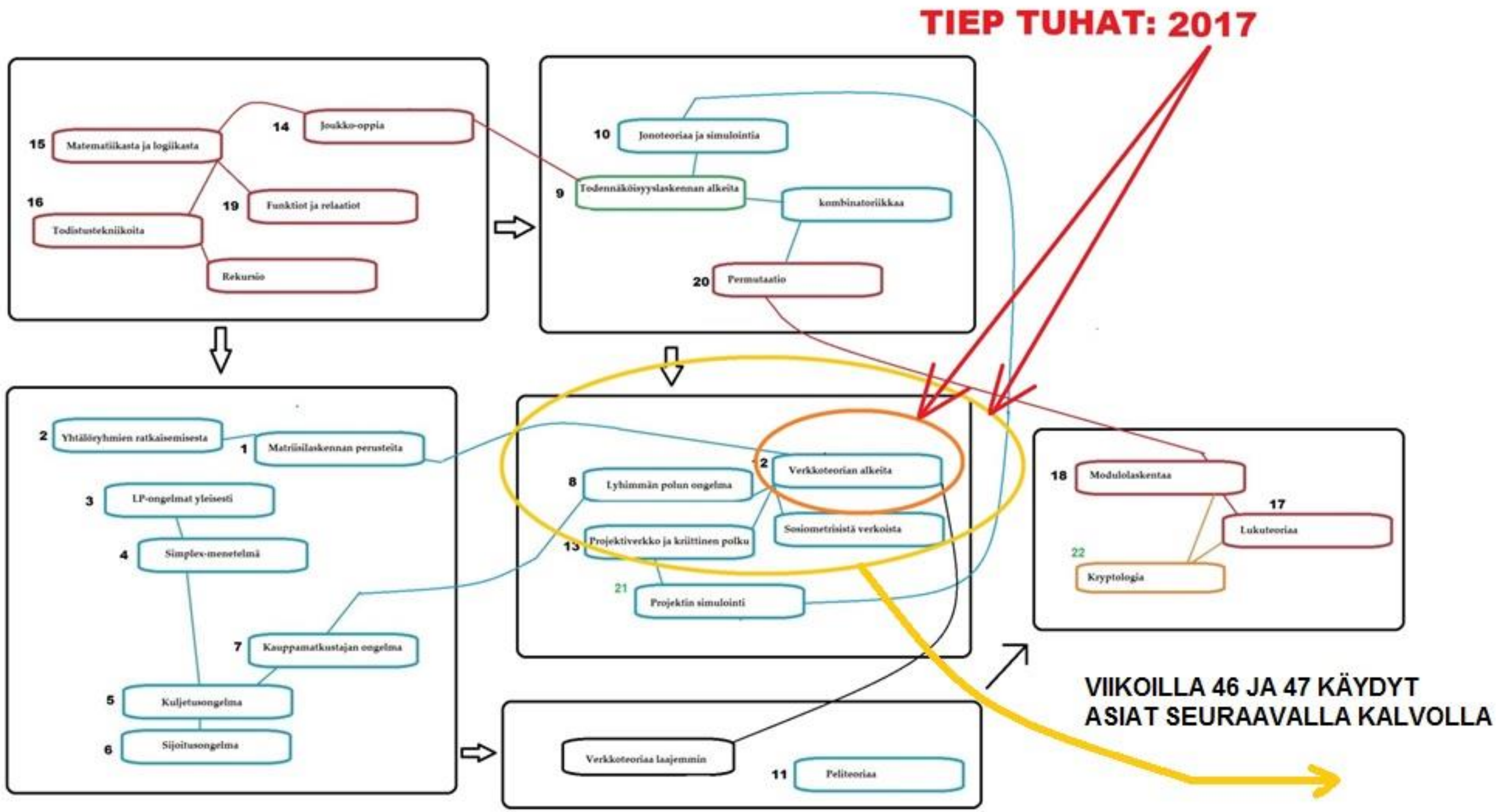
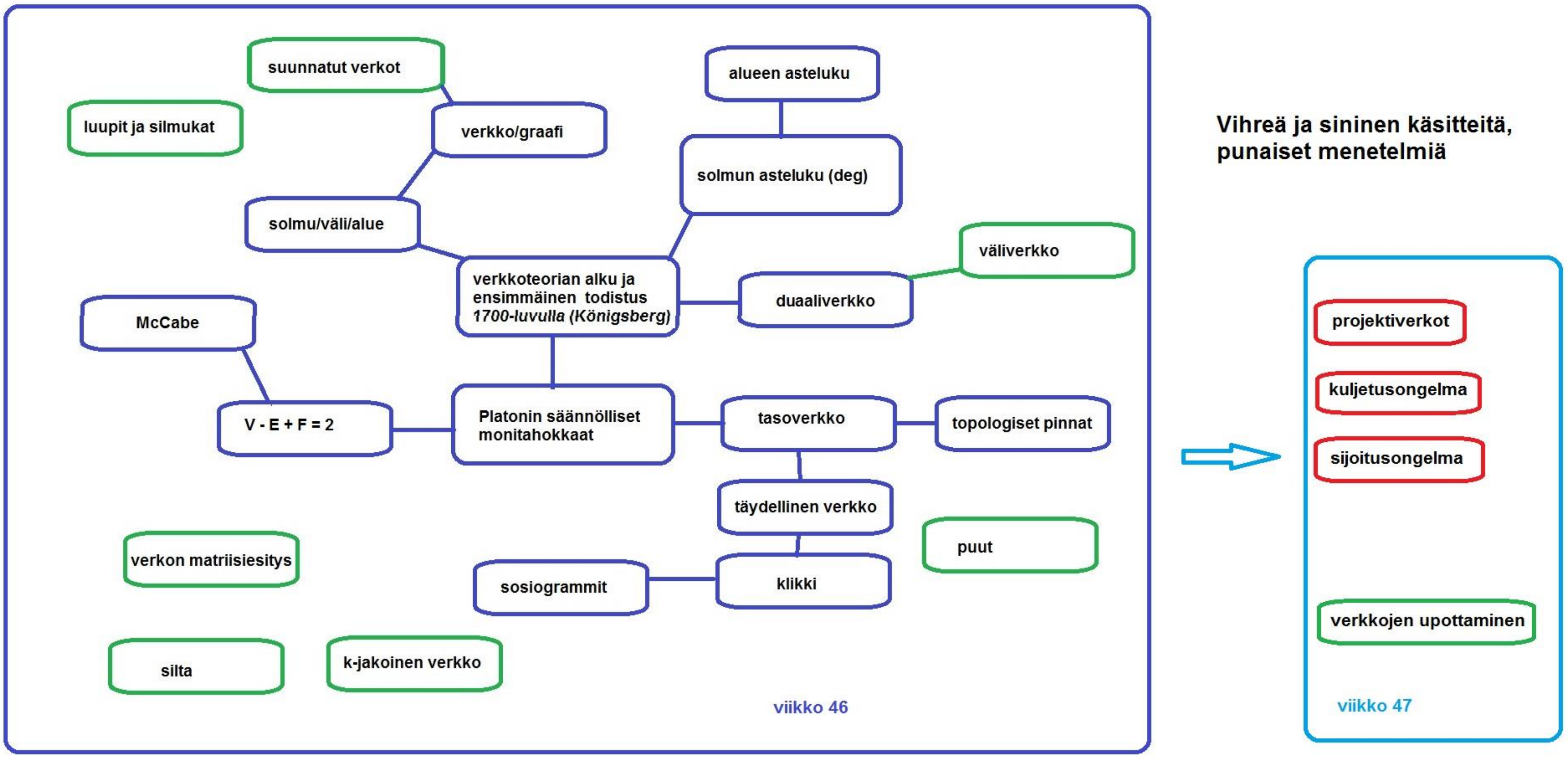


TIEP TUHAT: 2017
15 Matematiikasta ja logiikasta
14 Joukko-oppia
16 Todistustekniikoita
19 Funktiot ja relaatiot
Rekursio
10 Jonoteoriaa ja simulointia
9 Todennäköisyyslaskennan alkeita
kombinatoriikkaa
20 Permutaatio
2 Yhtälöryhmien ratkaisemisesta
1 Matriisilaskennan perusteita
3 LP-ongelmat yleisesti
4 Simplex-menetelmä
7 Kauppamatkustajan ongelma
5 Kuljetusongelma
6 Sijoitusongelma
8 Lyhimmän polun ongelma
12 Verkkoteorian alkeita
Sosiometrisistä verkoista
13 Projektiverkko ja kriittinen polku
21 Projektin simulointi
Verkkoteoriaa laajemmin
11 Peliteoriaa
18 Modulolaskentaa
17 Lukuteoriaa
22 Kryptologia
VIIKOILLA 46 JA 47 KÄYDYT
ASIAT SEURAAVALLA KALVOLLA

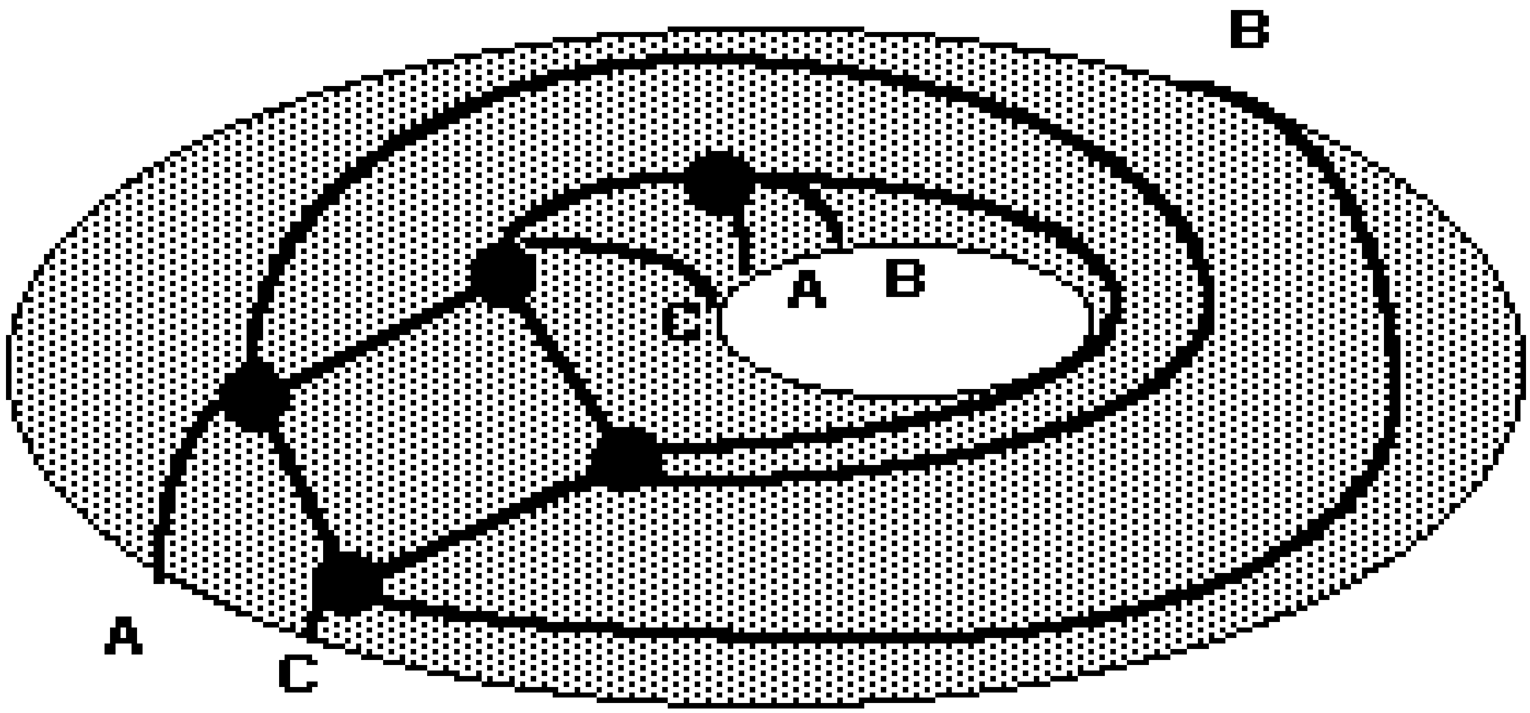

suunnatut verkot
luupit ja silmukat
verkko/graafi
alueen asteluku
solmun asteluku (deg)
solmu/väli/alue
väliverkko
verkkoteorian alku ja ensimmäinen todistus 1700-luvulla (Königsberg)
duaaliverkko
McCabe
V - E + F = 2
Platonin säännölliset monitahokkaat
tasoverkko
topologiset pinnat
täydellinen verkko
puut
verkon matriisiesitys
sosiogrammit
klikki
silta
k-jakoinen verkko
viikko 46
Vihreä ja sininen käsitteitä, punaiset menetelmiä
projektiverkot
kuljetusongelma
sijoitusongelma
verkkojen upottaminen
viikko 47

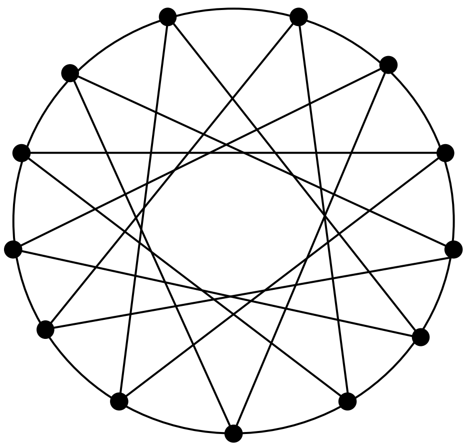
Viikolla 46 läpikäydyt käsitteet

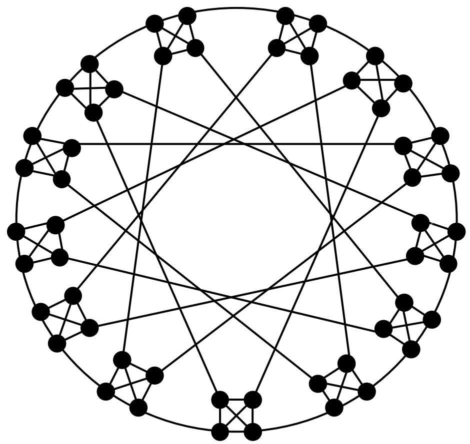
Viikolla 47 läpikäydyt käsitteet

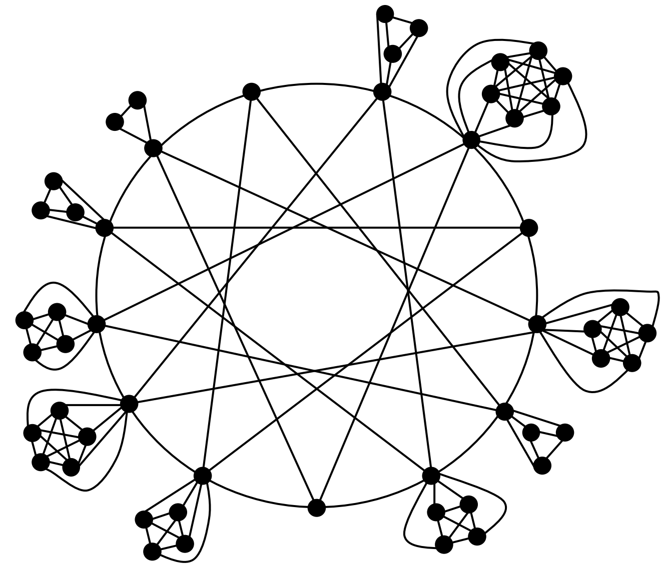
Verkko eli graafi, tasoverkko, solmut, välit, alueet, suunnatut verkot, isomorfiset verkot, verkon duaali, verkon upottaminen, verkon genus, verkon komplementti, aliverkko, täydellinen verkko, solmun ja alueen asteluku, väliverkko, säännöllinen verkko, Eulerin ja Hamiltonin verkot, kauppamatkustajan ongelma, verkon kromaattinen luku, kaksijakoinen verkko, erilaiset polut ja silmukat, verkon yhtenäisyys, puurakenteet, projektiverkko ja sen kriittinen polku …

4CC


B
A
B
C
A
C







-2.65109	(4)
0.273891	(4)
1.3772	(4)
4	

-2	(14)
-1.52531	(4)
-1.2965	(4)
-0.532614	(4)
0	(13)
-2.53261	(4)
3.2965	(4)
3.52531	(4)
4	

-3.29593	-3.17393
-3.11764	-3.01577
-1	(29)
-0.7877	-0.748681
-0.675554	-0.317924
-0.0533663	0.173402
0.587342	0.733935
1.52438	2.37093
3.04718	3.17003
3.26022	4.04955
4.14063	4.51908
4.94997	5.48287
6.17698	

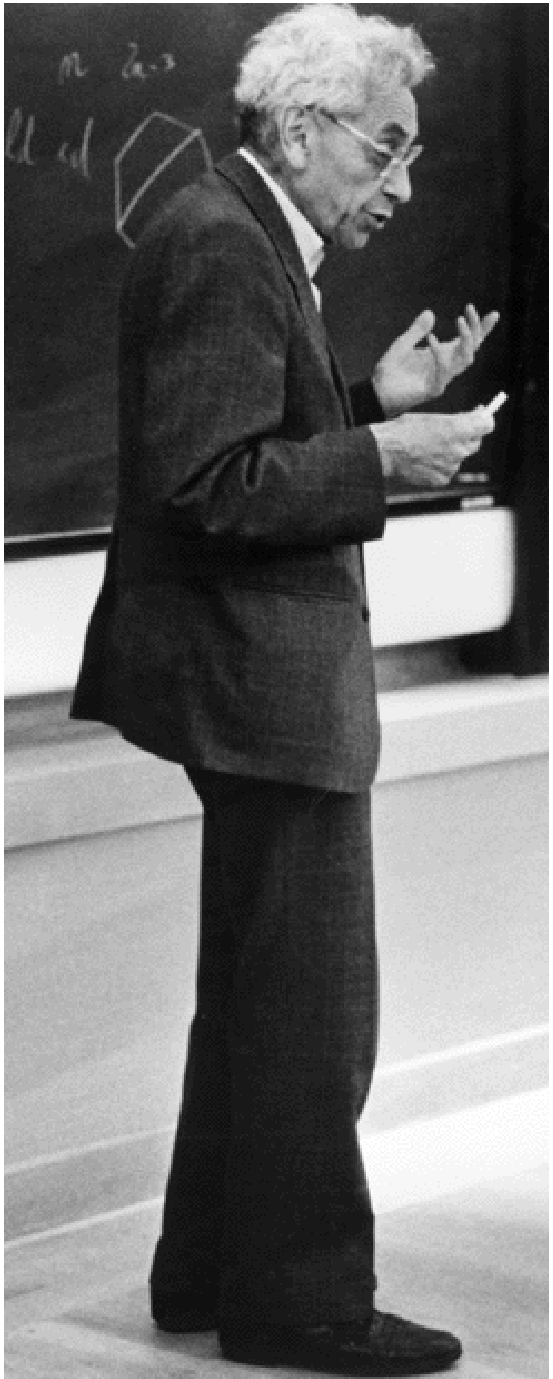
6 degrees of separation ! Stanley Milgram (1967)

160 letters Omaha -Nebraska- -> Boston

C'est petit le monde !!

What a small-world !

¡ El mundo es un pañuelo !



Erdös number

http://www.acs.oakland.edu/~grossman/erdoshp.html

1- 493
2- 5608

D=11 R=6
Δ = 1- 485
$\Delta_{mitjà}$ = 5.56

......
Collins, P. J.
Colwell, Peter
Comellas, Francesc
Comets, Francis M.
Comfort, W. Wistar
Compton, Kevin J.
Conder, Marston
Conrey, J. Brian
CONWAY, JOHN HORTON
Conze-Berline, Nicole
Cook, Curtis R.
Cook, Janice
.....

Notable Erdös coautors :

Frank Harary (257 coautors)
Noga Alon (143 coautors)
Saharon Shela (136)
Ronald Graham (120)
Charles Colbourn (119)
Daniel Kleitman (115)
A. Odlyzko (104)

Erdös had no common articles with his Ph D supervisor, Leopold Fejér

Projektiverkoista

•*Töiden järjestely* muistuttaa sijoitteluongelmaa, haetaan n:lle eri työlle tapa tehdä ne m:n eri vaihtoehdon joukosta. Erotuksena sijoitteluongelmaan on töiden suoritusjärjestyksiin liittyvät ehdot. Töiden järjestelyn eräs sovellusalue on *projektisuunnittelu*.

•Projektisuunnitteluun on tehty erilaisia *projektinhallintamenetelmiä*, joista tunnetuimpia ovat 50-luvun lopulla kehitetyt CPM (Critical Path Method) ja PERT (Program Evaluation and Review Technique). Nämä menetelmät eroavat selvimmin ajan arvioinnissa. CPM:ssä se on <u>deterministinen</u> eli perustuu ennalta määrättyihin aikoihin kun taas PERT:issä ajan arviointi tapahtuu <u>stokastisesti</u> eli apuna käytetään satunnaislukujakaumia.

•Projektinhallinta on selvästi eräs verkkoteorian sovellusalue ja keskeistä osaa näyttelee ns. *projektiverkko* ja *kriittisen polun* hakeminen projektiverkosta. Verkon kriittinen polku tarkoittaa lyhyesti pisintä polkua alkusolmusta loppusolmuun kun verkon kaikki välit käydään läpi täsmälleen yhden kerran. Kriittisen polun pituus (pituus tarkoittaa ajan pituutta) antaa samalla lyhimmän ajan jolla projekti on mahdollista viedä läpi.

•Projektiverkon (*) piirtämisessä on noudatettava eräitä sääntöjä: 1. Verkon jokaisella välillä on suunta (verkko on suunnattu) ja verkolla on aina yksi alku- ja loppusolmu, 2. Välit kuvaavat töitä eli aktiviteetteja, 3. Verkon solmut on numeroitu nousevasti s.e. jokaisen välin päätesolmulla on aina suurempi numero kuin sen lähtösolmulla, 4.

Verkko ei saa sisältää silmukoita (eli projekti ei saa jäädä “luuppiin”) 5. Jokaisen aktiviteetin kuvaamiseen käytetään ainoastaan yhtä väliä ja jokainen väli esittää ainoastaan yhtä aktiviteettia, 6. Kahden solmun välissä on vain yksi aktiviteetti (suunnattu väli), 7. Jotta verkko voitaisiin piirtää s.e. ehdot 5 ja 6 eivät tulisi rikotuiksi, on verkkoon joskus lisättävä ns. *valeaktiviteetteja* (dummy activity), joiden kesto on nolla. Esimerkiksi jos työ A edeltää töitä C sekä D ja työ B edeltää myös D:tä, mutta ei C:tä niin verkon piirtäminen onnistuu lisäämällä valetyö A:n päätesolmusta (2) B:n päätesolmuun (3). Katso oheisen esimerkin kuvaa alla. Huom! Jos jollakulla aktiviteetilla ei ole edeltäjää niin sen lähtösolmu on tietenkin solmu nro 1.

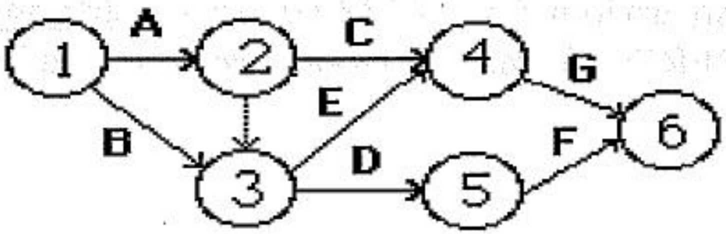
•Projektin kokonaiskesto saadaan selville kriittisen polun avulla. Kriittisen polun määräämiseksi on tarpeen laskea seuraavia lukuarvoja: varhaisin aloitusaika (ES, earliest star time), myöhäisin aloitusaika (LF, latest start time), varhaisin lopetusaika (EF, earliest finish time), myöhäisin lopetusaika (LF, latest finish time) ja aktiviteetin pelivara (TS,). Nämä arvot voidaan laskea seuraavin yhtälöin kun aktiviteetin kestoa merkitään T:llä: ES(alkusolmusta lähtevä aktiviteetti) = 0, ES(x) = max{EF(x:ää edeltävä aktiviteetti)} eli suurin x:ää edeltävien aktiviteettien varhaisimmasta päättymisajasta, EF(x) = ES(x) + T(x). Projektin aikaisin mahdollinen päättymisaika on EF(n) kun n on viimeinen aktiviteetti. Jatketaan laskemalla takaperin: LF(n) = EF(n), LS(n) = LF(n) - T(n), ..., LF(x) = min{LS((x:ää seuraava aktiviteetti)} eli pienin x:ää seuraavien aktiviteettien myöhäisimmästä alkamisajasta, LS(x) = LF(x) - T(x). Kun verkko on laskettu “läpi” niin saadaan pelivarat yksinkertaisesti: TS(x) = LS(x) - ES(x).

- Jos TS(x) = 0 niin aktiviteetti kuuluu projektiverkon kriittiseen polkuun.
- Kriittisellä polulla ei siis ole pelivaraa ja vastaavasti ne aktiviteetit, joilla TS>0, on ns. vapaata pelivaraa. Vapaa pelivara on se aika, minkä ko. työ voi myöhästyä vaikuttamatta seuraavan työn alkamiseen.

Tiistain luennolla 21.11. käyty esimerkki

Esimerkki.

On piirrettävä projektiverkko, jossa töiden järjestys on määrätty seuraavasti: Työtä C edeltää työ A, töitä E ja D edeltävät työt A ja B, työtä G edeltävät työt C ja E sekä työtä F edeltää työ D. Tutki missä ajassa projekti on mahdollista viedä läpi kun töiden kestoajat ovat seuraavat: A=3, B=2, C=3, D=5, E=3, F=2 ja G=5.



Lasketaan arvot kullekin aktiviteetille. Esimerkiksi EF(A) = ES(A) + T(A) = 0 +3 = 3,... (lasketaan verkko loppuun), LF(A) = min {LS(C), LS(D), LS(E)} = min{3, 3, 4} = 3, LS(A) = LS(A) - T(A) = 3 - 3 = 0 ja TS(A) = LS(A) - ES(A) = 3 - 3 = 0. Ohessa taulukko.

	T	ES	EF	LS	LF	TS
A	3	0	3	0	3	0
B	2	0	2	1	3	1
C	3	3	6	3	6	0
D	5	3	8	4	9	1
E	3	3	6	3	6	0
F	2	8	10	9	11	1
G	5	6	11	6	11	0

Kriittiseen polkuun kuuluvat siis työt A, C, E ja G.

KULJETUSONGELMAssa on joukko tavaran tuottajia (m kpl) ja joukko tavaran hankkijoita (n kpl). Kukin tuottaja voi toimittaa vain tietyn määrän tavaraa (ai, i=1..m) ja kukin hankkija tarvitsee tietyn määrän tavaraa (bj, j=1..n). Kuljetuskustannuksia tuottaja/hankkija-parin välillä merkitään cij. Tarkoituksena on suunnitella toimitetun tavaramäärän (xij) kuljetuksen jakautuminen tuottajilta hankkijoille siten että kustannukset tulevat minimoitua kun samalla huomioidaan annetut rajoitukset. Kuljetusongelmaa voidaan kuvata kaksijakoisella suunnatulla verkolla (koostuu kahdesta solmujoukosta vailla sisäisiä välejä, *kaksijakoinen=bipartite*, *suunnattu verkko=digraph*). Kuljetusongelma voidaan esittää LP-ongelmana allaolevassa muodossa:

$$\text{Min } z = \sum_{i=1}^{m} \sum_{j=1}^{n} c_{ij} x_{ij}$$

$$\sum_{i=1}^{m} x_{ij} = a_i \quad \text{kun } i = 1 \ldots m$$

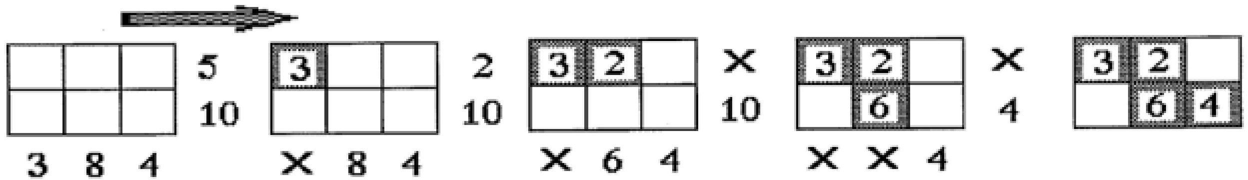
$$\sum_{j=1}^{n} x_{ij} = b_j \quad \text{kun } j = 1 \ldots n$$

$$x_{ij} \geq 0$$

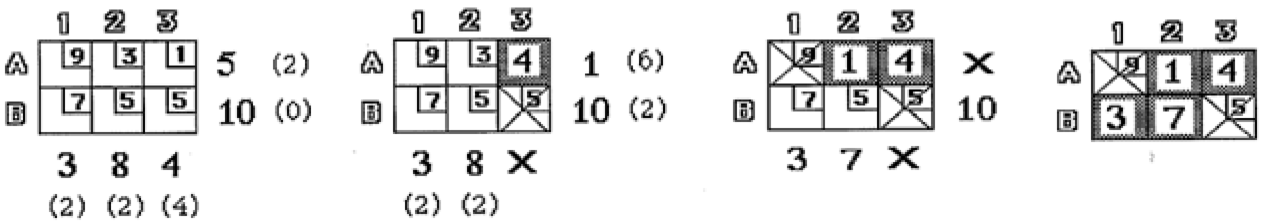
Kuljetusongelma voidaan ratkaista mm. siten että muodostetaan kustannusmatriisi, josta haetaan jokin alkukantaratkaisu, jossa on aina m+n-1 alkiota. Alkukantaratkaisun pohjalta lähdetään hakemaan optimiratkaisua kuljetusalgoritmin avulla. Samalla tehtävällä voi olla useitakin optimiratkaisuja. Kantaratkaisun hakemiseen voidaan käyttää mm. seuraavia menetelmiä: luoteiskulmamenetelmä, minimikustannusmenetelmä tai Vogelin menetelmä. Näistä ensin mainittu on vaivattomin, mutta antaa yleensä kehnoimmat alkuarvot. Paras tulos saadaan Vogelin menetelmällä, joka on työläin mutta palkitsee vähäisemmillä iteraatioilla itse kuljetusalgoritmissa.

Esimerkki: A tuottaa tavaraa 5 yksikköä ja B 10 yksikköä. Hankkijat 1, 2 ja 3 tarvitsevat k.o. tavaraa 3, 8 ja 4 yksikköä. Kuljetuskustannukset ovat markkoina: A1=9, A2=3, A3=1, B1=7, B2=5, B3=5. Haetaan ohessa alkukantaratkaisut luoteiskulmamenetelmää ja Vogelin menetelmää käyttäen.

1° Luoteiskulmamenetelmä (lähdetään “luoteiskulmasta” ja sijoitetaan ruutuun suurin mahdollinen tavaramäärä, poistetaan <u>yksi</u> rivi tai sarake):



2° Vogelin menetelmä (lasketaan kullekin riville ja sarakkeelle sakkoluku - kahden pienimmän arvon erotus - ja valitaan suurimman sakkoluvun riviltä/sarakkeelta pienimmän kustannuksen ruutu, sijoitetaan ruutuun suurin mahdollinen tavaramäärä):

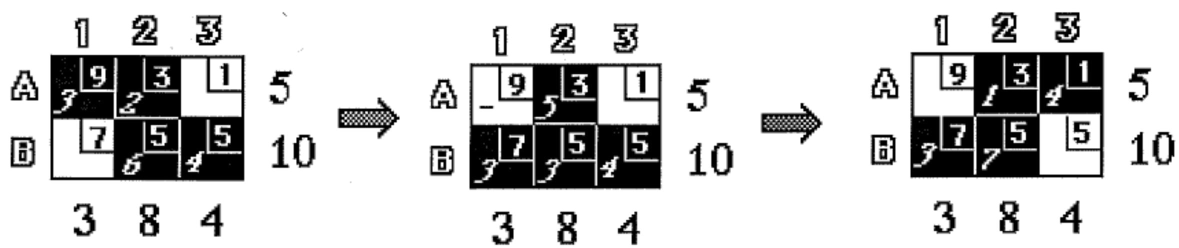


Lopputulokset:

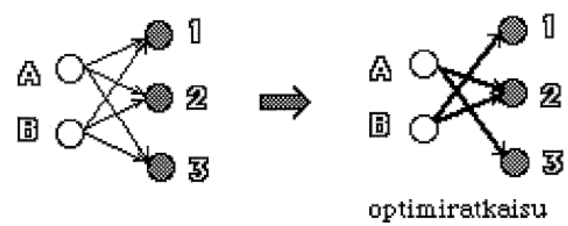
Luoteiskulmamenetelmällä 3*9+2*3+6*5+4*5 = 83

Vogelin menetelmällä 3*7+7*5+1*3+4*1 = 63

Luoteiskulmamenetelmällä saatiin kantaratkaisu, jonka ulkopuolelle jäivät ruudut A3 ja B1. Kuljetusalgoritmissa lasketaan sakot näille kantaratkaisun ulkopuolelle jääneille ruuduille, joita esimerkkitapauksessa on vain kaksi. Sakon arvo saadaan kiertämällä ruutua vastaava polku (ks. sivun alin kuva) kannassa ja vuorotellen summaamalla ja vähentämällä kustannukset. Polku kierretään liikkumalla vuoroin pysty- ja vaakasuoraan, kiertosuunta ei vaikuta tulokseen. Saadaan: S(A3) = A3 - B3 + B2 - A2 = 1 - 5 + 5 - 3 = -2 ja S(B1) = B1 - B2 + A2 - A1 = 7 - 5 + 3 - 9 = -4. Algoritmi pysähtyy jos yksikään lasketuista sakoista ei ole negatiivinen. Mikäli negatiivisia sakkolukuja löytyy niin valitaan kantaan itseisarvoltaan suurimman arvon omaava ruutu; tässä tapauksessa siis B1. Kuormitus uudelle kantaruudulle saadaan kiertämällä sama polku jolta sakko laskettiin ja siirtämällä polulta kyseiseen kantaruutuun suurin mahdollinen kuorma. Esimerkkitapauksessa se on min{3,6} = 3. Kun polku kierretään ympäri niin siirretään kuormia ruudusta toiseen siten että lopputilanteessa on tasapaino jälleen saavutettu eli rivi ja sarakesummat täsmäävät. Siis seuraavasti: B1 ↵ +3 ⇒ 3, A1 ↵ -3 ⇒ 0, A2 ↵ +3 ⇒ 5, B2 ↵ -3 ⇒ 3. Se ruutu, jonka kuormitus nollautuu (A1) poistuu kannasta. Jos useamman ruudun kuorma tulee nollaksi niin kannasta poistetaan ruutu, jolla on korkeimmat kustannukset.



Lasketaan uuden kantaratkaisun ulkopuolelle jääneiden ruutujen sakot: S(A1) = 9 - 3 + 5 - 7 = 4 ja S(A3) = 1 - 5 + 5 - 3 = -2. Kantaan otetaan siis ruutu A3, johon siirretään kuormaa 4 yksikköä. Siis: A3 ↵ +4 ⇒ 4, B3 ↵ -4 ⇒ 0, B2 ↵ +4 ⇒ 7, A2 ↵ -4 ⇒ 1, joten kannasta poistuu B3. Ulkopuolelle jääneet ruudut ovat nyt A1 ja B3, joiden sakot, S(A1) = 9 - 7 + 5 - 3 = 4 ja S(B3) = 5 - 5 + 3 - 1 = 2 ovat molemmat positiivisia. Näiden algoritmi päättyy ja saatu ratkaisu on optimi. Huomataan, että se on sama kuin Vogelin menetelmällä haettu alkukantaratkaisu. Ratkaisua voidaan kuvata alla olevalla kaksijakoisella suunnatulla verkolla.



SIJOITUSONGELMAssa on kaksi yhtä suurta joukkoa ja kummankin joukon jokaiselle alkiolle on haettava vastinalkio toisesta joukosta. On esimerkiksi jaettava k kpl tehtäviä k:n henkilön kesken s.e. kullekin tulee täsmälleen yksi (sopivin) tehtävä. Sijoitusongelma on *kuljetusongelman erikoistapaus*. Siis sellainen kuljetusongelma, jossa tuottajia ja hankkijoita on yhtä paljon ja jossa kukin tuottaja vie optimiratkaisussa tavaraa ainoastaan täsmälleen yhdelle hankkijalle, joten optimiratkaisussa on alkioita n kpl (kuljetusongelmassahan määrä oli 2n-1 kun m=n). Mikäli tuottajien ja hankkijoiden määrä ei ole sama niin lisätään varjomuuttujia (dummy person), joille kustannuskerroin (cij) on nolla. Sijoitusongelma voidaan esittää LP-ongelmana allaolevassa muodossa (vertaa vastaavaan kuljetusongelman esitykseen):

$$\text{Min } z = \sum_{i=1}^{n} \sum_{j=1}^{n} c_{ij} x_{ij}$$

$$\sum_{i=1}^{n} x_{ij} = 1 \qquad \text{kun } i = 1 \ldots n$$

$$\sum_{j=1}^{n} x_{ij} = 1 \qquad \text{kun } j = 1 \ldots n$$

$$x_{ij} = 0 \text{ tai } 1$$

Sijoitusongelma voitaisiin kuljetusongelman erikoistapauksena myös ratkaista sellaisena, mikä kuitenkin on turhan työlästä. On syytä muistaa, että tuolloin saadaan 2n-1 ratkaisua, joista n valitaan (loput n-1 alkiota ovat aina nollia). Sijoitusongelman ratkaisemiseksi on kehitetty ns. unkarilainen menetelmä, joka sisältää seuraavat vaiheet: 1° ongelmasta muodostetun nxn-matriisin kultakin riviltä vähennetään sen riviminimi, 2° sen jälkeen kultakin sarakkeelta vähennetään sarakeminimi, 3° yliviivataan kaikki 0-alkiot siten että viivoja on mahdollisimman vähän (aluksi eniten nollia sisältävät rivit/sarakkeet) 4° jos viivojen lukumäärä on suurempi tai yhtä suuri kuin n niin mennään viimeiseen askeleeseen (6°), 5° etsitään pienin alkio (merk. L), jota ei ole peitetty viivalla ja *vähennetään se kaikista <u>peittämättömien rivien</u> alkioista* ja *lisätään kaikkien <u>peitettyjen sarakkeiden</u> alkioihin* ja palataan askeleeseen 3°, 6° valitaan ratkaisuun tulevat nolla-alkiot siten että kultakin riviltä ja sarakkeelta tulee täsmälleen yksi alkio (huom! ratkaisu ei ole aina yksiselitteinen).