

1. Muodosta oman ”virtuaalilevysi” X: juureen kansio (hakemisto) nimeltä DISKREET, ja tälle vielä alikansio RELAATIO. Siirry Matlabissa kyseiseen kansioon kirjoittamalla

```
» cd X:\DISKREET\RELAATIO
```

a) Kirjoita Matlab-funktio RandRel.m, joka laskee satunnaisen relaation matriisin, kun lähtöetoina annetaan matriisin koko ja ykkösten suhteellinen osuus prosentteina. Valitse ensin **File** valikosta **New** ja **M-file** (tai kirjoita » edit RandRel). Avautuvassa Matlab-editorissa kirjoitetaan määrittely (%:lla alkavat rivit kommentteja):

```
function M = RandRel(m, n, prosenttiosuus);  
% satunnaisrelaatio RandRel.m: function M = RandRel(n,m,prosenttiosuus);  
% laskee ja tulostaa satunnaisen relaation matriisin, missä  
% m,n = matriisin koko, prosenttiosuus = ykkösten suhteellinen osuus  
apuM = rand(m,n); M = (apuM <= prosenttiosuus/100);
```

Matlabin käsky rand antaa tasaisesti välille [0, 1] jakautuneita lukuja. Normaalijakaumaa käyttää randn.

Funktiossa siis generoidaan ensin satunnaismatriisi apuM, ja sen avulla logiikkamatriisi M niin, että saadaan (keskimäärin) prosenttiosuus ykkösiä. Jos tämä on esimerkiksi 20, niin ykkönen tulee niihin kohtiin, joissa luku on pienempi kuin 0.2.

Talleta funktio kansioon X:\DISKREET\RELAATIO tyyliin **File, Save As** etc.

b) Muodosta nyt funktiollasi RandRel 100×100-satunnaismatriisi, jossa on noin 30% ykkösiä. Laske sen ykkösten määrä Matlabin sum-käskyn avulla, sekä ykkösten todellinen prosentuaalinen osuus.

```
» MA = RandRel(100, 100, 30)  
» ykk = sum(sum(MA))  
» ykk/10000
```

Mitä laskee käsky sum kohdistettuna matriisiin, mitä kaksinkertaisena? _____

2. a) Hae kurssimme Web-sivulta 'Kurssimateriaali'

<http://wanda.uef.fi/matematiikka/kurssit/Diskreetti/Kurssimateriaali/>
paketti Diskreet1.zip ja pura se Winzip-ohjelman Extract-käskyllä juurikansioon X:.

HUOM! Säilytä kansiorakenne! Siirry Matlabissakin kyseiseen kansioon kirjoittamalla » cd ..

Kokeile käskyjä pwd, help DISKREET, dir, !dir, what. **Aja lopuksi M-ohjelma** alustus.m (kirjoittamalla käsky alustus).

Siirry kansioon X:\DISKREET\RELAATIO vaikkapa ajamalla rel.m ja kokeile taas pwd, help DISKREET, dir etc. Suorita sitten seuraavat käskyt:

```
» M = SatRel(7,10) % toimii kuten RandRel  
» PiirRel2(M)  
» PiirAlk2(M, [1, 4], [6], '*b', '*g')  
» PiirNuo2(M, [1, 4], '-g')  
» M = zeros(7,10); PiirRel2(M)  
» PiirAlk2(M, [1, 4], [6], '*b', '*g')  
» PiirNuo2(M, [1, 6; 4, 6], '--g')  
» M(1,6) = 1; M(4,6) = 1; PiirRel2(M)
```

b) Muodosta Matlab-funktio LisaaNuo, joka sekä lisää nuolen matriisiin että piirtää sen kuvion. Miten? _____

3. Sinun pitäisi olla edelleen kansiossa

X:\DISKREET\RELAATIO

ja Matlab-funktio alustus.m suoritettuna.

Relaatioiden piirtofunktiot. Joukossa $X \times X$ määriteltyjä relaatioita varten on laadittu piirto-työkalut

```
piirtofunktioita 1: function PiirRel1( M, vari);
piirtofunktioita 1: function PiirAlk1( M, alkiot, vari);
piirtofunktioita 1: function PiirNuol(M, alkiopari, vari);
piirtofunktioita 1: function PiirSilm( M, alkio, vari);
lisäys ja piirto 1: function L = LisaNuol(M, alkiopari, vari);
poisto ja piirto 1: function L = PoisNuol(M, alkiopari, uusi); (tee, jos ei ole
```

Relaatioita $R \subseteq X \times Y$ varten on vastaavat piirtofunktiot 2, ks. » help RELAATIO

Suorita seuraavat käskyt ja vastaa kysymykseen:

```
» M = zeros(6)
» PiirRel1(M)
» M = LisaNuol(M, [2, 4])
» M = LisaNuol(M, [4, 6])
» M = LisaNuol(M, [6, 5])
» K = sign(M*M)
» [I,J] = find(K)
» N = [I,J]
» N(1, :)
» PiirNuol(M,N(1, :), '-b')
» PiirNuol(M,N(2, :), '-b')
```

Huomaatko mitään yhteyttä matriisiin M ja $K = M * M$ nuolilla? _____

4. Matlabissa voi useamman rivin ohjelman kirjoittaa myös suoraan komentoikkunassa. Voit ladata seuraavat matriisit myös tiedostosta relaatio.mat käskyllä load relaatio.mat

```
» A = [0 0 0 1 1; 1 1 0 0 1; 0 0 1 0 0; 1 1 0 0 0; 0 0 0 0 0]
» B = [0 1 0 0 0; 0 0 1 0 0; 1 0 0 0 0; 1 0 0 0 0; 0 1 0 0 1]
» C = [0 0 0 0 0; 0 0 0 0 0; 1 0 1 0 1; 0 0 0 0 0; 1 0 0 1 1]
» A, B, C
» any(A(1, :) & B(:, 2).') % Mitä tehdään
» any(A(1, :) & B(:, 3).') % näissä? _____
» T = zeros(5, 5)
» for i = 1:5 % paina enter nelosen perään!
    for j = 1:5 % samoin jne
        T(i, j) = any(A(i, :) & B(:, j)');
    end; end;
» T
» T | C
```

Mitä yllä tehtiin? _____

5. Nuoliluettelon ja Boolean tulon muodostavat funktiot.

a) Muodosta Matlab-funktio `Nuolet` (tiedostoksi `Nuolet.m`), joka saatuaan parametrina relaation matriisin antaa ulos nuolten 2-sarakkeisen matriisin (kuten 3:ssa N). Testaa funktiiasi:

```
» Nuolet(M)
» Nuolet(K)
» Nuolet(zeros(5))
```

Mitä antaa `Nuolet(zeros(5))`? _____

b) Muodosta funktio `BT`, joka laskee ja palauttaa syötteenä annettujen kahden logiikkamatriisin välisen matriisitulon Boolean operaatioilla. Voit lainata koodia Tehtävästä 4. Koeta sitten:

```
» A, B
» BT(A, B)
» BT(A, B) | C
```

Onko sama tulos kuin 4:ssä ja kotilaskussa 5/3. harjoitus? _____

6. Tulorelaatio ja piirtelyä.

a) Harjoitellaan relaatioiden yhdistämistä ja useamman kuvion piirtoa.

```
» C = SatRel(3, 4, 20)
» D = SatRel(4, 5, 20)
» E = BT(C, D)
» figure(1); PiirRel2(C)
» figure(2); PiirRel2(D)
» figure(3); PiirRel2(E)
```

Kuvioihin liittyy nyt ”kahva” (handle) 1, 2 tai 3. Muotoile kuvat niin, että 1 ja 2 ovat vierekkäin näytön yläreunassa ja 3 alareunassa.

Tutki onko alempi nyt todella ylempien tulorelaatio? _____

b) Mitä relaation yhdistäminen oikein tekee? Kokeillaan:

```
» P = [0      0      1      1      0      % paina enter rivin viimeisen jälkeen
0      0      0      0      0
0      1      0      0      0
1      0      0      0      0
0      0      0      0      1]
» figure(1); PiirRel1(P)
» P2 = BT(P, P), figure(2), PiirRel1(P2)
» P3 = BT(P2, P), figure(3), PiirRel1(P3)
» BT(P, P) <= P
» PP = P|P2
» BT(PP, PP) <= PP
» PP = P|P2|P3
» BT(PP, PP) <= PP
» figure(4); PiirRel1(PP)
```

Selitä, kuinka `P`:stä kehittyy lopullinen `PP`, ja mitä erikoista siinä on: _____

7. **Relaatioiden ominaisuustestejä: ekvivalenssi.** Tarkastellaan äärellisten relaatioiden ominaisuuksia refleksiivisyys, symmetrisyys, antisymmetrisyys, transitiivisuus ja olla täysi (vrt. kotilaskut 2).

a) Refleksiivisyys tarkoitti, että jokainen alkio on relaatioissa itsensä kanssa: matriisien kielellä tämä tarkoittaa, että päälävistäjällä on ykköset. Diagonaalia käsitellään vaikkapa seuraavasti:

```
» M = SatRel(10,10,50)
» diag(M)
» all(diag(M))
» D = diag(diag(M))
» all(diag(D))
» S = M | eye(size(M))
» all(diag(S))
```

Kirjoita Matlab-funktio `ReflKo` (viittaa kysymykseen ”refleksiivinenkö”), joka palauttaa arvon 1, jos syötteenä annettua matriisia vastaava relaatio on refleksiivinen, ja muutoin arvon 0. Testaa sitä yllä määritellyillä matriiseilla `M` ja `S`.

b) Miten tarkastetaan Matlabissa kahden matriisin samuus? _____

Kirjoita relaation symmetrisyyttä testaava funktio `SymmKo`.

c) Miten tarkastetaan Matlabissa onko relaatio toisen osajoukko? _____

Kirjoita transitiivisuutta testaava funktio `TransKo`.

8. Kirjoita funktio `EkvivKo`, joka palauttaa arvon 1, jos syötteenä annettu matriisi on ekvivalenssirelaation matriisi, ja arvon 0 muutoin. Funktion pitää siis tutkia ekvivalenttisuuden määritelmän mukaisesti relaation refleksiivisyys, symmetrisyys ja transitiivisuus. Kukin havaittu ominaisuus funktion pitää tulostaa tekstinä näytölle: esimerkiksi, jos relaatio on transitiivinen, tulostetaan sana ’transitiivinen’.

9. **Antisymmetrisyys ja järjestys.** Järjestysrelaatioita tarkastellaan myöhemmin lisää, mutta muodosta jo tässä antisymmetrisyyttä testaava funktio `Antisko`. Voit käydä vaikkapa `for`-silmukoilla läpi diagonaalin yläpuolen, tai sitten tempuilla matriisien logiikka- ja samuustesteillä.