.

Tehtävä 2

Viiden solmun sykli $\mathcal{C}_5$ (viisikulmio) on kromaattiselta luvultaan 3, mutta sen suurimman klikin koko on 2.

Tehtävä 3 (a)

Olkoon $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ graafi, ja $\mathcal{W} \subseteq \mathcal{V}$. Solmujoukko $\mathcal{W}$ on *riippumaton* (independent), jos minkään sen kahden solmun välillä ei ole kaarta. Graafi $\mathcal{G}$ on *kaksijakoinen* (bipartite) jos sen solmujoukko $\mathcal{V}$ voidaan jakaa kahteen ei-tyhjään riippumattomaan joukkoon $\mathcal{W}$ ja $\mathcal{V} - \mathcal{W}$.

Kaksijakoinen graafi voidaan aina värittää kahdella värillä, koska molempiin riippumattomiin joukkoihin riittää yksi väri.

Täydellinen kaksijakoinen (complete bipartite) graafi $(\mathcal{V}, \mathcal{E})$ sisältää kaikki mahdolliset kaaret sen kahden riippumattoman joukon välillä, ts. kaikilla $u \in \mathcal{W}$ ja kaikilla $v \in \mathcal{V} - \mathcal{W}$ pätee $\{u, v\} \in \mathcal{E}$.

Konstruoidaan graafiperhe seuraavasti: Otetaan täydellinen kaksijakoinen graafi, jonka molemmissa riippumattomissa joukoissa on n solmua, ja järjestetään solmut siten, että ensimmäinen riippumaton joukko koostuu parittomista solmuista $\{1, 3, \ldots, 2n-1\}$, ja toinen parillisista $\{2, 4, \ldots, 2n\}$. Poistetaan graafista kaaret $\{1, 2\}$, $\{3, 4\}$, $\{5, 6\}, \ldots, \{2n-1, 2n\}$.

Ahne algoritmi värittää graafin seuraavasti: Solmulle 1 annetaan väri 1. Solmulle 2 annetaan väri 1, koska graafissa ei ole kaarta $\{1, 2\}$. Solmulle 3 annetaan väri 2, koska graafissa on kaari $\{2, 3\}$, ja väriä 1 ei näin ollen voi käyttää. Solmulle 4 annetaan väri 2, koska kaaren $\{1, 4\}$ takia väriä 1 ei voi käyttää. Solmulle 5 annetaan väri 3, koska kaarien $\{2, 5\}$ ja $\{4, 5\}$ takia värejä 1 ja 2 ei voi käyttää. Solmulle 6 annetaan väri 3, koska kaarien $\{1, 6\}$ ja $\{3, 6\}$ takia värejä 1 ja 2 ei voi käyttää, jne.

Yleisessä tapauksessa pariton solmu $2k-1$ ($k \geq 2$) väritetään värillä k, koska kaaret $\{2j, 2k-1\}$, $1 \leq j \leq k-1$ estävät k:ta pienempien värien käytön, sillä solmu $2j$ on väritetty värillä j kaikilla $1 \leq j \leq k-1$. Toisaalta parillinen solmu $2k$ ($k \geq 2$) väritetään värillä k, koska kaaret $\{2j-1, 2k\}$, $1 \leq j \leq k-1$ estävät k:ta pienempien värien käytön, sillä solmu $2j-1$ on väritetty värillä j kaikilla $1 \leq j \leq k-1$.

Graafin ahneeseen väritykseen tarvitaan siis lopulta n väriä, vaikka graafi on väritettävissä ainoastaan kahdella värillä.

Tehtävä 3 (b)

On selvää, että värityksestä riippumatta samalla värillä väritetyt graafin solmut muodostavat riippumattoman joukon. Oletetaan, että käytettävissä on optimaalinen väritys. Jos graafin solmut nyt listataan väriluokka kerrallaan (esim. ensin kaikki punaiset solmut, sitten siniset, jne.), saadaan solmuille järjestys, jolla ahne algoritmi tuottaa optimaalisen värityksen. Saatu ahne väritys ei välttämättä ole identtinen alkuperäisen värityksen kanssa, mutta värien lukumäärä on sama.

Laskuharjoitus 5

14.2.2002

1. Jatkoa edellisen laskuharjoituksen 1(b) tehtävään: Etsi keinoja tehostaa peräytyvää hakua miinaharavapelin konsistenttien joukkojen määrittämiseen.

2. Kirjan tehtävä 4.13: *Latinalainen neliö* on $n \times n$-taulukko, jonka jokaisessa rivissä ja sarakkeessa kukin luvuista $1, \ldots, n$ esiintyy tasan kerran. Latinalaista neliötä kutsutaan redusoiduksi latinalaiseksi neliöksi, jos sen ensimmäisen rivin ja ensimmäisen sarakkeen alkiot esiintyvät järjestyksessä $1, 2, \ldots, n$. Kirjoita ohjelma, joka laskee peräytymishaulla redusoitujen $n \times n$ latinalaisten neliöiden lukumäärän.

3. Esitä jokin mielekäs kustannusfunktio ja naapuristo seuraaville vaikeille kombinatorisille optimointiongelmille.

   (a) Symmetrinen kauppamatkustajan ongelma.

   (b) Graafin väritys n värillä siten, että naapurisolmuilla on eri väri.

   (c) Graafin ositus: Jaa graafin solmut, joita on parillinen määrä, kahteen yhtäsuureen osaan siten, että osien välisten kaarien määrä on mahdollisimman pieni.

   (d) Lukujen ositus: jaa joukko $A = \{a_{1,}a_{2,}, \ldots, a_n\}$ kahteen osaan A_1 ja A_2 siten, että $A_1 \cup A_2 = A$, $A_1 \cap A_2 = \emptyset$ ja

$$\sum_{a \in A_1} a = \sum_{a \in A_2} a.$$

Laskuharjoituksen 5 vastaukset

Tehtävä 1

Jatketaan edellisen viikon (7.2.2002) tehtävän 1 (b) yhteydessä määriteltyjen käsitteiden käyttöä.

Tehtävänä on siis löytää annetusta miinaharavan peliasemasta joukon $\mathsf{boundary}$ konsistentit osajoukot, ts. kaikki mahdolliset tavat asetella miinat joukon $\mathsf{boundary}$ ruutuihin siten, että miina-asetelma vastaa miinaharavan antamia lukemia.

Naiivi ratkaisu toimii "tuota ja testaa"-periaatteella ("generate and test"), ts. tuottaa kaikki joukon $\mathsf{boundary}$ osajoukot, ja testaa jälkikäteen onko tuotettu joukko konsistentti. Tämä ei luonnollisesti ole kovin tehokasta, koska miinaharavan antamat lukemat joukon $\mathsf{boundary}$ ympäristössä rajoittavat konsistenttien joukkojen lukumäärän tyypillisesti vain murto-osaan kaikista joukon $\mathsf{boundary}$ osajoukoista. Erityisesti, miinaharavan lukemat $d(x, y)$ joukon

$$\mathsf{constraints} = \{(x, y) \in \mathsf{known} \ : \ N(x, y) \cap \mathsf{boundary} \neq \emptyset\}$$

ruuduissa asettavat (kaikki) rajoitteet konsistenttien joukkojen rakenteelle.

Muotoillaan nyt perätyvä hakualgoritmi siten, että joukon $\mathsf{constraints}$ määräämien ehtojen toteutumista (toteutumattomuutta) tarkkaillaan jo osittaisratkaisuille, jolloin läpikäytävästä hakuavaruudesta tulee huomattavasti pienempi kuin naiivin ratkaisun tapauksessa. Algoritmin pseudokoodi on alla:

```
CONSISTENT_SETS(E, R)
    if R = ∅
        check consistency of E, output solution if consistent.
        return
    for (x, y) ∈ constraints
        if d(x, y) = |E ∩ N(x, y)| and R ∩ N(x, y) ≠ ∅                        (i)
            CONSISTENT_SETS(E, R \ N(x, y))
            return
        if d(x, y) = |E ∩ N(x, y)| + |R ∩ N(x, y)| and R ∩ N(x, y) ≠ ∅        (ii)
            CONSISTENT_SETS(E ∪ (R ∩ N(x, y))), R \ N(x, y))
            return
        if d(x, y) > |E ∩ N(x, y)| and R ∩ N(x, y) = ∅                        (iii)
            return
        if d(x, y) < |E ∩ N(x, y)|                                            (iv)
            return
    P ← any point from R
    CONSISTENT_SETS(E, R \ {P})
    CONSISTENT_SETS(E ∪ {P}, R \ {P})
    return
```

Algoritmi CONSISTENT_SETS ottaa kaksi argumenttia E, R, missä

- $E \subseteq \mathsf{boundary}$ on joukko, joka sisältää ruudut (x, y), joihin algoritmi on päättänyt asettaa miinan.
- $R \subseteq \mathsf{boundary}$ on joukko, joka sisältää ruudut (x, y), joihin algoritmi ei ole vielä ottanut kantaa (asetetaanko miina vaiko ei).

Kaikkien konsistenttien joukkojen määrittämiseksi algoritmia kutsutaan argumenteilla

$$E = \emptyset \quad \text{ja} \quad R = \mathsf{boundary}.$$

Algoritmin jokaisella rekursioaskeleella joukko R pienenee, kunnes lopulta $R = \emptyset$, jolloin tarkistetaan joukon E konsistenssi ja tämän jälkeen palataan takaisin edelliselle tasolle hakupuussa. “for”-silmukan tehtävänä on toisaalta estää haun haarautumista (ehdot (i) ja (ii)) ja toisaalta katkaista haku, jos osittaisratkaisu E on mahdoton täydentää konsistentiksi joukoksi (ehdot (iii) ja (iv)). Huomaa, että algoritmi toimii (kuten em. naiivi algoritmi) vaikka koko “for”-silmukka jätettäisiin pois.

Tarkastellaan ehtoja (i)–(iv) yksi kerrallaan. Ehdot (i) ja (ii) havaitsevat nk. “pakkotilanteet”, joissa osittaisratkaisua E on jatkettava täsmälleen tietyllä tavalla, jotta lopputuloksena syntyvästä joukosta tulisi konsistentti. Erityisesti,

- ehto (i) havaitsee tilanteen, jossa miinaharavan lukema $d(x, y)$ ruudussa $(x, y) \in \mathsf{constraints}$ vastaa täsmälleen ruudun ympäristöön $N(x, y)$ jo asetettujen miinojen lukumäärää $|E \cap N(x, y)|$. Nyt jos johonkin joukon $N(x, y)$ ruutuun ei vielä ole otettu kantaa, on selvää, että tällaisiin ruutuihin *ei* voida asettaa miinaa, koska muutoin tunnettu miinaharavan lukema $d(x, y)$ ylittyisi; toisaalta
- ehto (ii) havaitsee tilanteen, jossa miinaharavan lukema $d(x, y)$ ruudussa $(x, y) \in \mathsf{constraints}$ vastaa täsmälleen ruudun ympäristöön $N(x, y)$ jo asetettujen miinojen lukumäärää $|E \cap N(x, y)|$ lisättynä niiden ympäristön $N(x, y)$ ruutujen lukumäärällä, joihin ei ole vielä otettu kantaa. On selvää, että tällöin kaikkiin ruutuihin joukossa $R \cap N(x, y)$ on asettava miina, jotta lopullisessa ratkaisussa olisi riittävästi miinoja toteuttamaan lukema $d(x, y)$.

Ehdot (iii) ja (iv) karsivat osittaisratkaisuja, joita ei voida mitenkään täydentää konsistentiksi joukoksi.

- Ehto (iii) havaitsee tilanteen, jossa ruudun (x, y) ympäristöön on asetettu liian vähän miinoja, eikä tilannetta voida enää korjata, koska kaikkiin ruutuihin ympäristössä $N(x, y)$ on jo otettu kantaa.

- Ehto (iv) havaitsee tilanteen, jossa ruudun (x, y) ympäristössä on liikaa miinoja.

Tehtävä 2

Oheinen ohjelma toteuttaa tehtävän yksinkertaisella tavalla. Koska redusoidussa latinalaisessa neliössä ensimmäinen rivi ja sarake ovat määrätyt, $n \times n$-kokoisessa latinalaisessa neliössä riittää sijoittaa $(n-1)^2$ lukua, mikä siis on haun maksimisyvyys. Ohjelma ei pidä yllä konstruoitavaa latinalaista neliötä, vaan ainoastaan boolean-muuttujia `n_in_row_i` ja `n_in_col_j`, joista nähdään, joko luku n on laitettu riville i tai sarakkeeseen j.

```
/* Etsi kertalukua N olevat redusoidut latinalaiset neliöt */
#define N 4

int n_in_row_i[N+1][N+1]; /* onko numero n jo rivissä i     */
int n_in_col_j[N+1][N+1]; /* onko numero n jo sarakkeessa j */

int place(int depth) {
  int i,j,n,sum;

  if (depth==(N-1)*(N-1)) /* yksi neliö löytyi, koska on */
    return 1;             /* sijoitettu (N-1)**2 numeroa */

  i=2+depth%(N-1); /* lasketaan depthistä, mihin kohtaan */
  j=2+depth/(N-1); /* valitaan numero seuraavaksi        */

  sum=0;
  for (n=1; n<=N; n++) {
    if(!n_in_row_i[n][i] && !n_in_col_j[n][j]) {
      n_in_row_i[n][i]=1;  /* kullekin n:lle vuorollaan:         */
      n_in_col_j[n][j]=1;  /* jos n:ä ei ole laitettu riviin i   */
      sum+=place(depth+1); /* eikä sarakkeeseen j, laitetaan se  */
      n_in_row_i[n][i]=0;  /* näiden risteyskohtaan ja jatketaan */
      n_in_col_j[n][j]=0;  /* hakua; lopuksi otetaan luku pois   */
    }
  }
  return sum;
}

int main(int argc, char **argv) {
  int i,j;
```

```
  for(i=1;i<=N;i++) {      /* nollataan taulukot */
    for(j=1;j<=N;j++) {
      n_in_row_i[i][j]=n_in_col_j[i][j]=0;
    }
  }

  for(i=1;i<=N;i++) {   /* asetetaan luvut 1...n     */
    n_in_row_i[i][i]=1; /* ensimmäiseen sarakkeeseen */
    n_in_col_j[i][i]=1; /* ja riviin                 */
  }
  printf("%d\n", place(0));
  return 0;
}
```

Tehtävä 3 (a)

Kustannusfunktiona luonnollisin valinta on reitin pituus. Eräs mahdollisuus naapuristoksi on ns. 2-OPT -siirrot: muodostetaan nykyistä reittiä kuvaavasta graafista uusi reitti poistamalla kaksi kaarta ja lisäämällä kaksi kaarta.

Tehtävä 3 (b)

Tässä kuvataan ns. fixed-k lähestymistapa. Ratkaisu on mikä tahansa solmujen ositus k osaan (tyhjät osat sallitaan). Kaksi väritystä ovat toistensa naapurit, jos toinen saadaan toisesta siirtämällä jokin solmu osasta toiseen. Kustannusfunktiona esim. niiden kaarien lukumäärä, joiden molemmat solmut ovat samassa osituksen osassa. Naapuristoa voidaan rajoittaa seuraavasti: nykyistä ratkaisua voidaan muuttaa vain niiden solmujen osalta, joista lähtee ainakin yksi väritysehtoa rikkova kaari; tällöin tosin naapuristo ei enää ole symmetrinen.

Tehtävä 3 (c)

Yksinkertainen lähestymistapa: ratkaisujoukkona on kaikki solmujen ositukset kahteen osaan. Siirto on kahden solmun vaihto osien välillä. Kustannusfunktiona kaarien määrä osien välillä.

Parempi suorituskyky on saatu simuloidulla jäähdytyksellä, kun on sallittu mielivaltaiset ositukset kahteen osaan V_1 ja V_2. Siirtoina on nyt yhden solmun siirto osasta toiseen, ja kustannusfunktiossa on lisäksi sakkona summatermi

$$\alpha(|V_1| - |V_2|)^2,$$

joka lisää kustannusta aina kun osat ovat erikokoiset. Optimointialgoritmi saattaa sakkotermistä huolimatta päätyä epätasapainoiseen ratkaisuun, jolloin voidaan joko kasvattaa α:n arvoa tai lisätä jokin yksinkertainen heuristiikka, joka muutamaa solmua siirtämällä pääsee tasapainotettuun ratkaisuun.

Tehtävä 3 (d)

Otetaan ratkaisujoukoksi kaikki ositukset kahteen osaan. Määritellään k-naapuristo sellaiseksi, jossa yhdellä siirrolla saadaan siirtää enintään k alkiota osituksesta toiseen. Kustannusfunktioksi voidaan ottaa

$$\left| \sum_{a \in A_1} a - \sum_{a \in A_2} a \right| .$$

Tämä ongelma on hiukan hankala sikäli, että siinä on pahimmillaan hyvin paljon lokaaleja optimeja. Simuloitu jäähdytys esitetyllä naapuristolla ei pärjää erityisesti tähän ongelmaan kehitetylle Karmarkar-Karp -heuristiikalle, mutta parempaa on kehitetty.

Laskuharjoitus 6

28.2.2002

Tutkitaan paikallisen haun naapurustojen määräämiä graafeja.

Konfiguraatiograafin solmuina ovat kaikki hakuavaruuden $\mathcal{X}$ alkiot, ja kahden solmun välillä on kaari jos ja vain jos solmut kuuluvat toistensa naapurustoon. (Oletetaan naapurusto $\mathsf{N} : \mathcal{X} \to 2^{\mathcal{X}}$ symmetriseksi, ts. kaikilla $x, y \in \mathcal{X}$ pätee $x \in \mathsf{N}(y)$ jos ja vain jos $y \in \mathsf{N}(x)$.)

1. Esitä mielekäs konfiguraatiograafi seuraaville ongelmille:

   (a) Eksakti peitto: etsi annetusta kokoelmasta perusjoukon osajoukkoja sellainen kokoelma osajoukkoja, jossa jokainen perusjoukon piste esiintyy täsmällään yhdessä osajoukossa.

   (b) Graafin solmujen väritys enintään k:lla värillä.

   (c) Kauppamatkustajan ongelma (transpositionaapurustolla).

   (d) Graafin ositus: Jaa graafin solmut, joita on parillinen määrä, kahteen yhtäsuureen osaan siten, että osien välisten kaarien määrä on mahdollisimman pieni/suuri.

   (e) Etsi k-klikki annetusta graafista.

2. Paikallisen haun kannalta mielenkiintoisia konfiguraatiograafin ominaisuuksia ovat mm.

   (a) solmujen lukumäärä (order)

   (b) solmun naapurisolmujen lukumäärä (vertex degree)

   (c) onko graafi kytketty (connected), eli onko jokaisesta solmusta polku kaikkiin muihin solmuihin

   (d) lyhimmän syklin pituus (girth)

   (e) pisimmän syklin pituus (circumference)

   (f) graafin halkaisija (diameter), eli pisimmän kahta solmua yhdistävän lyhimmän polun pituus.

   Määritä nämä ominaisuudet 1. tehtävän konfiguraatiograafeille.

Laskuharjoituksen 6 vastaukset

Tehtävä 1 (a,b)

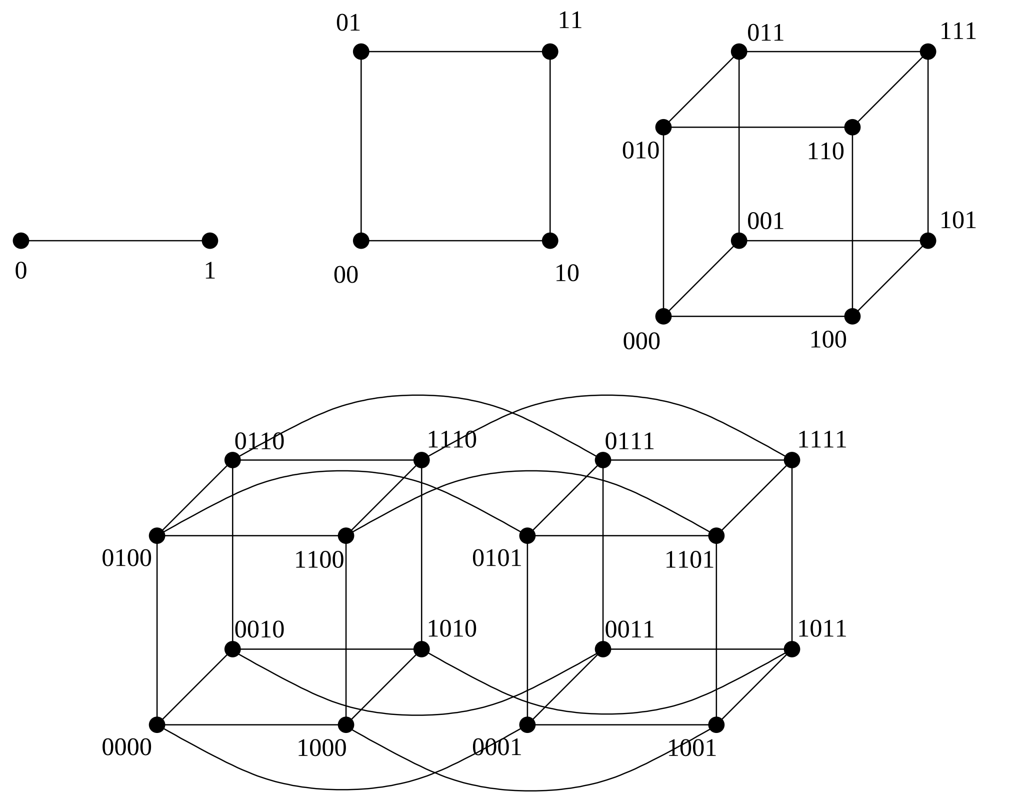
Oletetaan (a)-kohdassa, että annetussa osajoukkokokoelmassa on täsmälleen n osajoukkoa $E_1, E_2, \ldots, E_n \subseteq F$, missä F on perusjoukko. Paikallisen haun tehtävä on nyt löytää osajoukko $\{j_1, \ldots, j_w\} \subseteq \{1, \ldots, n\}$ siten, että $E_{j_1} \cup \cdots \cup E_{j_w} = F$ ja $E_{j_s} \cap E_{j_t} = \emptyset$ aina kun $s \neq t$. (Ts. kokoelma $E_{j_1}, \ldots, E_{j_w}$ osittaa joukon F.)

Valitaan hakuavaruudeksi $\mathcal{X}$ kaikki joukon $\{1, \ldots, n\}$ osajoukot, ja määritellään naapurusto N minimimuutosperiaatteella: osajoukko $x \in \mathcal{X}$ on osajoukon $y \in \mathcal{X}$ naapuri jos ja vain jos y on saatu x:stä lisäämällä tai poistamalla yksi alkio. Hakuavaruuden alkiot voidaan samaistaa n-mittaisten binäärijonojen kanssa jo tutuksi käyneellä tavalla.

Oletetaan (b)-kohdassa, että väritettävässä graafissa on täsmälleen n solmua, jotka on nimetty $v_1, \ldots, v_n$. Otetaan hakuavaruudeksi $\mathcal{X}$ kaikki tavat valita jokaiselle graafin solmulle jokin q:sta erilaisesta väristä. (Huomaa, että tällainen värivalinta ei yleisessä tapauksessa ole laillinen graafin väritys q:lla värillä.) Värivalinnat voidaan tällöin selvästi mieltää pituutta n oleviksi sanoiksi yli q-alkioisen aakkoston: sanan i:s kirjainpaikka määrää solmun v_i värin kaikilla $i = 1, 2, \ldots, n$. Naapurusto voidaan jälleen määrätä minimimuutosperiaatteella: värivalinnat x ja y ovat toistensa naapureita jos ja vain jos ne poikkeavat toisistaan täsmälleen yhdessä paikassa.

(a)- ja (b)-kohtien naapurustovalintoja vastaava konfiguraatiograafi on seuraava: *Hamming-graafin* $H(n, q)$ solmujoukko koostuu kaikista pituutta n olevista sanoista yli q-alkioisen aakkoston $\Sigma_q = \{0, 1, 2, \ldots, q-1\}$, ts. $\mathcal{V} = \Sigma_q^n$. Solmujen $x, y \in \mathcal{V}$ välillä on kaari jos ja vain jos sanat x ja y poikkeavat toisistaan täsmälleen yhdessä paikassa.

Alla esimerkkinä Hamming-graafit $H(n, 2)$ kun $n = 1, 2, 3, 4$.

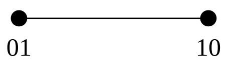


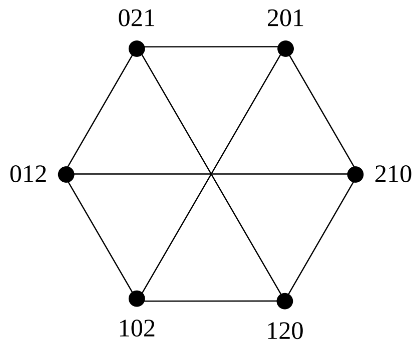
Tehtävä 1 (c)

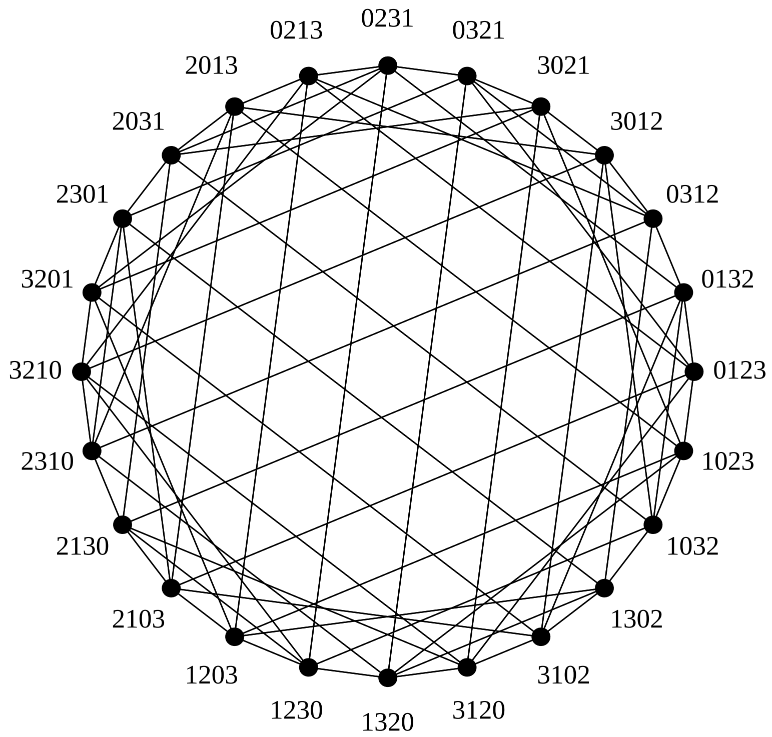
Oletetaan, että kaupunkeja on n kappaletta. Mallinnetaan kauppamatkustajan mahdollisia kierroksia joukon $\{0, 1, 2, \ldots, n-1\}$ permutaatioina. Tällöin permutaation $\pi \in S_n = \mathcal{X}$ listaesitys $[\pi(0), \pi(1), \ldots, \pi(n-1)]$ kertoo missä järjestyksessä kaupungit käydään läpi: lähdetään kaupungista $\pi(0)$, sitten kaupunkiin $\pi(1)$, ..., kunnes palataan kaupungista $\pi(n-1)$ takaisin kaupunkiin $\pi(0)$. Määritellään naapurusto siten, että kaksi permutaatiota $\pi_1, \pi_2 \in \mathcal{X}$ ovat toistensa naapureita jos ja vain jos toinen saadaan toisesta yhden transposition avulla (ts. toinen saadaan toisesta vaihtamalla kahden kaupungin paikkaa permutaation listaesityksessä).

Muodostuva konfiguraatiograafi on transpositioiden generoima symmetrisen ryhmän *Cayley-graafi*. Cayley-graafin $\Gamma(S_n, T_n)$ solmuina ovat kaikki joukon $\{0, 1, 2, \ldots, n-1\}$ permutaatiot $\pi \in S_n$. Permutaatioiden $\pi_1, \pi_2 \in S_n$ välillä on kaari jos ja vain jos $\pi_1^{-1}\pi_2 \in T_n$, missä T_n on kaikkien joukon $\{0, 1, 2, \ldots, n-1\}$ transpositioiden joukko.

Alla esimerkkinä Cayley-graafit $\Gamma(S_n, T_n)$ kun $n = 2, 3, 4$.





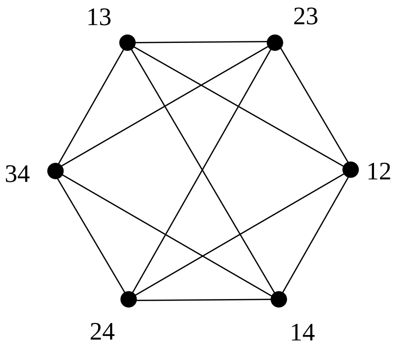


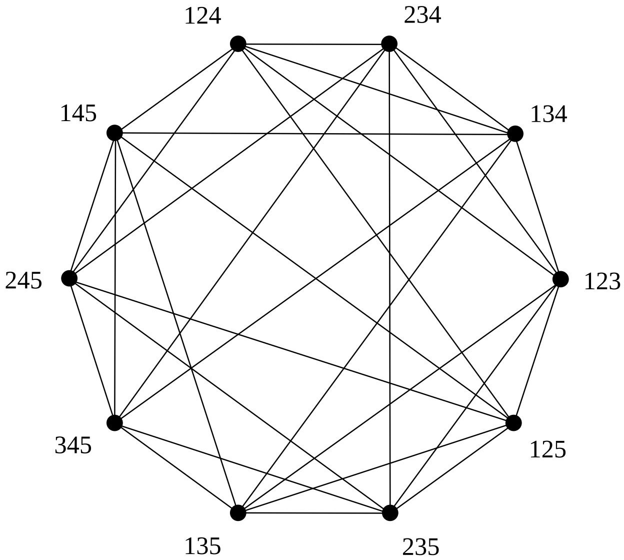
Tehtävä 1 (d,e)

Oletetaan (d)-kohdassa, että annetussa graafissa on $2n$ solmua. Graafinositusongelman hakuavaruudeksi $\mathcal{X}$ voidaan valita esimerkiksi kaikki annetun solmujoukon n-osajoukot. Tällöin kulloisenkin osituksen muodostavat tarkasteltava n-osajoukko ja sen komplementti. Vastaavasti (e)-kohdan klikkihaun tapauksessa hakuavaruudeksi $\mathcal{X}$ voidaan valita esimerkiksi kaikki solmujoukon k-osajoukot. Naapuruusto voidaan jälleen määritellä minimimuutosperiaatteella: osajoukot ovat toistensa naapureita jos ja vain jos toinen saadaan toisesta lisäämällä yksi ja poistamalla yksi alkio.

Muodostuva konfiguraatiograafi on *Johnson-graafi* $J(v,k)$: graafin solmuina ovat kaikki joukon $\{1, 2, \ldots, v\}$ k-osajoukot; kahden solmun E_1, E_2 välillä on kaari jos ja vain jos $|E_1 \cap E_2| = k - 1$.

Alla esimerkkinä Johnson-graafit $J(4, 2)$ ja $J(5, 3)$.





Tehtävä 2

Tarkastellaan ominaisuuksia graafiluokka kerrallaan.

1. Hamming-graafi $H(n, q)$

   (a) Solmujen lukumäärä on selvästi q^n.

   (b) Jokaisella solmulla on $n(q - 1)$ naapurisolmua, koska poikkeava koordinaatti voidaan valita n tavalla, ja poikkeava koordinaattiarvo $q - 1$ tavalla.

   (c) Hamming-graafi on kytketty. Mielivaltaisesta solmusta päästään mielivaltaiseen solmuun graafin kaaria pitkin muuttamalla lähtösolmun koordinaattiarvot koordinaatti kerrallaan maalisolmun koordinaattiarvoiksi.

   (d) Lyhimmän syklin pituus on 4 kun $q = 2$, ja 3 kun $q > 2$. Tämä nähdään

   (e) Pisimmän syklin pituus on q^n, ts. graafista löytyy sykli, jossa esiintyvät kaikki sen solmut. Tapauksessa $q = 2$ tällaisen syklin muodostaa esimerkiksi aiemmin tarkasteltu peilattu Gray-koodi G^n. Peilatun Graykoodin konstruktio yleistyy suoraviivaisesti tapauksiin, missä q on parillinen. Parittoman q:n tapauksessa peilauskonstruktio toimii hieman muutettuna: Olkoon $x_1, \ldots, x_N$ pituutta n oleva koodi. Muodostetaan pituutta $n+1$ oleva koodi seuraavasti:

- Toistetaan kaikille $0 \leq p \leq q-3$:
  Jos p parillinen, tulosta osakoodi $px_1, px_2, \ldots, px_N$.
  Jos p pariton, tulosta osakoodi $px_N, px_{N-1}, \ldots, px_1$.
- Tulosta osakoodi $(q-2)x_N, (q-2)x_{N-1}, \ldots, (q-2)x_2$.
- Tulosta osakoodi $(q-1)x_2, (q-1)x_3, \ldots, (q-1)x_N$.
- Tulosta osakoodi $(q-1)x_1, (q-2)x_1$.

(f) Graafin halkaisija on selvästi n, koska mistä tahansa solmusta pääsee mihin tahansa solmuun muuttamalla enintään n koordinaattipaikkaa. Toisaalta n koordinaattipaikkamuutosta tarvitaan, jotta päästäisiin esimerkiksi solmusta $00\cdots0$ solmuun $11\cdots1$.

2. Cayley-graafi $\Gamma(S_n, T_n)$

(a) Koska joukon $\{0, 1, 2, \ldots, n-1\}$ permutaatioita on $n!$ kpl, on graafissa $n!$ solmua.

(b) Joukon $\{0, 1, 2, \ldots, n-1\}$ transpositioita on $\binom{n}{2}$ kpl, joten millä tahansa graafin solmulla on yhtä monta naapurisolmua.

(c) Graafi on kytketty, koska transpositiot generoivat symmetrisen ryhmän.

(d) Tapauksessa $n = 2$ graafissa ei ole syklejä. Muutoin lyhimmän syklin pituus on enintään 4: permutaatiot

$$\pi, \quad \pi(0,1), \quad \pi(0,1)(1,2), \quad \pi(0,1)(1,2)(0,1)$$

muodostavat 4-syklin, koska $\pi(0,1)(1,2)(0,1)(0,2) = \pi$. Toisaalta graafi ei voi sisältää 3:n mittaista sykliä, koska tällöin syklin kaaria vastaavat 3 transpositiota muodostaisivat identiteettipermutaation, mikä on mahdotonta koska identiteetti on parillinen permutaatio.

(e) Pisimmän syklin pituus on $n!$, joka saadaan esim. Trotter-Johnson minimimuutosjärjestyksestä.

(f) Graafin halkaisija on $n-1$, sillä mikä tahansa permutaatio voidaan esittää enintään $n-1$ transposition tulona. Toisaalta esimerkiksi identiteetistä n-sykliin $(0, 1, 2, \ldots, n-1)$ pääsemiseksi tarvitaan ainakin $n-1$ transpositiota.

3. Johnson-graafi $J(v, k)$

(a) Solmuja on yhtä monta kuin v-joukolla k-osajoukkoja, ts. $\binom{v}{k}$ kpl.

(b) Annetusta v-joukon k-osajoukosta voidaan ottaa alkio pois k:lla tavalla, ja siihen lisätä poistetusta poikkeava alkio $v - k$ tavalla. Näin ollen kaikilla solmuilla on $k(v - k)$ naapurisolmua.

(c) Graafi on kytketty, koska on selvää että alkioita lisäämällä ja poistamalla mikä tahansa k-osajoukko voidaan muuttaa miksi tahansa toiseksi k-osajoukoksi.

(d) Tapauksissa $v = k$ tai $v = 2$ graafissa ei ole syklejä. Lyhimmän syklin pituus on selvästi 3, koska $\{x, y\}$, $\{x, z\}$, $\{y, z\}$ ovat toistensa naapureita kun $k \geq 2$. Tapaus $k = 1$ on suoraviivainen.

(e) Pisimmän syklin pituus on $\binom{v}{k}$, mikä saadaan esim. revolving door -minimimuutosjärjestyksestä (kirjan s. 48–52).

(f) Graafin halkaisija on enintään k, mikä saavutetaan kun graafista on löydettävissä kaksi solmua, jotka eivät leikkaa toisiaan. Näin on selvästi täsmälleen silloin kun $v \geq 2k$. Kun $v < 2k$, mielivaltaiset kaksi k-osajoukkoa leikkaavat ainakin $2k - v$ pisteessä. Näin ollen graafin halkaisijaksi saadaan yleisessä tapauksessa $k - \max(2k - v, 0) = \min(k, v - k)$.

Kotitehtävä 2

Euroopan unionin yhteisvaluutta euro on ollut tilivaluuttana käytössä jo jonkin aikaa, mutta eurokolikot ja setelit tulivat käyttöön vuoden 2002 alussa. Käyttöön tulivat senttiarvoisten kolikoiden lisäksi 1 ja 2 euron kolikot sekä 5, 10, 20, 50, 100, 200 ja 500 euron setelit.

Eräs hyvyyskriteeri rahojen nimellisarvovalikoimalle on, millaisia rahasummia voidaan maksaa siten, että vain pieni määrä rahoja (kolikoita tai seteleitä) vaihtaa omistajaa. Esimerkiksi 46 euroa voidaan maksaa 50 euron setelillä ja saada kaksi 2 euron kolikkoa takaisin. Se voidaan myös maksaa 50 euron setelillä ja 1 euron kolikolla ja saada 5 euron seteli takaisin.

$L(k,m)$ on suurin L, jolle voidaan löytää k nimellisarvon nimellisarvovalikoima $W=\{w_1,\ldots,w_k\}$, jolla jokainen rahasumma $1,2,3,\ldots,L$ voidaan maksaa niin, että enintään m rahaa vaihtaa omistajaa. Toisin sanoen, $L(k,m)$ on suurin sellainen L, että löytyy nimellisarvovalikoima, jolla jokainen $1,\ldots,L$ voidaan kirjoittaa muodossa $\sum_{i=1}^{k} m_i w_i$, missä $\sum_{i=1}^{k} |m_i| \leq m$. (kaikki kokonaislukuja)

- Mikä on pienin rahasumma, jonka maksamiseen eurorahoilla ei riitä, että kolme rahaa vaihtaa omistajaa?

- Millainen nimellisarvovalikoima Euroopan unionissa olisi tullut ottaa käyttöön? Laadi ei-triviaaliin heuristiikkaan (simuloitu jäähdytys, tabu-haku, geneettinen algoritmi tai vastaava) perustuva tietokoneohjelma, joka etsii mahdollisimman hyvän alarajan ja vastaavan nimellisarvovalikoiman $L(9,3)$:lle. Tarkastele myös muita tapauksia $L(k,3)$. Voit tutkia, löytääkö ohjelmasi $L(k,m)$:n tunnettuja arvoja $L(1,3) = 3$, $L(2,3) = 12$, $L(3,3) = 24$, $L(4,3) = 47$, $L(5,3) = 73$ ja $L(6,3) = 110$ vastaavia nimellisarvovalikoimia.

Erityisen ansiokkaista ratkaisuista voi saada bonuspisteen tenttiin. Bonuspisteen tavoittelemiseksi voi esimerkiksi esittää erinomaisen tehokkaan heuristiikan ja löytää tarvittavaan laskenta-aikaan suhteutettuna erinomaisia alarajoja $L(k,m)$:lle eri k:n ja m:n arvoilla tai vertailla eri lähestymistapojen tehokkuutta.

Palautukseen tulee sisältyä ratkaisun sanallinen kuvaus (esim. tekstinä tai hyvälaatuisena Postscriptinä), josta käy ilmi löydetyt $L(k,m)$:n alarajat vastaavine nimellisarvojoukkoineen sekä tarvittu laskenta-aika. Palautuksen liitteenä tulee olla käytettyjen tietokoneohjelmien lähdekoodi.

Palautus: Palautus assistentille (Petteri Kaski, Petteri.Kaski@hut.fi, TB358) sähköpostitse esim. tekstinä tai hyvälaatuisena Postscriptinä tiistaihin 26.3.2002 mennessä.

Laskuharjoitus 7

14.3.2002

1. Kuinka monella toisistaan poikkeavalla tavalla voidaan numeroida kuution kuusi sivutahkoa, kun jokaiselle tahkolle annetaan oma numero? (Kuutio oletetaan täysin symmetriseksi, ja numeroiden asennolla sivutahkossa ei ole merkitystä.) Entäpä dodekaedrin 12 sivutahkoa?

2. Kuinka monella tavalla 19-joukko voidaan osittaa yhteen 4-alkioiseen osajoukkoon, ja kolmeen 5-alkioiseen osajoukkoon? Yksi tällainen ositus on esimerkiksi

$$\{\{1,2,3,4,5\},\{6,7,8,9,10\},\{11,12,13,14,15\},\{16,17,18,19\}\}.$$

3. Kirjan tehtävä 6.7: Tutkitaan joukon $\{0,1,\ldots,9\}$ permutaatioryhmää G, jonka Schreier-Sims-esitys on

$$\overrightarrow{G} = (\beta : [\mathcal{U}_0, \mathcal{U}_1, \ldots, \mathcal{U}_9]) ,$$

missä $\beta = \mathbf{I}$ ja

$$\mathcal{U}_0 = \left\{ \begin{array}{l} \mathbf{I},\ (0,1,3,6)\,(2,5,9,7)\,(4,8)\,,\ (0,2,5,9,6)\,(1,4,8,3,7)\,, \\ (0,3)\,(1,6)\,(2,9)\,(5,7)\,,\ (0,4,9,6)\,(1,2,5,8)\,(3,7)\,, \\ (0,5,6,2,9)\,(1,8,7,4,3)\,,\ (0,6,9,5,2)\,(1,7,3,8,4)\,, \\ (0,7,6)\,(1,2,8)\,(3,4,9)\,,\ (0,8,4,1,9)\,(2,5,3,6,7)\,, \\ (0,9,2,6,5)\,(1,3,4,7,8) \end{array} \right\},$$

$$\mathcal{U}_1 = \left\{ \begin{array}{l} \mathbf{I},\ (1,2)\,(3,4)\,(6,7)\,,\ (1,3,6)\,(2,4,7)\,(5,9,8)\,, \\ (1,4)\,(2,3)\,(6,7)\,(8,9)\,,\ (1,6)\,(2,7)\,(5,9)\,, \\ (1,7,3,2,6,4)\,(5,8,9) \end{array} \right\},$$

$$\mathcal{U}_2 = \{\mathbf{I}\}\,, \qquad \mathcal{U}_3 = \{\mathbf{I},\ (3,6)\,(4,7)\,(5,8)\}\,,$$

$$\mathcal{U}_4 = \mathcal{U}_5 = \mathcal{U}_6 = \mathcal{U}_7 = \mathcal{U}_8 = \mathcal{U}_9 = \{\mathbf{I}\}\,.$$

Mitkä seuraavista permutaatioista kuuluvat ryhmään G?

(a) $\alpha = (0,1,2,3,4,5,6)$

(b) β=$(0,1,2,3,4)\,(5,6,7,8,9)$

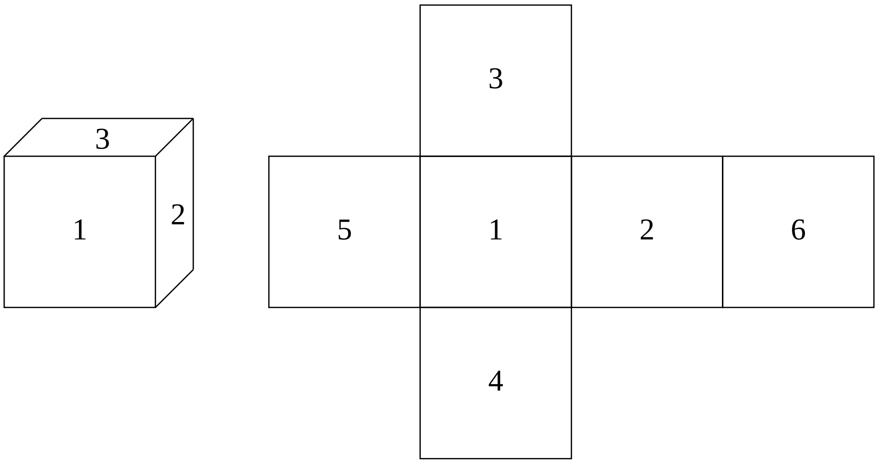
(c) γ=$(0,3,5,8,7)\,(1,9,2,6,4)$

Mikä on ryhmän G koko?

Laskuharjoituksen 7 vastaukset

Tehtävä 1

Nimetään aluksi kuution kuusi sivutahkoa jollakin tavalla (esim. kansi, pohja, etu, taka, vasen, oikea); käytetään tässä tapauksessa kuitenkin kokonaislukuja 1, 2, 3, 4, 5, 6 kuten alla:



Tällöin mikä tahansa kuution sivujen numerointi kokonaisluvuilla 1, 2, 3, 4, 5, 6 voidaan esittää joukon $\{1, 2, 3, 4, 5, 6\}$ permutaationa π: sivutahkoon i kiinnitetty numero on $\pi(i)$ kaikilla $i = 1, 2, 3, 4, 5, 6$.

Kuution symmetrian vuoksi tällöin oleellisesti sama kuutio esiintyy kuitenkin useampaan kertaan. Esimerkiksi

$$\pi(1) = 1, \quad \pi(2) = 2, \quad \pi(3) = 3, \quad \pi(4) = 4, \quad \pi(5) = 5, \quad \pi(6) = 6$$

on sama kuutio kuin

$$\pi'(1) = 2, \quad \pi'(2) = 6, \quad \pi'(3) = 3, \quad \pi'(4) = 4, \quad \pi'(5) = 1, \quad \pi'(6) = 5,$$

koska π' saadaan π:sta kiertämällä kuutiota sivujen 3 ja 4 keskipisteen läpi kulkevan akselin ympäri 90 astetta. Tämä vastaa sivutahkojen nimien permutaatiota $\gamma = (1, 5, 6, 2)$. (Sivutahko i kiertyy sivutahkoksi $\gamma(i)$ kaikilla $i = 1, 2, 3, 4, 5, 6$.)

Permutaatiot $(1,5,6,2)$ ja $(1,4,6,3)$ generoivat kuution sivutahkojen kaikki $6 \cdot 4 = 24$ kiertosymmetriaa:

$$G = \left\{ \begin{array}{lll} \mathbf{I}, & (2,3,5,4), & (2,4,5,3), \\ (2,5)(3,4), & (1,2)(3,4)(5,6), & (1,2,3)(4,6,5), \\ (1,2,4)(3,6,5), & (1,2,6,5), & (1,3,2)(4,5,6), \\ (1,3,6,4), & (1,3)(2,5)(4,6), & (1,3,5)(2,6,4), \\ (1,4,2)(3,5,6), & (1,4,6,3), & (1,4)(2,5)(3,6), \\ (1,4,5)(2,6,3), & (1,5,6,2), & (1,5,4)(2,3,6), \\ (1,5,3)(2,4,6), & (1,5)(2,6)(3,4), & (1,6)(3,4), \\ (1,6)(2,3)(4,5), & (1,6)(2,4)(3,5), & (1,6)(2,5) \end{array} \right\}.$$

Kuution numeroinnit π ja π' ovat nyt symmetriaa vaille samat jos ja vain jos on olemassa $\gamma \in G$ siten, että $\pi' = \pi\gamma^{-1}$. Edellinen on ekvivalentti ehdon $\pi^{-1}\pi' \in G$ kanssa, joten saadaan, että numeroinnit π ja π' ovat samat jos ja vain jos ne kuuluvat samaan G:n vasempaan sivuluokkaan symmetrisessä ryhmässä $\mathsf{Sym}(\{1,2,3,4,5,6\})$.

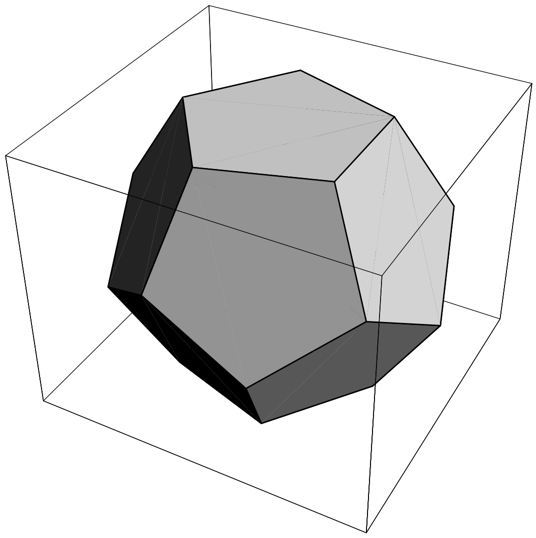
Ryhmän G sivuluokkien lukumääräksi ryhmässä $\mathsf{Sym}(\{1,2,3,4,5,6\})$ saadaan Lagrangen lauseen (kirja s. 193, Lause 6.2) perusteella

$$\frac{|\mathsf{Sym}(\{1,2,3,4,5,6\})|}{|G|} = \frac{6!}{24} = \frac{720}{24} = 30.$$

Sivuluokkien jokin vasen transversaali voidaan määrittää esimerkiksi kirjan algoritmilla 6.17 (s. 224), joka laskee juuri vasemman transversaalin; tätä ei erikseen mainita kirjassa.

$$\begin{array}{lll} [1,2,3,4,5,6], & [1,2,3,4,6,5], & [1,2,3,5,4,6], \\ [1,2,3,6,4,5], & [1,2,3,5,6,4], & [1,2,3,6,5,4], \\ [1,2,4,3,5,6], & [1,2,4,3,6,5], & [1,2,5,3,4,6], \\ [1,2,6,3,4,5], & [1,2,5,3,6,4], & [1,2,6,3,5,4], \\ [1,2,4,5,3,6], & [1,2,4,6,3,5], & [1,2,5,4,3,6], \\ [1,2,6,4,3,5], & [1,2,5,6,3,4], & [1,2,6,5,3,4], \\ [1,2,4,5,6,3], & [1,2,4,6,5,3], & [1,2,5,4,6,3], \\ [1,2,6,4,5,3], & [1,2,5,6,4,3], & [1,2,6,5,4,3], \\ [1,6,2,3,4,5], & [1,6,2,3,5,4], & [1,6,2,4,3,5], \\ [1,6,2,5,3,4], & [1,6,2,4,5,3], & [1,6,2,5,4,3]. \end{array}$$

Dodekaedri koostuu 12 5-kulmiosta (kuva alla), ja sen kiertosymmetriaryhmän koko on $12 \cdot 5 = 60$. (Kiinnitetään jokin dodekaedrin sivutahko ja sen naapuritahko. Mikä tahansa 12 sivutahkosta voidaan kiertää kiinnitetyn tahkon paikalle, jonka jälkeen naapuritahko voidaan valita 5 tavalla. Kaksi kiinnitettyä tahkoa määräävät yksikäsitteisesti käytetyn kierron.)



Dodekaedrille erilaisten numerointien määräksi saadaan siis

$$\frac{12!}{60} = \frac{479001600}{60} = 7983360.$$

Tehtävä 2

Olkoon P joukon $X = \{1, 2, \ldots, 19\}$ ositus

$$\{\{1, 2, 3, 4, 5\}, \{6, 7, 8, 9, 10\}, \{11, 12, 13, 14, 15\}, \{16, 17, 18, 19\}\}.$$

Kaikki joukon X annetut ehdot täyttävät ositukset voidaan selvästi muodostaa osituksesta P nimeämällä siinä esiintyvät pisteet uudestaan. Toisin sanoen, kaikkien ehdot täyttävien ositusten joukko on osituksen P rata ryhmän $G = \mathsf{Sym}(X)$ suhteen, missä permutaatio π operoi ositukseen P nimeämällä sen pisteet uudelleen. Esimerkiksi

$$P' = \{\{1, 3, 4, 5, 7\}, \{2, 6, 8, 10, 12\}, \{9, 11, 13, 14, 15\}, \{16, 17, 18, 19\}\}$$

saadaan P:stä vaihtamalla pisteet 2 ja 7 sekä 9 ja 12 keskenään, ts. $P' = \pi(P)$, missä $\pi = (2, 7)(9, 12)$.

Rata-stabiloijalauseen (kirja s. 213, Lemma 6.9) perusteella saadaan radan $G(P)$ pituudeksi

$$|G(P)| = \frac{|G|}{|G_P|},$$

missä G_P on se G:n aliryhmä, joka koostuu kaikista permutaatioista jotka stabiloivat osituksen P, ts.

$$G_P = \{\pi \in G \,:\, \pi(P) = P\}.$$

Nyt $|G_P| = 3!(5!)^3 4!$, koska osituksen P 4- ja 5-osajoukkojen sisältöjä saa permutoida mielivaltaisesti osituksen P muuttumatta; lisäksi 5-osajoukot voi järjestää keskenään 3! tavalla. Erilaisten osituksien lukumääräksi saadaan siis

$$|G(P)| = \frac{19!}{3!(5!)^3 4!} = \frac{121645100408832000}{248832000} = 488864376.$$

Tehtävä 3

(a) Koska α kuvaa alkion 0 alkiolle 1, etsitään $h_0 \in \mathcal{U}_0$, joka kuvaa alkion 0 alkiolle 1. Sellainen löytyy: $h_0 = (0,1,3,6)(2,5,9,7)(4,8)$. Kertomalla α tämän käänteispermutaatiolla saadaan $h_0^{-1}\alpha = (0)(1,7,9,5,3,8,4,2)(6)$. Koska tässä 1 kuvautuu 7:lle, etsitään $h_1 \in \mathcal{U}_1$, joka kuvaa 1:n 7:lle. Sellainen löytyy: $h_1 = (1,7,3,2,6,4)(5,8,9)$. Nyt $h_1^{-1}h_0^{-1}\alpha = (0)(1)(2,4,3,5,7,8,6)(9)$. Koska tässä 2 kuvautuu 4:lle, etsitään $h_2 \in \mathcal{U}_2$, joka kuvaa 2:n 4:lle. Sellaista ei löydy, joten $\alpha \notin G$.

(b) Koska β kuvaa alkion 0 alkiolle 1, etsitään $h_0 \in \mathcal{U}_0$, joka kuvaa alkion 0 alkiolle 1. Sellainen löytyy: $h_0 = (0,1,3,6)(2,5,9,7)(4,8)$. Kertomalla β tämän käänteispermutaatiolla saadaan $h_0^{-1}\beta = (0)(1,7,4,6,9,2)(3,8,5)$. Koska tässä 1 kuvautuu 7:lle, etsitään $h_1 \in \mathcal{U}_1$, joka kuvaa 1:n 7:lle. Sellainen löytyy: $h_1 = (1,7,3,2,6,4)(5,8,9)$. Nyt $h_1^{-1}h_0^{-1}\beta = (0)(1)(2,4)(3,5,7,6,8,9)$. Koska tässä 2 kuvautuu 4:lle, etsitään $h_2 \in \mathcal{U}_2$, joka kuvaa 2:n 4:lle. Sellaista ei löydy, joten $\beta \notin G$.

(c) Koska γ kuvaa alkion 0 alkiolle 3, etsitään $h_0 \in \mathcal{U}_0$, joka kuvaa alkion 0 alkiolle 3. Sellainen löytyy: $h_0 = (0,3)(1,6)(2,9)(5,7)$. Kertomalla γ tämän käänteispermutaatiolla saadaan $h_0^{-1}\gamma = (0)(1,2)(3,7)(4,6)(5,8)(9)$. Koska tässä 1 kuvautuu 2:lle, etsitään $h_1 \in \mathcal{U}_1$, joka kuvaa 1:n 2:lle. Sellainen löytyy: $h_1 = (1,2)(3,4)(6,7)$. Nyt $h_1^{-1}h_0^{-1}\gamma = (0)(1)(2)(3,6)(4,7)(5,8)(9)$. Tässä 2 kuvautuu 2:lle; etsitään $h_2 \in \mathcal{U}_2$, joka kuvaa 2:n 2:lle. Sellainen löytyy: $h_2 = \mathbf{I}$. Nyt $h_2^{-1}h_1^{-1}h_0^{-1}\gamma = (0)(1)(2)(3,6)(4,7)(5,8)(9)$. Koska tässä

3 kuvautuu 6:lle, etsitään $h_3 \in \mathcal{U}_3$, joka kuvaa 3:n 6:lle. Sellainen löytyy: $h_3 = (3,6)(4,7)(5,8)$. Nyt $h_3^{-1}h_2^{-1}h_1^{-1}h_0^{-1}\gamma = \mathbf{I}$. Tämä kuvaa kaikki alkiot itselleen, ja $\mathbf{I} \in \mathcal{U}_n$ kaikille $n \geq 4$, joten $\gamma \in G$.

Laskuharjoitus 8

21.3.2002

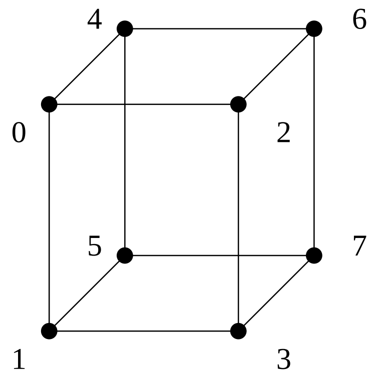
1. Neliö jaetaan 9 pikkuneliöön, joista osa väritetään mustiksi. Kaksi väritystä samaistetaan, jos toinen saadaan ensimmäisestä kiertämällä ja/tai peilaamalla koko neliötä.

0	1	2
3	4	5
6	7	8

   (a) Kuinka monella eri tavalla voidaan värittää 5 pikkuneliötä?

   (b) Anna kaikki kahdeksan erilaista 2 pikkuneliön väritystä.

2. Olkoon $\mathcal{G}$ oheinen graafi.



Määritä osajoukkojen $\{0, 7\}$ ja $\{0, 1, 2, 3\}$ radat ja stabiloija-aliryhmät ryhmän $\mathsf{Aut}(\mathcal{G})$ suhteen. Määritä lisäksi molempien stabiloija-aliryhmien jokin vasen transversaali ryhmän $\mathsf{Aut}(\mathcal{G})$ suhteen. (Ryhmän $\mathsf{Aut}(\mathcal{G})$ alkiot on listattu kirjan taulukossa 6.1, ja Schreier-Sims esitys esimerkissä 6.7.)

Laskuharjoituksen 8 vastaukset

Tehtävä 1 (a)

Annetun neliön symmetriaryhmä G on isomorfinen dihedraaliryhmän D_8 kanssa. Ryhmän muodostavat permutaatiot

$$\begin{aligned}
g_0 &= (0)(1)(2)(3)(4)(5)(6)(7)(8)\\
g_1 &= (0,2,8,6)(1,5,7,3)(4)\\
g_2 &= (0,8)(1,7)(2,6)(5,3)(4)\\
g_3 &= (0,6,8,2)(1,3,7,5)(4)\\
g_4 &= (0,2)(3,5)(6,8)(1)(4)(7)\\
g_5 &= (1,5)(0,8)(3,7)(2)(4)(6)\\
g_6 &= (0,6)(1,7)(2,8)(3)(4)(5)\\
g_7 &= (1,3)(2,6)(5,7)(0)(4)(8).
\end{aligned}$$

Permutaatioiden tyypit ovat

$$\begin{aligned}
\mathrm{type}(g_0) &= (9,0,0,0,0,0,0,0,0)\\
\mathrm{type}(g_{1,3}) &= (1,0,0,2,0,0,0,0,0)\\
\mathrm{type}(g_2) &= (1,4,0,0,0,0,0,0,0)\\
\mathrm{type}(g_{4,5,6,7}) &= (3,3,0,0,0,0,0,0,0).
\end{aligned}$$

Montako 5-osajoukkoa kukin permutaatio kuvaa itselleen? Jos permutaation minkä tahansa syklin jokin alkio kuuluu osajoukkoon, kaikkien ko. syklin alkioiden on kuuluttava osajoukkoon, jotta permutaatio kuvaisi osajoukon itselleen. Lasketaan siis, montako tapaa on valita permutaation syklejä siten, että syklien yhteenlaskettu pituus on 5.

$$\begin{aligned}
\chi_5(g_0) &= \binom{9}{5} = 126\\
\chi_5(g_{1,3}) &= \binom{2}{1} = 2\\
\chi_5(g_2) &= \binom{4}{2} = 6\\
\chi_5(g_{4,5,6,7}) &= \binom{3}{1}\binom{3}{2} + \binom{3}{3}\binom{3}{1} = 12
\end{aligned}$$

Burnsiden lemmalla (kirjan s. 215, Lause 6.10)

$$N_k = \frac{1}{|G|}\sum_{g\in G}\chi_k(g)$$

saadaan

$$N_5 = \frac{1}{8}(126 + 2 \cdot 2 + 6 + 4 \cdot 12) = 23.$$

Tehtävä 1 (b)

Muodostetaan k-osajoukkojen ratojen edustajajoukot R_k, kun $0 \leq k \leq 2$. Joukko R_2 tulee olemaan vastaus tehtävään.

$R_0 = \{\emptyset\}$, sillä tyhjä joukko on ainoa 9 alkion osajoukko, jonka koko on 0. Lisätään nyt vuorollaan kuhunkin R_0:n alkioon kukin alkioista $\{0, \ldots, 8\}$ ja lisätään syntynyt tulos joukkoon R_1, ellei joukossa R_1 ole jo leksikografisessa järjestyksessä aikaisemmin tullutta 1-osajoukkoa samalta radalta.

$$R_1 = \{\{0\}, \{1\}, \overbrace{\{2\}}^{\{0\}}, \overbrace{\{3\}}^{\{1\}}, \{4\}, \overbrace{\{5\}}^{\{1\}}, \overbrace{\{6\}}^{\{0\}}, \overbrace{\{7\}}^{\{1\}}, \overbrace{\{8\}}^{\{0\}}\} = \{\{0\}, \{1\}, \{4\}\}$$

Otetaan nyt vuorollaan kukin R_1:n joukoista ja lisätään siihen vuorollaan kukin alkio, joka on suurinta joukon alkiota suurempi. Lisätään syntynyt joukko R_2:een, ellei siellä jo ole aikaisempaa 2-osajoukkoa samalta radalta.

$$\begin{aligned} R_2 = \{ \quad & \{0,1\}, \{0,2\}, \overbrace{\{0,3\}}^{\{0,1\}}, \{0,4\}, \{0,5\}, \overbrace{\{0,6\}}^{\{0,2\}}, \overbrace{\{0,7\}}^{\{0,5\}}, \{0,8\}, \\ & \overbrace{\{1,2\}}^{\{0,1\}}, \{1,3\}, \{1,4\}, \overbrace{\{1,5\}}^{\{1,3\}}, \overbrace{\{1,6\}}^{\{0,5\}}, \{1,7\}, \overbrace{\{1,8\}}^{\{0,5\}}, \\ & \overbrace{\{4,5\}}^{\{1,4\}}, \overbrace{\{4,6\}}^{\{0,6\}}, \overbrace{\{4,7\}}^{\{1,4\}}, \overbrace{\{4,8\}}^{\{0,4\}} \} \end{aligned}$$

ja jäljelle jää siis

$$R_2 = \{\{0,1\}, \{0,2\}, \{0,4\}, \{0,5\}, \{0,8\}, \{1,3\}, \{1,4\}, \{1,7\}\}.$$

Huomautus: Yllä R_{k+1}:ä konstruoitaessa riittää lisätä edellisen R_k:n joukkoihin vuorollaan aiempia suuremmat alkiot, sillä tässä algoritmin versiossa kunkin radan edustajana on aina radan leksikografisessa järjestyksessä pienin alkio. Jos $(k+1)$-osajoukko $S \in R_{k+1}$ on ratansa leksikografisessa järjestyksessä pienin alkio, pätee, että joukko $S' = S \setminus \{\max(S)\}$ on myös ratansa pienin alkio, ts. $S' \in R_k$. Tästä seuraa, että jokainen R_{k+1}:n alkio voidaan konstruoida jostakin R_k:n alkiosta lisäämällä siihen aiempia suurempi alkio.

Tehtävä 2

Graafin $\mathcal{G}$ automorfismiryhmässä $\mathsf{Aut}(\mathcal{G})$ on 48 alkiota:

$\mathbf{I}$	$(0,2,3,1)(4,6,7,5)$	$(0,5,3)(2,4,7)$
$(2,4)(3,5)$	$(0,2,6,7,5,1)(3,4)$	$(0,5)(2,7)$
$(1,2)(5,6)$	$(0,3,6)(1,7,4)$	$(0,5,3,6)(1,7,2,4)$
$(1,2,4)(3,6,5)$	$(0,3,5,6)(1,7,4,2)$	$(0,5,6)(1,7,2)$
$(1,4,2)(3,5,6)$	$(0,3,6,5)(1,2,7,4)$	$(0,6,3)(1,4,7)$
$(1,4)(3,6)$	$(0,3)(1,2)(4,7)(5,6)$	$(0,6,3,5)(1,4,2,7)$
$(0,1,3,7,6,4)(2,5)$	$(0,3,5)(2,7,4)$	$(0,6,5,3)(1,2,4,7)$
$(0,1,3,2)(4,5,7,6)$	$(0,3)(4,7)$	$(0,6,5)(1,2,7)$
$(0,1,5,4)(2,3,7,6)$	$(0,4,6,7,3,1)(2,5)$	$(0,6)(1,7)(2,4)(3,5)$
$(0,1,5,7,6,2)(3,4)$	$(0,4,5,1)(2,6,7,3)$	$(0,6)(1,7)$
$(0,1)(2,3)(4,5)(6,7)$	$(0,4,6,2)(1,5,7,3)$	$(0,7)(1,6)(2,5)(3,4)$
$(0,1)(2,5)(3,4)(6,7)$	$(0,4)(1,5)(2,6)(3,7)$	$(0,7)(1,6)(2,3)(4,5)$
$(0,2,6,4)(1,3,7,5)$	$(0,4,5,7,3,2)(1,6)$	$(0,7)(1,5)(2,6)(3,4)$
$(0,2)(1,3)(4,6)(5,7)$	$(0,4)(1,6)(2,5)(3,7)$	$(0,7)(1,5,4,6,2,3)$
$(0,2,3,7,5,4)(1,6)$	$(0,5,6,3)(1,4,7,2)$	$(0,7)(1,3,2,6,4,5)$
$(0,2)(1,6)(3,4)(5,7)$	$(0,5)(1,4)(2,7)(3,6)$	$(0,7)(1,3)(2,5)(4,5)$

Ryhmän generoivat permutaatiot $(0,1,3,7,6,4)(2,5)$ ja $(0,1,3,2)(4,5,7,6)$. Stabiloija-aliryhmät voi määrittää joko algoritmisesti ensin generoimalla ryhmälle Schreier-Sims esityksen (huomaa, että kirjan algoritmissa 6.9 on painovirhe; errata löytyy osoitteesta `http://www.math.mtu.edu/~kreher/cages.html`) ja tämän jälkeen käymällä algoritmin 6.6 avulla ryhmän kaikki alkiot läpi ja tulostamalla ne, jotka kuvaavat joukot $\{0,7\}$ ja $\{0,1,2,3\}$ itselleen. Toisaalta ainakin joukon $\{0,7\}$ stabiloijat

$\mathbf{I}$	$(1,4,2)(3,5,6)$	$(0,7)(1,5)(2,6)(3,4)$
$(2,4)(3,5)$	$(1,4)(3,6)$	$(0,7)(1,5,4,6,2,3)$
$(1,2)(5,6)$	$(0,7)(1,6)(2,5)(3,4)$	$(0,7)(1,3,2,6,4,5)$
$(1,2,4)(3,6,5)$	$(0,7)(1,6)(2,3)(4,5)$	$(0,7)(1,3)(2,5)(4,5)$

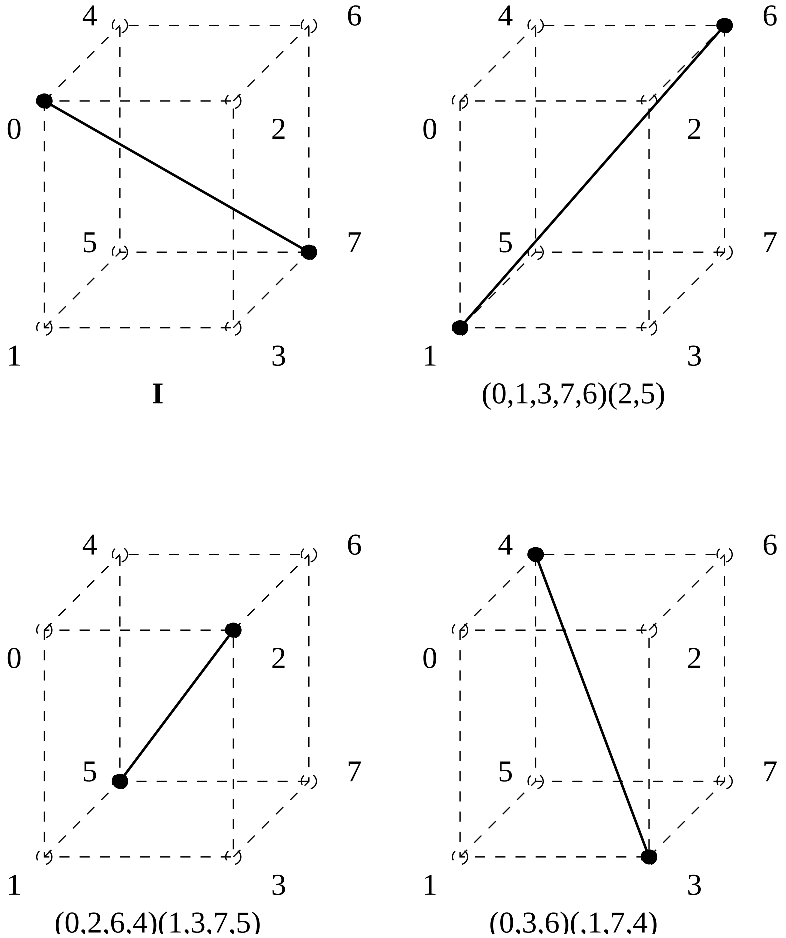
on helppo poimia kaikkien ryhmän permutaatioiden listasta suoraan. Stabiloija-aliryhmän $\mathsf{Aut}(\mathcal{G})_{\{0,7\}}$ koko on siis 12, joten joukon $\{0,7\}$ radan ja toisaalta stabiloijan vasemman transversaalin koko on

$$|\mathsf{Aut}(\mathcal{G})|/|\mathsf{Aut}(\mathcal{G})_{\{0,7\}}| = 48/12 = 4.$$

Vasemman transversaalin voi määrittää esim. kirjan algoritmilla 6.17, joka laskee juuri vasemman transversaalin; tätä ei erikseen mainita kirjan tekstissä. Erääksi vasemmaksi transversaaliksi saadaan

$$\begin{array}{ll} g_1 = \mathbf{I} & g_3 = (0,2,6,4)(1,3,7,5) \\ g_2 = (0,1,3,7,6,4)(2,5) & g_4 = (0,36)(1,7,4) \end{array}.$$

Osajoukon $\{0,7\}$ rata ja vastaava transversaali on esitetty alla.



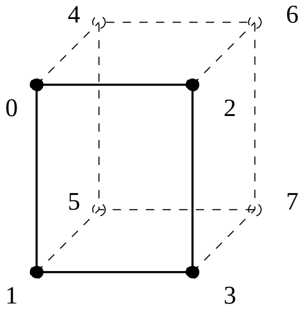
Vastaavasti osajoukon $\{0,1,2,3\}$ stabiloija-aliryhmä $\mathsf{Aut}(\mathcal{G})_{\{0,1,2,3\}}$ koostuu permutaatioista

$$\begin{array}{ll} \mathbf{I} & (0,2)(1,3)(4,6)(5,7) \\ (1,2)(5,6) & (0,2,3,1)(4,6,7,5) \\ (0,1,3,2)(4,5,7,6) & (0,3)(1,2)(4,7)(5,6) \\ (0,1)(2,3)(4,5)(6,7) & (0,3)(4,7) \end{array},$$

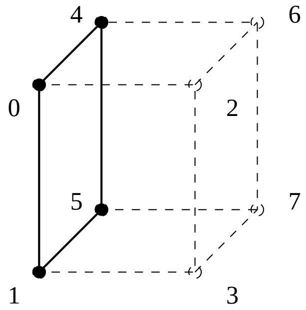
joten osajoukon $\{0,1,2,3\}$ radan pituus ja toisaalta $\mathsf{Aut}(\mathcal{G})_{\{0,1,2,3\}}$:n vasemman transversaalin koko on $48/8 = 6$. Eräs vasen transversaali koostuu permutaatioista

$$\begin{array}{lll} g_1 = \mathbf{I} & g_3 = (1,2,4)(3,6,5) & g_5 = (0,2,6,4)(1,3,7,5) \\ g_2 = (2,4)(5,3) & g_4 = (0,1,3,7,6,4)(2,5) & g_6 = (0,4)(1,5)(2,6)(3,7) \end{array}.$$

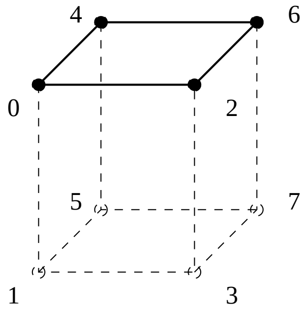
Osajoukon rata ja vastaavat transversaalin alkiot on esitetty alla.



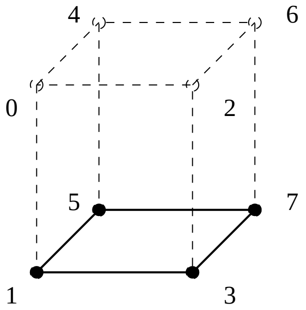
I



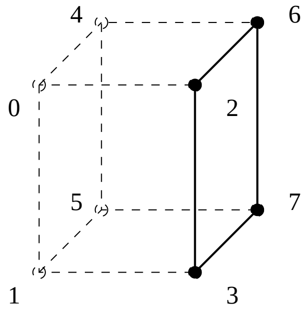
(2,4)(5,3)



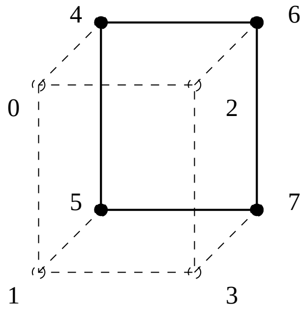
(1,2,4)(3,6,5)



(0,1,3,7,6,4)(2,5)



(0,2,6,4)(1,3,7,5)

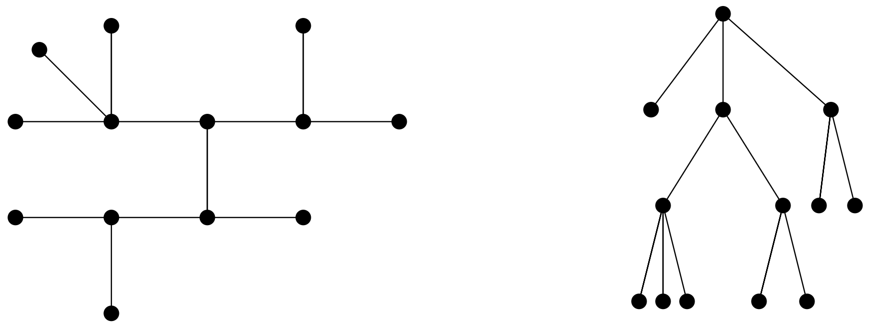


(0,4)(1,5)(2,6)(3,7)

Laskuharjoitus 9

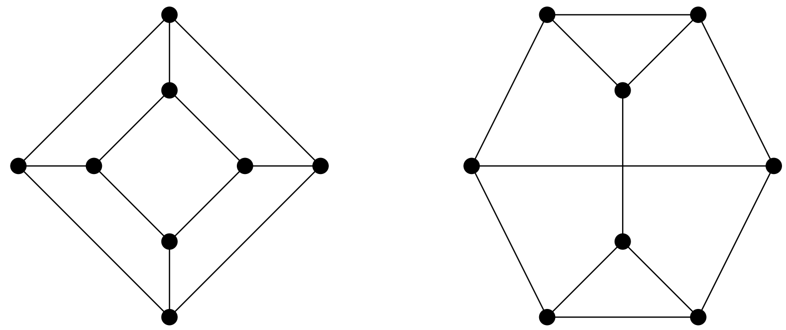
4.4.2002

1. Tutki ovatko oheiset puut isomorfisia laskemalla niille sertifikaatit.

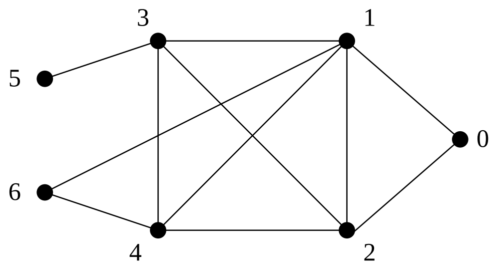


2. Muunna edellisen tehtävän tuloksena saatu sertifikaatti takaisin puuksi.

3. Osoita sopivan invariantin avulla, että allaolevat graafit eivät ole isomorfisia.



4. Osita allaolevan graafin solmujoukko solmujen astelukujen mukaan, ja muodosta osituksen karkein tasasuhtainen hienonnus (coarsest equitable refinement).



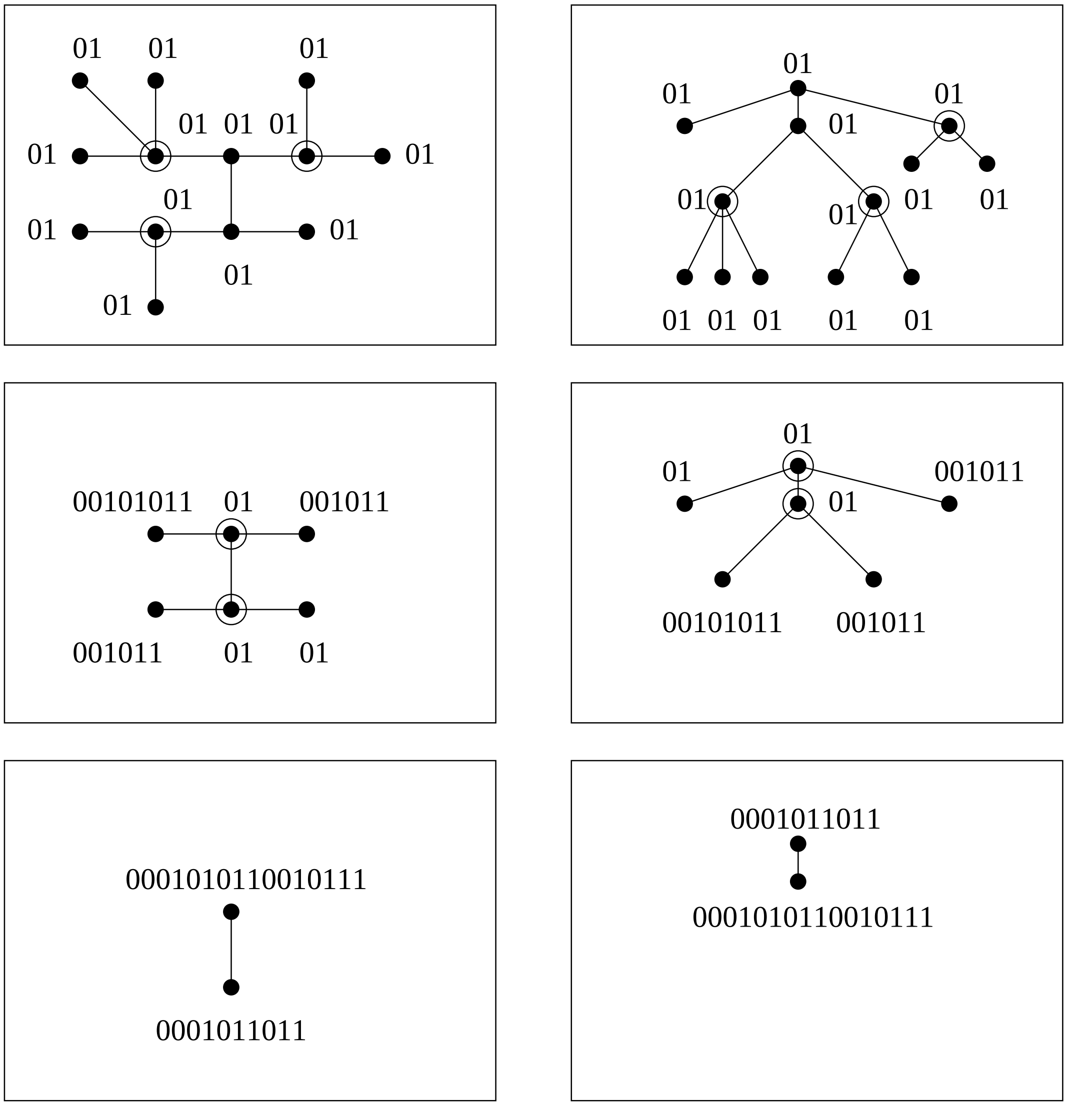
Laskuharjoituksen 9 vastaukset

Tehtävä 1

Käytetään kirjan kappaleessa 7.3.1. esitettyä algoritmia hieman tarkennettuna.

1. Nimeä puussa kaikki solmut 01:ksi.
2. Niin kauan kuin puussa on enemmän kuin kaksi solmua, toista seuraavaa.
   (a) Määritä kaikki ne ei-lehtisolmut, joilla on naapurina enintään yksi ei-lehtisolmu, ja aseta nämä solmut joukkoon T. (Puun solmu on lehtisolmu jos sillä on enintään yksi naapurisolmu.)
   (b) Tee kullekin $x \in T$ seuraavaa:
      i. Kerää joukkoon Y x:n naapurina olevien lehtisolmujen nimet sekä solmun x nimi, josta on poistettu alku-0 ja loppu-1.
      ii. Korvaa x:n nimi nimellä, joka saadaan järjestämällä Y:n alkiot leksikografiseen järjestykseen, liittämällä ne yhdeksi bittijonoksi, ja lisäämällä alkuun 0 ja loppuun 1.
      iii. Poista x:n naapurina olevat lehtisolmut.
3. Jos jäljellä on vain yksi solmu, sen nimi on puun sertifikaatti.
4. Jos jäljellä on kaksi solmua, niiden nimet leksikografisessa järjestyksessä yhteenliitettyinä muodostavat puun sertifikaatin.

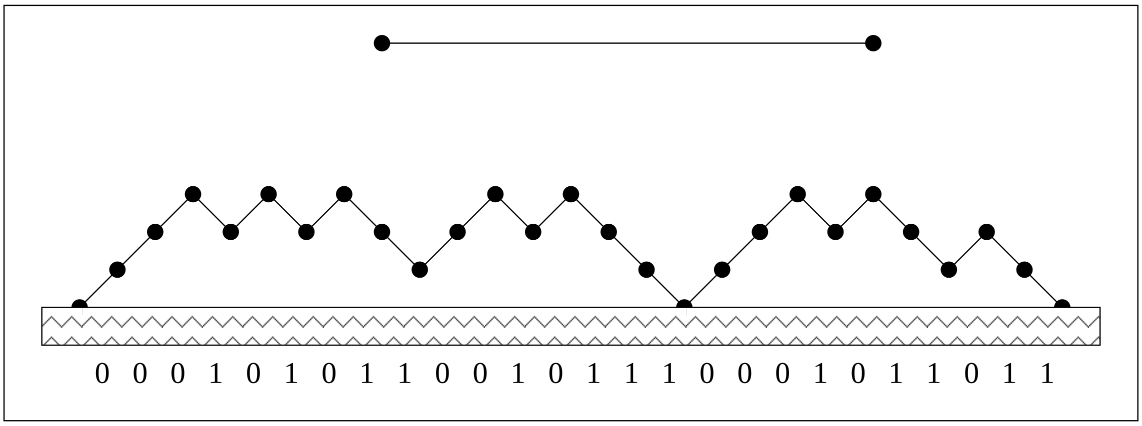
Algoritmin eteneminen tehtävässä annetuille puille on esitetty seuraavassa kuvassa. Ympyröidyt solmut ilmoittavat kulloinkin joukkoon T kuuluvat solmut.

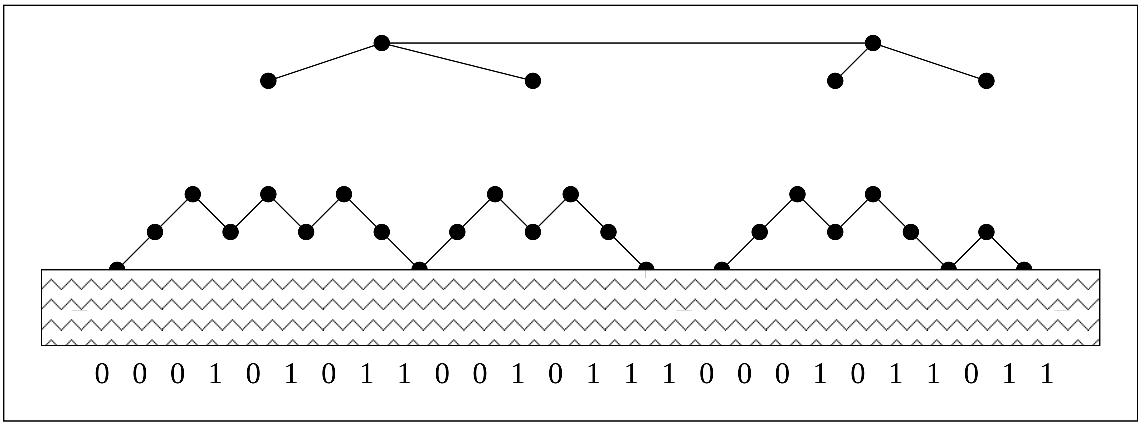


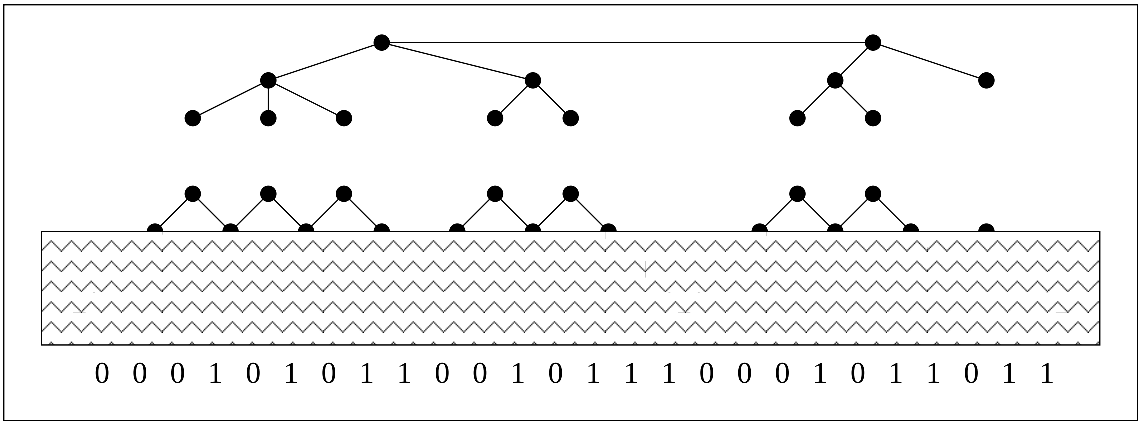
Tuloksena saadaan molemmille puille sama sertifikaatti, joten puut ovat isomorfiset.

Tehtävä 2

Käytetään kirjassa esitettyä nousevan veden algoritmia (s. 248–252). Algoritmin kulku on esitetty seuraavassa kuvassa.







Tehtävä 3

Graafit $\mathcal{G}_1 = (\mathcal{V}_1, \mathcal{E}_1)$ ja $\mathcal{G}_2 = (\mathcal{V}_2, \mathcal{E}_2)$ ovat *isomorfisia* (merkintä: $\mathcal{G}_1 \cong \mathcal{G}_2$) jos on olemassa bijektio $\pi : \mathcal{V}_1 \to \mathcal{V}_2$ siten, että kaikilla $u, v \in \mathcal{V}_1$ pätee $\{u, v\} \in \mathcal{E}_1$

jos ja vain jos $\{\pi(u), \pi(v)\} \in \mathcal{E}_2$. Bijektio π on *isomorfismi*.

Olkoon $\mathcal{F}$ perhe graafeja ja X jokin joukko. Kuvaus $\Phi : \mathcal{F} \to X$ on *invariantti* perheelle $\mathcal{F}$ jos kaikille $\mathcal{G}_1, \mathcal{G}_2 \in \mathcal{F}$ pätee $\Phi(\mathcal{G}_1) = \Phi(\mathcal{G}_2)$ aina kun $\mathcal{G}_1 \cong \mathcal{G}_2$. Näin ollen jos $\Phi(\mathcal{G}_1) \neq \Phi(\mathcal{G}_2)$ niin on oltava $\mathcal{G}_1 \ncong \mathcal{G}_2$.

Tehtävän ratkaisemiseksi riittää siis löytää invariantti, joka erottelee annetut graafit toisistaan. Tällainen on esimerkiksi kolmioiden lukumäärä graafissa: isomorfismin π on kuvattava mielivaltainen kolmio $\{\{u, v\}, \{u, w\}, \{v, w\}\} \subseteq \mathcal{E}_1$ vastaavalle kolmiolle $\{\{\pi(u), \pi(v)\}, \{\pi(u), \pi(w)\}, \{\pi(v), \pi(w)\}\} \subseteq \mathcal{E}_2$, joten isomorfisissa graafeissa $\mathcal{G}_1, \mathcal{G}_2$ on oltava sama lukumäärä kolmioita. Nyt huomataan, että vasemmanpuoleisessa graafissa ei ole yhtään kolmiota, ja oikeanpuoleisessa graafissa on kaksi kolmiota. Näin ollen graafit eivät voi olla isomorfiset.

Tehtävä 4

Olkoon $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ tarkasteltava graafi. Merkitään solmun $v \in \mathcal{V}$ naapurisolmujen joukkoa $N_{\mathcal{G}}(v)$:llä. Määritellään osajoukolle $T \subseteq \mathcal{V}$:

$$D_T(v) = |N_{\mathcal{G}}(v) \cap T|,$$

ts. $D_T(v)$ on solmun v joukossa T olevien naapurisolmujen lukumäärä.

Solmujoukon $\mathcal{V}$ ositus

$$B = [B[0], B[1], \ldots, B[|B| - 1]]$$

on *tasasuhtainen* (equitable) jos kaikilla $i, j \in \{0, 1, \ldots, |B|-1\}$ pätee $D_{B[j]}(u) = D_{B[j]}(v)$ kaikilla $u, v \in B[i]$.

Järjestetty ositus B on järjestettyä ositusta A *hienompi* jos

- jokaiselle osajoukolle $B[i]$ on olemassa osajoukko $A[j]$ siten, että $B[i] \subseteq A[j]$; ja
- jos $u \in A[i_1]$ ja $v \in A[j_1]$, missä $i_1 < j_1$, niin tällöin $u \in B[i_2]$, $v \in B[j_2]$, missä $i_2 < j_2$.

Kääntäen vastaavassa tilanteessa voidaan sanoa, että A on B:tä *karkeampi* ositus. B on aidosti hienompi (aidosti karkeampi) kuin A jos B on A:ta hienompi (karkeampi) ja lisäksi pätee $A \neq B$.

Olkoon A graafin solmujoukon järjestetty ositus. Ositukselle A on olemassa (osituksen komponenttien järjestystä vaille) yksikäsitteinen karkein tasasuhtainen ositus B, joka on hienompi kuin A. Allaoleva algoritmi löytää (erään) järjestetyn karkeimman tasasuhtaisen osituksen B, joka on hienompi kuin A.

1. Aseta $B = A$.

2. Tallenna B:n osajoukot järjestyksessä pinoon $\mathcal{S}$.

3. Toista kunnes $\mathcal{S} = \emptyset$:

   (a) Poista osajoukko T pinon $\mathcal{S}$ päältä.

   (b) Toista kaikille B:n osajoukoille $B[i]$:

      i. Aseta $L[h] = \{v \in B[i] \ : \ D_T(v) = h\}$ kaikilla $h = 0, 1, \ldots, |\mathcal{V}|-1$.

      ii. Jos L sisältää enemmän kuin yhden ei-tyhjän joukon, korvaa $B[i]$ ei-tyhjillä joukoilla $L[0], L[1], L[2], \ldots,$ ja tallenna ko. ei-tyhjät joukot pinoon $\mathcal{S}$.

Tehtävän graafin solmujen asteluvut ovat

Solmu	0	1	2	3	4	5	6
Asteluku	2	5	4	4	4	1	2

.

Ositetaan solmut kasvavan asteluvun mukaiseen järjestykseen

$$A = [\{5\}, \{0, 6\}, \{2, 3, 4\}, \{1\}].$$

Suoritetaan annettua algoritmia askel kerrallaan. Alkutilassa

$$\begin{aligned} B &= [\{\overset{0}{5}\}, \{\overset{1}{0}, \overset{1}{6}\}, \{\overset{1}{2}, \overset{1}{3}, \overset{1}{4}\}, \{\overset{0}{1}\}] \\ \mathcal{S} &= [\{5\}, \{0, 6\}, \{2, 3, 4\}, \underbrace{\{1\}}_{=T}] \end{aligned}$$

(kunkin B:n solmun v yläpuolelle on merkitty vastaava arvo $D_T(v)$). Selvästi mikään osituksen komponentti ei jakaannu, koska $D_T(v)$ on vakio kaikissa komponenteissa. Tilanne on vastaava seuraavalla askelella:

$$\begin{aligned} B &= [\{\overset{1}{5}\}, \{\overset{1}{0}, \overset{1}{6}\}, \{\overset{2}{2}, \overset{2}{3}, \overset{2}{4}\}, \{\overset{3}{1}\}] \\ \mathcal{S} &= [\{5\}, \{0, 6\}, \underbrace{\{2, 3, 4\}}_{=T}]. \end{aligned}$$

Kolmannella askelella komponentti $\{2, 3, 4\}$ jakaantuu osiin $\{3\}$ ja $\{2, 4\}$:

$$\begin{aligned} B &= [\{\overset{0}{5}\}, \{\overset{0}{0}, \overset{0}{6}\}, \{\overset{1}{2}, \overset{0}{3}, \overset{1}{4}\}, \{\overset{2}{1}\}] \\ \mathcal{S} &= [\{5\}, \underbrace{\{0, 6\}}_{=T}]. \end{aligned}$$

Nyt ositus säilyy muuttumattomana pinon $\mathcal{S}$ tyhjentymiseen asti:

$$\begin{aligned} B &= [\{\overset{0}{5}\}, \{\overset{1}{0}, \overset{1}{6}\}, \{\overset{2}{3}\}, \{\overset{1}{2}, \overset{1}{4}\}, \{\overset{2}{1}\}] \\ \mathcal{S} &= [\{5\}, \{3\}, \underbrace{\{2, 4\}}_{=T}]. \end{aligned}$$

$$\begin{aligned} B &= [\{\overset{1}{5}\}, \{\overset{0}{0}, \overset{0}{6}\}, \{\overset{0}{3}\}, \{\overset{1}{2}, \overset{1}{4}\}, \{\overset{1}{1}\}] \\ \mathcal{S} &= [\{5\}, \underbrace{\{3\}}_{=T}]. \end{aligned}$$

$$\begin{aligned} B &= [\{\overset{0}{5}\}, \{\overset{0}{0}, \overset{0}{6}\}, \{\overset{1}{3}\}, \{\overset{0}{2}, \overset{0}{4}\}, \{\overset{0}{1}\}] \\ \mathcal{S} &= [\underbrace{\{5\}}_{=T}]. \end{aligned}$$

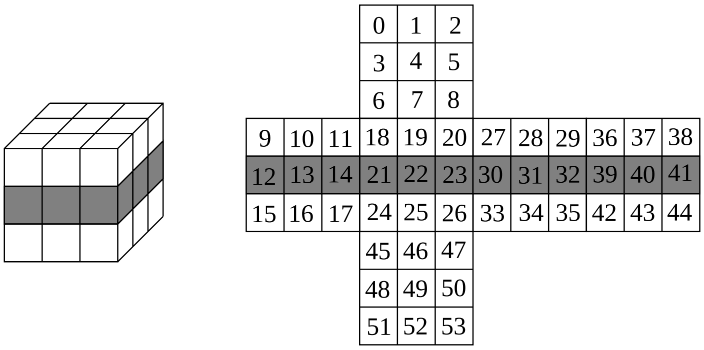
Haluttu ositus on siis

$$B = [\{5\}, \{0, 6\}, \{3\}, \{2, 4\}, \{1\}].$$

Kotitehtävä 3

Kolmannessa kotitehtävässä tutkitaan Rubikin kuutiota käyttäen hyväksi permutaatioryhmien käsittelyyn kehitettyjä algoritmeja.

Väritetään Rubikin kuution 54 pikkuneliötä mustalla ja valkoisella kuten oheisessa kuvassa.



Tehtävänä on määrittää, kuinka moneen erilaiseen tilaan kuvattu Rubikin kuutio voidaan saattaa siirtosarjoilla, jotka koostuvat koko kuution ja/tai sen sivutahkojen kierroista. (Kahta tilaa pidetään samoina, jos ne voidaan muuntaa toisikseen koko kuutiota kiertämällä.)

Tehtävän ratkaisua varten numeroidaan kuution 54 pikkuneliötä kuten kuvassa yllä. Tällöin kuution sallitut siirrot määräävät kukin joukon $X = \{0, 1, \ldots, 53\}$ permutaation, ja siirtosarjoja vastaavat permutaatiot saadaan siirtojen määräämien permutaatioiden yhdisteinä. Nämä permutaatiot muodostavat ryhmän G.

G:n virittävät esimerkiksi seuraavat kolme permutaatiota:

$$\begin{aligned}
\alpha_1 =& (0,2,8,6)(1,5,7,3)(9,36,27,18)(10,37,28,19)(11,38,29,20) \\
& \cdots(12,39,30,21)(13,40,31,22)(14,41,32,23)(15,42,33,24) \\
& \cdots(16,43,34,25)(17,44,35,26)(45,51,53,47)(46,48,52,50) \\
\alpha_2 =& (0,18,45,44)(1,19,46,43)(2,20,47,42)(3,21,48,41)(4,22,49,40) \\
& \cdots(5,23,50,39)(6,24,51,38)(7,25,52,37)(8,26,53,36) \\
& \cdots(9,11,17,15)(10,14,16,12)(27,33,35,29)(28,30,34,32) \\
\beta =& (0,2,8,6)(1,5,7,3)(9,36,27,18)(10,37,28,19)(11,38,29,20)
\end{aligned}$$

Permutaatiot α_1, α_2 vastaavat koko kuution kiertoa akselien $\{4, 49\}$ ja $\{13, 31\}$ ympäri, ja permutaatio β vastaa sivun $\{0, 1, \ldots, 8\}$ kiertoa.

(Käännä)

Tehtävät.

1. Määritä ryhmän G koko.

2. Olkoon $H \subseteq G$ permutaatioiden α_1, α_2 virittämä G:n aliryhmä. Mikä on tämän aliryhmän koko?

3. Olkoon $S = \{12, 13, 14, 21, 22, 23, 30, 31, 32, 39, 40, 41\} \subset X$ mustiksi väritettyjen ruutujen joukko. Määritä radan $G(S) = \{g(S) \ : \ g \in G\}$ pituus $|G(S)|$.

4. Määritellään ekvivalenssirelaatio joukkoon $G(S)$ siten, että joukot $T, T' \in G(S)$ ovat ekvivalentteja, jos on olemassa $h \in H$ siten, että $h(T) = T'$. Kuinka monta ekvivalenssiluokkaa joukkoon $G(S)$ muodostuu, ja minkä kokoisia ne ovat?

5. Selvitä kohtien (3) ja (4) avulla kuinka moneen erilaiseen tilaan väritetty kuutio voidaan lopulta saattaa. Perustele vastauksesi.

Erityisen ansiokkaasta ratkaisusta voi saada bonuspisteen tenttiin.

Valmisohjelmat. Ongelman ratkaisussa kannattaa käyttää hyväksi jo olemassaolevia työkaluja. Pienimuotoinen permutaatioryhmien käsittelyyn soveltuva ohjelma löytyy osoitteesta

⟨`http://www.tcs.hut.fi/Teaching/T-79.161/group.c`⟩.

Ohjelman tulisi kääntyä ainakin laskentakeskuksen koneissa esimerkiksi komennolla `gcc -O2 -Wall -o group group.c`. Huomattavasti monipuolisempi (mutta myös raskaampi) työkalu on esimerkiksi `GAP`, joka on saatavissa osoitteesta ⟨`http://www-groups.dcs.st-and.ac.uk/~gap/`⟩.

Palautus. Kotitehtävä on palautettava sähköpostitse assistentille (Petteri Kaski, `Petteri.Kaski@hut.fi`) viimeistään maanantaina 29.4.2002. Selostuksen voi palauttaa esimerkiksi hyvälaatuisena PostScriptinä. Selostuksen lisäksi palautuksessa on oltava liitteenä ongelman ratkaisuun käytetty oma lähdekoodi.